



Cisco MDS 9000 Family Cookbook for Cisco MDS SAN-OS Release 3.1

Seth Mason
Venkat Kirishnamurthy

October 2007

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-14856-01

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

P, the Cisco logo, and Welcome to the Human Network are trademarks of Cisco Systems, Inc.; Changing the Way We Work, Live, Play, and Learn is a service mark of Cisco Systems, Inc.; and Access Registrar, Aironet, BPX, Catalyst, CCDA, CCDP, CCIE, CCIP, CCNA, CCNP, CCSP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unity, Enterprise/Solver, EtherChannel, EtherFast, EtherSwitch, Fast Step, How Me Browsing, FormShare, GigaDrive, HomeLink, Internet Quotient, IOS, iPhone, IP/TV, iQ Expertise, the iQ logo, iQ Net Readiness Scorecard, iQuick Study, iStream, Linksys, MeetingPlace, MGX, Networkers, Networking Academy, Network Registrar, PIX, ProConnect, ScriptShare, SMARTnet, StackWise, The Fastest Way to Increase Your Internet Quotient, and TransPath are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

Other trademarks mentioned in this document or Website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0710R)

Cisco MDS 9000 Family Cookbook for Cisco MDS SAN-OS Release 3.1
© 2007 Cisco Systems, Inc. All rights reserved.

ISBN 978-0-6151-7888-2

Comments: mds-cookbook@cisco.com



CONTENTS

Foreword ix

Preface xi

- Audience i-xi
- Organization i-xii
- About the Authors i-xiii
- Document Conventions i-xiii
- Related Documentation i-xv

CHAPTER 1

Managing a Cisco MDS 9000 Switch 1-1

- Using SNMP to Monitor MDS Switches 1-1
 - Events 1-1
 - Thresholds 1-2
 - Third Party Management Application Configuration 1-3
 - Advanced Cisco MDS Monitoring 1-10
- Cisco Fabric Services 1-10
 - Fabric Manager and CFS 1-12
 - How Does This Work? 1-12
 - CFS CLI Commands 1-13
 - Which Switches are CFS Capable? 1-13
 - What CFS Applications Do I Have and What Is Their Scope? 1-13
 - Why Am I Locked Out of An Application by CFS? 1-14
- Command Scheduler 1-14
 - Automated Switch Configuration Backup 1-15
- Copying Files to and from a Switch 1-17
 - Copying Files Using the CLI 1-17
 - Secure Copy Protocol 1-17
 - Secure File Transfer Protocol 1-18
- Managing Files on the Standby Supervisor 1-18
 - Delete a File from the Standby Supervisor 1-18
 - Deleting a File using Device Manager 1-19
- Firmware Upgrades and Downgrades 1-20
 - Upgrading Firmware with the CLI 1-20
 - Downgrading Firmware with the CLI 1-22

| | |
|--|------|
| Upgrading Firmware with Fabric Manager | 1-23 |
| Password Recovery | 1-26 |
| Installing a License | 1-28 |
| Using the CLI to Install a License | 1-28 |
| Using Fabric Manager to Install a License | 1-29 |
| Which Feature Enables the License Grace Period? | 1-32 |
| Check License Usage With Fabric Manager | 1-32 |
| Check with the CLI | 1-33 |
| Copying Core Files from the Switch | 1-34 |
| Restoring a Fixed Switch Configuration | 1-34 |
| Configuring an NTP Server | 1-37 |
| Configuring NTP with CFS | 1-37 |
| Configure NTP without CFS | 1-38 |
| What to Do Before Calling TAC | 1-39 |
| Saving the Configuration Across the Fabric | 1-41 |
| Device Aliases | 1-41 |
| Standard Device Aliases | 1-41 |
| Enhanced Device Aliases | 1-42 |
| Zone Set Output with Enhanced Device Aliases | 1-42 |
| Manipulating Device Aliases with the CLI | 1-43 |
| Displaying Device Aliases with the CLI | 1-43 |
| Creating Device Aliases with the CLI | 1-43 |
| Converting Fibre Channel Aliases to Device Aliases | 1-44 |
| Device Aliases with Fabric Manager | 1-46 |
| Enabling Fabric Manager to use Device Aliases | 1-46 |
| Creating a Device Alias for an Existing Device | 1-46 |
| Creating a Device Alias for a New Device | 1-47 |
| Implementing Syslog | 1-48 |
| Configuring Call Home | 1-50 |
| What are Alert Groups? | 1-50 |
| Configure Call Home to Send All Notifications to a Single E-Mail Address | 1-51 |

CHAPTER 2
Managing Fabric Manager Server 2-1

| | |
|--|-----|
| Managing Fabric Manager | 2-1 |
| Optimizing Fabric Manager Server Performance | 2-1 |
| Installing the Correct Java Runtime Environment | 2-1 |
| Performance Manager Database Sizing | 2-2 |
| Configuring Fabric Manager Server to Use an External Oracle Database | 2-3 |

| | |
|--|------|
| Adjusting Memory Usage of Fabric Manager | 2-4 |
| Authenticating Fabric Manager Through TACACS | 2-4 |
| Operating Fabric Manager Through a Firewall Using SNMP Proxy | 2-5 |
| Configuration Using a Non-NAT Packet Filter | 2-6 |
| Performance Manager Using Fabric Manager Server | 2-8 |
| Creating Flows Within Fabric Manager | 2-8 |
| Creating a Collection in Performance Manager | 2-10 |

CHAPTER 3**Security and Access Management 3-1**

| | |
|---|------|
| Creating a User Role | 3-2 |
| Creating a Role with Device Manager | 3-2 |
| Creating a Role with CLI | 3-6 |
| Creating User Accounts | 3-7 |
| User Accounts Through Command-Line | 3-7 |
| User Accounts Through Fabric Manager | 3-7 |
| Configuring TACACS+ with Cisco SecureACS | 3-8 |
| Authentication and Authorization with TACACS+ | 3-9 |
| Configuring the SecureACS Server | 3-9 |
| Configuring TACACS+ on the MDS Switch | 3-14 |
| Accounting with TACACS+ | 3-15 |
| Configuring the MDS Switch to Use TACACS Accounting | 3-16 |
| Configuring SecureACS to Receive TACACS+ Accounting | 3-16 |
| Providing Password-Free Access Using SSH | 3-19 |
| Disabling the Web Server | 3-21 |

CHAPTER 4**Physical Interfaces 4-1**

| | |
|--|-----|
| Configuring Fibre Channel Ports | 4-2 |
| Port Description | 4-2 |
| Port Speed | 4-3 |
| Port Mode Auto | 4-3 |
| Port Mode E | 4-3 |
| Configuring and E Port on a 32-Port Module | 4-4 |
| Configuring an E Port on 24- and 48-Port Modules | 4-4 |
| Configuring Trunking E Ports | 4-4 |
| Trunk Port Mode | 4-4 |
| Configuring Trunk Ports to Filter-Specific VSANs | 4-5 |
| Port Mode F | 4-5 |
| Port Mode FL | 4-5 |
| Port Mode Fx | 4-5 |

| | |
|--|------|
| Port Mode SD | 4-6 |
| Port Mode ST | 4-6 |
| Port Mode TL | 4-6 |
| Enabling Port Beacons | 4-6 |
| Oversubscription Management of Ports and Rate Limiting | 4-7 |
| Strict Oversubscription Mode Recipes | 4-8 |
| Unlimited Oversubscription Mode Recipe | 4-10 |
| Configuring Gigabit Ethernet Ports | 4-11 |
| Configuring VRRP | 4-12 |
| Implementing WWN-Based VSANs | 4-13 |
| Adding Existing Devices to DPVM | 4-15 |
| Adding New Devices to DPVM | 4-18 |
| Modifying the VSAN Assignment of a DPVM Entry | 4-20 |
| DPVM Conflicting Entries | 4-22 |
| DPVM with the CLI | 4-23 |
| Adding Existing Devices to DPVM | 4-23 |
| Adding New Devices to DPVM | 4-23 |
| Modifying the VSAN Assignment of a DPVM Entry | 4-24 |

CHAPTER 5

Logical Interfaces 5-1

| | |
|---|------|
| PortChannels | 5-1 |
| Quiesce a PortChannel or ISL Link | 5-1 |
| Creating a PortChannel Using Fabric Manager | 5-2 |
| Creating a PortChannel from the CLI | 5-5 |
| Adding a New Member to a PortChannel Using Fabric Manager | 5-7 |
| Adding New Members to a PortChannel from the CLI | 5-9 |
| Modifying the VSAN Allowed List on a PortChannel Using Fabric Manager | 5-10 |
| Modifying the VSAN Allowed List on a PortChannel From the CLI | 5-10 |

CHAPTER 6

VSANs 6-1

| | |
|---|------|
| Creating a VSAN and Adding Interfaces Using Fabric Manager | 6-2 |
| Modifying VSAN Attributes with Fabric Manager | 6-5 |
| Changing the Domain ID and Its Configuration of VSAN Using Fabric Manager | 6-6 |
| Changing the FCID Configuration of VSAN Using Fabric Manager | 6-9 |
| Modifying VSAN Attributes with the CLI | 6-10 |
| Creating a VSAN on a Single Switch and Adding an Interface | 6-10 |
| Setting VSAN Interop Mode | 6-10 |
| Interop Mode 1 | 6-12 |
| Interop Mode 2 | 6-12 |

Interop Mode 3 6-12**Interop Mode 4 6-12**

Changing the Load-Balancing Scheme 6-13

Sequence Level Load-Balancing (Source_ID, Destination_ID) 6-13

Exchange Level Load-Balancing (S_ID, D_ID, OX_ID) 6-13

Converting an Existing VSAN to Static Domain ID and Enabling a Persistent FCID Using the CLI 6-13

Changing the Domain ID in a VSAN and Making It Static 6-14

Assigning a Predetermined FC ID to a pWWN 6-15

Assigning a New Predetermined FCID to a Currently Logged In pWWN 6-15

CHAPTER 7**Zoning 7-1**

Enhanced Zoning 7-2

Enabling Enhanced Zoning 7-3

Enabling Enhanced Zoning with the CLI 7-4

Enabling Enhanced Zoning with Fabric Manager 7-4

Displaying a User with the Current Lock in CLI and Fabric Manager 7-5

Zone Sets 7-6

Distributing Zone Sets 7-6

Distributing Zone Sets Automatically 7-7

Distributing Zone Sets Manually 7-7

Zones 7-8

Creating a Zone and Adding It to a Zone Set with Fabric Manager 7-8

Creating Non-pWWN-Based Zones 7-14

Creating a Zone and Adding It to a Zone Set with the CLI Standalone Method 7-15

Creating a Device Alias-Based Zone with the CLI 7-16

Creating a pWWN-based Zone with the CLI 7-17

Creating a Zone and Adding it to a Zone Set with the CLI Inline Method 7-19

Creating a FC Alias-Based Zone with the CLI 7-20

Creating an Interface-Based Zone with the CLI 7-22

CHAPTER 8**Inter-VSAN Routing 8-1**

IVR Core Components 8-2

IVR Topology 8-2

Auto Topology 8-2

Transit VSANs 8-3

Configuring a Three Switch, Two Transit VSAN Topology with CFS 8-4

IVR Zones and Zone Sets 8-7

IVR with CFS 8-8

IVR-1 8-10

| | |
|---|------|
| Enabling IVR-1 | 8-10 |
| Enabling IVR-1 with the CLI | 8-10 |
| Enabling IVR-1 with Fabric Manager | 8-11 |
| Configuring a Single Switch and Two VSANs | 8-12 |
| Creating the IVR Topology | 8-12 |
| Creating the IVR Zone Set and Zones | 8-13 |
| IVR-2 with FC NAT | 8-15 |
| Enabling IVR-2 (FC NAT) | 8-15 |
| Upgrading from IVR-1 to IVR-2 | 8-19 |
| Configuring Persistent FC IDs in IVR from the CLI | 8-21 |
| Configuring Persistent FC IDs in IVR Using Fabric Manager | 8-22 |
| Configuring a Single Switch with Two VSANs | 8-25 |
| Adding a New IVR-Enabled Switch | 8-28 |

CHAPTER 9

FCIP 9-1

| | |
|--|------|
| Enabling FCIP | 9-2 |
| Configuring FCIP on a Switch with CLI | 9-2 |
| Enabling FCIP Write Acceleration | 9-6 |
| Enabling FCIP Compression | 9-7 |
| Enabling Tape Acceleration | 9-9 |
| Enabling Tape Acceleration from the CLI | 9-9 |
| Tuning FCIP | 9-12 |
| TCP Tuning: Latency and Available Bandwidth | 9-12 |
| Configuring Multiple FCIP Tunnels Using a Single Gigabit Ethernet Port | 9-13 |
| Configuring FCIP Using Fabric Manager | 9-21 |
| Enabling Tape Acceleration | 9-36 |
| Testing and Tuning the FCIP Link with SET | 9-37 |

CHAPTER 10

iSCSI 10-1

| | |
|---|-------|
| iSLB Configuration Mode | 10-1 |
| Configuring iSLB on an MDS Switch | 10-2 |
| Configuring iSCSI on an MDS Switch in Transparent Mode | 10-7 |
| Configuring iSCSI on the MDS Switch in Proxy Initiator Mode | 10-11 |
| Configuring iSCSI Client Initiators on Hosts | 10-15 |
| Configuring iSCSI on Microsoft Windows | 10-15 |



Foreword

It has been a pleasure to be associated with Seth and Venkat over the last few years and to see their expertise and extensive customer experience with storage and SAN technologies translate into a book that will provide help and guidance to customers in designing and managing their SANs. It is very good for me to see that their “labor of love” over the last few years is coming to fruition.

Recent trends in the data center are around consolidation, virtualization, and business continuity. The business drivers for consolidation and virtualization are driven by requirements for achieving higher asset utilization (storage, server, network, and so on), lowering operating expenses, lowering power consumption, and having integrated and simplified management.

Storage and SAN consolidation, along with storage replication for disaster recovery are a significant piece of the overall data center consolidation projects. From a storage perspective, to achieve these business benefits, customers are looking at consolidating their disparate application-specific SANs into a larger integrated physical SAN, with the ability to create virtual SANs. The other important area, specific to disaster recovery, is in leveraging technologies such as Fiber Channel, Optical, and IP for local, metro, and long-distance replication for storage.

The *Cisco MDS 9000 Cookbook for Cisco MDS SAN-OS Release 3.1* from Seth and Venkat has real-world “how to” examples to help customers and practitioners in the storage and SAN area. They explain how to use MDS switches in designing large scale consolidated SAN architectures, creating virtual SANs, implementing disaster recovery and replication scenarios, and creating security and management best practices. The book focuses on providing practical, topology-based, configuration-driven case studies that help simplify design and deployment scenarios with the Cisco MDS 9000 Family switches.

Seth and Venkat have been involved with a significant number of large and complex customer designs and deployments in the area of storage and SANs. They have also been prolific in presenting topics to our customers and partners at technical seminars that address Data Center consolidation, business continuity, and storage area networking. This book—the culmination of their expertise, experience, and two years of hard work—will be helpful to customers and technical folks who are looking for a practical guide to designing and deploying their SAN infrastructures.

Faiyaz Shahpurwala
VP of Advanced Services, Data Center Practice
Cisco Systems



Preface

This document addresses the configuration and implementation of fabrics using the Cisco MDS 9000 Family of Fibre Channel Switch and Director Class products. The configuration procedures and components provided have been tested and validated by Cisco's Solution-Interoperability Engineering department.

This cookbook provides simplified, concise recipes (procedures) for tasks that might be required to configure a Cisco MDS 9000 Family switch. This guide does not replace the MDS 9000 Family Configuration Guides, but compliments them with concise procedures for specific tasks..

Within this book, some sections include “tips” that look like this:



Tip

These tips are best practices for implementing the features of the Cisco MDS 9000 platform. They are a result of in-depth knowledge of the platform, as well as extensive experience implementing Storage Area Networks (SANs).

Audience

This document is designed for use by Cisco TAC, Sales, Support Engineers, Professional Service Partners, Systems Administrators, and others responsible for the design and deployment of SANs in the data center environment.

This is a field-driven book, meaning that the intended audience (storage administrators, technical support engineers, SEs, and CEs) is also the source of information for these procedures. Their requirements for a procedure are what determine the content.

If there are procedures that you feel should be covered in this book, or if you have any other comments or questions, please notify us through e-mail at mds-cookbook@cisco.com. Please state the document name, page number, and details of the request.

Organization

This guide is organized as follows:

| Chapter | Title | Description |
|-----------|----------------------------------|---|
| Chapter 1 | Managing a Cisco MDS 9000 Switch | This chapter discusses those topics that leverage features for managing or configuring the MDS platform. It provides the recipes to control the switch itself rather than controlling the flow of data. Topics such as Device Aliases and Cisco Fabric Services are addressed, which are powerful tools that previously might be unknown to the reader. These tools, while often overlooked, should be deployed before the first host or disk array connecting to it. |
| Chapter 2 | Managing Fabric Manager Server | This chapter focuses on the aspect of deploying, configuring, and optimizing the graphical interfaces for configuring and monitoring the MDS platform. Cisco Fabric Manager plays an important role in any deployment and this chapter addresses the issues and tips for a successful deployment of Fabric Manager Server and its web component. |
| Chapter 3 | Security and Access Management | This chapter builds upon Chapter 1 in that it provides recipes for Authentication, Authorization, and Accounting. The recipes contained enable the switch to be able to defend itself against user errors or unauthorized access. |
| Chapter 4 | Physical Interfaces | Starting at the lowest level in the Data Center, the physical layer, this chapter provides the different recipes necessary for working on the hardware interfaces. Topics such as port types and guaranteeing bandwidth are addressed. |
| Chapter 5 | Logical Interfaces | This chapter builds upon the physical interfaces chapter, and provides recipes for configuring PortChannels and Trunking. |
| Chapter 6 | VSANs | This chapter covers one of the core technologies of the MDS platform: VSANs. Creating, modifying, adding ports and working with domain-manager are covered. |
| Chapter 7 | Zoning | This chapter provides recipes for working with Fiber Channel zoning, the primary method of device access control within a SAN. Different types of zoning are covered as well as all the various member types and methods for configuring zoning. |
| Chapter 8 | Inter-VSAN Routing | This chapter builds upon the VSAN and Zoning chapters, so that you are able to route frames between VSANs using the inter-VSAN Routing (IVR) feature. It provides the background and insight into IVR topologies, configuration distribution, and zoning. |

| Chapter | Title | Description |
|------------|-------|---|
| Chapter 9 | FCIP | This chapter discusses how to deploy Fibre Channel over IP (FCIP), which is used to connect SANs together over an existing IP network. This chapter is a must-read for those looking to deploy MDS 9000 switches into disaster-recovery environments. |
| Chapter 10 | iSCSI | This chapter covers the fundamentals of deploying the iSCSI feature set of the MDS platform. Used for connecting mid range and low-end servers to their storage over an existing IP network. The recipes contained demonstrate how to effectively deploy iSCSI. |

About the Authors

Seth Mason is a Network Consulting Engineer with the DCN team at Cisco Systems. His areas of expertise are SAN migration, Disaster Recovery, interoperability, and IVR. He graduated from Auburn University in 1998 with a Bachelor of Computer Engineering and has focused on SANs ever since, including as Product Engineer with IBM's Storage Subsystems Group, Silicon Valley Operations team lead with StorageNetworks, and NCE with Andiamo Systems. Seth has continued to further his expertise in storage by authoring both the *MDS-9000 Family Cookbook for SAN-OS 1.x* and *MDS-9000 Family Cookbook for SAN-OS 2.x*, as well as the *MDS-9000 Switch to Switch Interoperability Configuration Guide*. He is a member of the team that authored the CCIE exam in Storage Networking.

Venkat Kirishnamurthy is a Network Consulting Engineer with the DCN team at Cisco Systems. His areas of expertise are SAN design, migration, and storage replication for disaster recovery. He graduated from Bangalore University in 1992 with a Bachelor of Electronics and Communications Engineering. Since then he has worked as a Systems Administrator at Hughes Software Systems, India and as a Senior Systems Administrator and Senior Storage Administrator at Cisco Systems. Venkat has continued his storage expertise by authoring SAN migration guides for HPUX and Solaris hosts, both the *MDS-9000 Family Cookbook for SAN-OS 1.x* and *MDS-9000 Family Cookbook for SAN-OS 2.x*. He is a member of the team that authored the CCIE exam for Storage Networking.

Document Conventions

Command descriptions use these conventions:

| Convention | Indication |
|----------------------|---|
| boldface font | Commands and keywords are in boldface. |
| <i>italic font</i> | Arguments for which you supply values are in italics. |
| [] | Elements in square brackets are optional. |
| { x y z } | Required alternative keywords are grouped in braces and separated by vertical bars. |
| [x y z] | Optional alternative keywords are grouped in brackets and separated by vertical bars. |
| string | A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks. |

Screen examples use these conventions:

| Convention | Indication |
|-----------------------------|---|
| screen font | Terminal sessions and information the switch displays are in <code>screen font</code> . |
| boldface screen font | Information you must enter is in boldface screen font . |
| <i>italic screen font</i> | Arguments for which you supply values are in <i>italic screen font</i> . |
| < > | 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:



Note

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



Tip

Means *the following information will help you solve a problem*. These tips are suggested as best practices and are based on in-depth knowledge of the Cisco MDS 9000 family platform and experience implementing SANs.



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 SAN-OS Documentation Locator at:

http://www.cisco.com/en/US/products/ps5989/products_documentation_roadmap09186a00804500c1.html.

Release Notes

- *Cisco MDS 9000 Family Release Notes for Cisco 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*

Compatibility Information

- *Cisco MDS 9000 SAN-OS Hardware and Software Compatibility Information*
- *Cisco MDS 9000 Family Interoperability Support Matrix*
- *Cisco MDS Storage Services Module Interoperability Support Matrix*
- *Cisco MDS SAN-OS Release Compatibility Matrix for Storage Service Interface Images*

Regulatory Compliance and Safety Information

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

Hardware Installation

- *Cisco MDS 9124 Multilayer Fabric Switch Quick Start Guide*
- *Cisco MDS 9500 Series Hardware Installation Guide*
- *Cisco MDS 9200 Series Hardware Installation Guide*
- *Cisco MDS 9100 Series Hardware Installation Guide*

Cisco Fabric Manager

- *Cisco MDS 9000 Family Fabric Manager Quick Configuration Guide*
- *Cisco MDS 9000 Family Fabric Manager Configuration Guide*
- *Cisco MDS 9000 Family Fabric Manager Database Schema*
- *Cisco MDS 9000 Family Data Mobility Manager Configuration Guide*

Command-Line Interface

- *Cisco MDS 9000 Family Software Upgrade and Downgrade Guide*
- *Cisco MDS 9000 Family Storage Services Module Software Installation and Upgrade Guide*
- *Cisco MDS 9000 Family CLI Quick Configuration Guide*
- *Cisco MDS 9000 Family CLI Configuration Guide*
- *Cisco MDS 9000 Family Command Reference*

Troubleshooting and Reference

- *Cisco MDS 9000 Family Troubleshooting Guide*
- *Cisco MDS 9000 Family MIB Quick Reference*
- *Cisco MDS 9000 Family SMI-S Programming Reference*
- *Cisco MDS 9000 Family System Messages Reference*

Installation and Configuration Note

- *Cisco MDS 9000 Family SSM Configuration Note*
- *Cisco MDS 9000 Family Port Analyzer Adapter Installation and Configuration Note*
- *Cisco 10-Gigabit X2 Transceiver Module Installation Note*
- *Cisco MDS 9000 Family CWDM SFP Installation Note*
- *Cisco MDS 9000 Family CWDM Passive Optical System Installation Note*

Obtaining Documentation, Obtaining Support, and Security Guidelines

For information on obtaining documentation, obtaining support, providing documentation feedback, security guidelines, and also recommended aliases and general Cisco documents, see the monthly *What's New* in Cisco Product Documentation, which also lists all new and revised Cisco technical documentation, at:

<http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html>



CHAPTER 1

Managing a Cisco MDS 9000 Switch

This chapter provides recipes for managing a Cisco MDS 9000 switch. These nondata path topics include access control, accounting, event resolution, and monitoring.

Using SNMP to Monitor MDS Switches

Cisco MDS 9000 switches support a large number of MIBs and events to notify administrators and support personnel. The monitoring solution should provide them with the relevant traps or notifications without overwhelming them with unneeded traps. To address this need, a standard list of events and thresholds have been identified for an SAN administrator to monitor.

[Table 1-1](#) lists a subset of the full set of events that the Cisco MDS 9000 switches support. [Table 1-2](#) lists standard thresholds to monitor. Customers have the flexibility to customize the monitoring solution to meet their specific needs. The MIBs listed in this chapter are a baseline to begin implementing your specific monitoring framework.

Events

The Cisco MDS SAN-OS software supports over 100 MIBs and supports Simple Network Management Protocol (SNMP) versions v1, v2, and v3.

Cisco MDS SAN-OS provides the ability to configure traps that are sent out. To enable traps listed in [Table 1-1](#), the following configuration changes are required on the Cisco MDS 9000 switch using the command-line interface (CLI). These changes enable Cisco MDS specific link-up and link-down traps, entity, fcdomain, and zone traps to be forwarded to the monitoring application using the Cisco MDS CLI commands shown here:

```
switch(config)# snmp enable traps link cisco //link interface events
switch(config)# snmp enable traps entity //enables entity events
switch(config)# snmp enable traps fcdomain//fcdomain events
switch(config)# snmp enable traps zone//zone events
```

Table 1-1 MDS Events

| Trap | MIB | Event Name |
|----------|------------------------|-------------|
| Link | | |
| LinkUp | CISCO-IF-EXTENSION-MIB | cieLinkUp |
| LinkDown | CISCO-IF-EXTENSION-MIB | cieLinkDown |

Table 1-1 MDS Events (continued)

| Trap | MIB | Event Name |
|---------------------|------------------------------|--------------------------------|
| (E)ISL Up | CISCO-FC-FE-MIB | fcTrunkIfUpNotify |
| (E)ISL Down | CISCO-FC-FE-MIB | fcTrunkIfDownNotify |
| VSAN | | |
| VSAN Segmentation | CISCO-DM-MIB | dmDomainIdNotAssignedNotify |
| Build Fabric | CISCO-DM-MIB | dmFabricChangeNotify |
| Zone | | |
| Merge Failure | CISCO-ZS-MIB | zoneMergeFailureNotify |
| Zone set Activation | CISCO-ZS-MIB | zoneActivateNotify |
| Sensor | | |
| Temperature | CISCO-ENTITY-SENSOR-MIB | entSensorThresholdNotification |
| FRU | | |
| Fan | CISCO-ENTITY-FRU-CONTROL-MIB | cefcFanTrayStatusChange |
| Power Supply | CISCO-ENTITY-FRU-CONTROL-MIB | cefcPowerStatusChange |
| Module | CISCO-ENTITY-FRU-CONTROL-MIB | cefcModuleStatusChange |
| Redundancy | | |
| Supervisor Failover | CISCO-RF-MIB | ciscoRFSwactNotify |

For more information about these MIBs, see to “Third Party Management Application Configuration” on page 3 of the *Cisco MDS 9000 Family MIB Quick Reference Guide*.

Thresholds

The Threshold Monitor triggers an SNMP event or logs a message when a selected statistic goes over a configured threshold value. Remote Monitoring (RMON) calls this a rising alarm threshold. RMON is an Internet Engineering Task Force (IETF) standard (RFC 2819) monitoring specification that allows various network agents and console systems to exchange network monitoring data. The following definitions are important:

- **Alarm:** Monitors a specific management information base (MIB) object for a specified interval, triggers an alarm at a specified value (rising threshold), and resets the alarm at another value (falling threshold). Alarms can be used with events; the alarm triggers an event, which can generate a log entry or an SNMP trap.
- **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.

Table 1-2 lists MDS thresholds.

Table 1-2 MIB Thresholds

| Threshold Variable | MIB | Object | Value | Sample (sec.) |
|--------------------|-----------------|--------------------|------------|---------------|
| Link Failures | CISCO-FC-FE-MIB | fcIfLinkFailures | 2 | 30 |
| Sync Loss | CISCO-FC-FE-MIB | fcIfSyncLosses | 2 | 30 |
| Signal Loss | CISCO-FC-FE-MIB | fcIfSigLosses | 2 | 30 |
| Invalid Words | CISCO-FC-FE-MIB | fcIfInvalidTxWords | 2 | 30 |
| Invalid CRCs | CISCO-FC-FE-MIB | fcInvalidCrcs | 2 | 30 |
| Link Performance | CISCO-FC-FE-MIB | fcInOctets | 1600000000 | 30 |
| Link Performance | CISCO-FC-FE-MIB | fcOutOctets | 1600000000 | 30 |

Thresholds can be configured through the CLI or using the Cisco Device Manager. Refer to the “Configuring RMON” section in the *Cisco MDS 9000 Family Fabric Manager Configuration Guide*.

Third Party Management Application Configuration

Network Management Systems (NMS) need to be configured to recognize the traps forwarded by the Cisco MDS SAN-OS software. The most common NMS applications on the market are HP OpenView and IBM Tivoli NetView. Both applications have a very similar architecture in terms of how the MIBs are loaded and how the applications identify the incoming traps and present a short message in the console with regards to the event.

Cisco provides executables to integrate events listed in Table 1-1 with HP OpenView and Tivoli NetView applications. For customers using other NMS applications, the event details in Table 1-3 through Table 1-16 should help configure the NMS to recognize Cisco MDS 9000 events.

NOTIFICATION, OBJECTS, DESCRIPTION, and OID represent the information from the MIB. SEVERITY and MESSAGE fields can be customized to customer needs. Use the information in these tables as a guideline.

Table 1-3 Link Down

| Information | Description |
|--------------|---|
| Notification | cieLinkDown |
| Objects | ifIndex, ifAdminStatus, ifOperStatus, ifName, ifType |
| Description | A Cisco Specific linkDown notification signifies that the SNMP entity, acting in an agent role, has detected that the ifOperStatus object for one of its communication links is about to enter the down state from some other state (but not from the notPresent state). The varbinds for this notification indicate the interface information of the communication link. |
| OID | 1.3.6.1.4.1.9.9.276.0.1 |
| MIB | CISCO-IF-EXTENSION-MIB |

Table 1-3 Link Down

| Information | Description |
|-------------|--------------------|
| Severity | Information |
| Message | Interface Down \$4 |

Table 1-4 Link Up

| Information | Description |
|--------------|---|
| Notification | cieLinkUp |
| Objects | ifIndex, ifAdminStatus, ifOperStatus, ifName, ifType |
| Description | A Cisco Specific linkUp trap signifies that the SNMP entity, acting in an agent role, has detected that the ifOperStatus object for one of its communication links left the down state and transitioned into some other state (but not into the notPresent state). The varbinds for this notification indicate the interface information of the communication link. |
| OID | 1.3.6.1.4.1.9.9.276.0.2 |
| MIB | CISCO-IF-EXTENSION-MIB |
| Severity | Information |
| Message | Interface Up \$4 |

**Note**

The fcTrunkIfDownNotify and fcTrunkIfUpNotify events by themselves do not specify the port interface. They are always followed by an cieLinkDown or cieLinkUp events that provide interface information.

Table 1-5 (E)ISL Port Down

| Information | Description |
|--------------|--|
| Notification | fcTrunkIfDownNotify |
| Objects | fcTrunkIfOperStatus, fcTrunkIfOperStatusCause, fcTrunkIfOperStatusCauseDescr, |
| Description | This notification is generated by the agent whenever the fcTrunkIfOperStatus object for this trunk interface is about to enter the down state from some other state. This other state is indicated by the included value of fcTrunkIfOperStatus. |
| OID | 1.3.6.1.4.1.9.9.289.1.3.0.1 |
| MIB | CISCO-FC-FE-MIB |

Table 1-5 (E)ISL Port Down (continued)

| Information | Description |
|-------------|----------------------------------|
| Severity | Information |
| Message | (T)E Port Link Down Notification |

Table 1-6 (E)ISL Port Up

| Information | Description |
|--------------|---|
| Notification | fcTrunkIfUpNotify |
| Objects | fcTrunkOperStatus, fcTrunkIfOperStatusCause, fcTrunkOperStatusCauseDescr |
| Description | This notification is generated by the agent whenever the fcTrunkIfOperStatus object for one of its trunk interfaces has left the down state and transitioned into some other state. This other state is indicated by the included value of fcTrunkIfOperStatus. |
| OID | 1.3.6.1.4.1.9.9.289.1.3.0.2 |
| MIB | CISCO-FC-FE-MIB |
| Severity | Information |
| Message | (T)E Port Link Up Notification |

Table 1-7 VSAN Status

| Information | Description |
|--------------|--|
| Notification | vsanStatusChange |
| Objects | notifyVsanIndex, vsanAdminState, vsanOperState |
| Description | A state change notification is generated whenever vsanOperState is changed. The index and both states of the VSAN after the change, are included as variables in the notification. vsanAdminState : active(1), suspended(2) vsanOperState : up(1), down(2) |
| OID | 1.3.6.1.4.1.9.9.282.1.3.0.1 |
| MIB | CISCO-VSAN-MIB |
| Severity | Information |
| Message | VSAN \$1 \$3 (Up(1), Down(2)) |

Table 1-8 VSAN Segmentation

| Information | Description |
|--------------|--|
| Notification | dmDomainIdNotAssignedNotify |
| Objects | notifyVsanIndex, cffFcFeElementName |
| Description | <p>If a Domain ID is not configured or assigned on a VSAN, then the switch may isolate E_ports on that VSAN. The conditions are:</p> <ul style="list-style-type: none"> • If the Domain Manager is enabled on the local switch and its request for a configured static Domain ID is rejected or no other Domain ID is assigned, then the E ports are isolated. • If the domain manager is not enabled and if a static Domain ID is not configured on the VSAN, then the switch will isolate all of its E ports on the VSAN. <p>This notification contains the vsanIndex of the VSAN on which the condition happened.</p> |
| OID | 1.3.6.1.4.1.9.9.302.1.3.0.1 |
| MIB | CISCO-DM-MIB |
| Severity | Critical |
| Message | Domain ID not configured or assigned on VSAN \$1, switch may isolate E ports on that VSAN. |

Table 1-9 Build Fabric (BF) or Reconfigure Fabric (RCF) Event

| Information | Description |
|--------------|----------------------|
| Notification | dmFabricChangeNotify |
| Objects | notifyVsanIndex |

Table 1-9 Build Fabric (BF) or Reconfigure Fabric (RCF) Event (continued)


| Information | Description |
|-------------|---|
| Description | <p>This notification is sent whenever a switch sends or receives a Build Fabric (BF) or a ReConfigure Fabric (RCF) message on a VSAN.</p> <p>A switch can receive or issue a BuildFabric (BF) or a ReConfigureFabric (RCF) message under following conditions:</p> <ul style="list-style-type: none"> • A new link causes two disjointed fabrics in a VSAN to merge into one fabric. The sent/received message is a BF if the Domain ID lists on the disjoint fabric does not overlap and it is a RCF if they overlap. • An upstream principal ISL connects to the principal switch and other switches if a VSAN fails. A BF is issued to see if there is an alternative path to the principal switch. If no paths exit, then a RCF is issued. • A switch asks for a different set of Domain IDs than the currently assigned list, the principal switch would issue a RCF. <p>The notification is not sent if a 'dmNewPrincipalSwitchNotify' notification is sent for the same transition. This notification contains the vsanIndex of the VSAN on which RCF was issued.</p> <div>  <p>Note BF is a nondisruptive event, while RCF is disruptive.</p> </div> |
| OID | 1.3.6.1.4.1.9.9.302.1.3.0.3 |
| MIB | CISCO-DM-MIB |
| Severity | Information |
| Message | Fabric Configuration Notification for VSAN \$1 |

Table 1-10 Zone Merge Failure Notification Event

| Information | Description |
|--------------|--|
| Notification | zoneMergeFailureNotify |
| Objects | ifIndex, zoneMergeFailureVSANNum |
| Description | <p>This notification is generated whenever there is a zone merge failure. If all VSANs on a link have a zone-merge failure at the same time, then just one notification is generated in which the zoneMergeFailureVSANNum object has a zero value.</p> |
| OID | 1.3.6.1.4.1.9.9.294.1.4.0.2 |
| MIB | CISCO-ZS-MIB |

Table 1-10 Zone Merge Failure Notification Event

| Information | Description |
|-------------|--|
| Severity | Alert |
| Message | Zone Merge Failure Notification for VSAN \$2 |

Table 1-11 Activate Zone Set Notification Event

| Information | Description |
|--------------|--|
| Notification | zoneActivateNotify |
| Objects | zoneSetActiveResult, zoneSwitchWwn |
| Description | This notification is generated whenever a zone set is activated or deactivated on a VSAN. The zoneSetActiveResult object denotes the outcome of the activation or deactivation. The zoneSwitchWwn object represents the WWN of the local device. |
| OID | 1.3.6.1.4.1.9.9.294.1.4.0.6 |
| MIB | CISCO-ZS-MIB |
| Severity | Information |
| Message | Zone Activation Status on Switch WWN \$2: \$1 (activateSuccess(1), activateFailure(2), deactivateSuccess(3), deactivateFailure(4), inProgress(5), newEntry(6)) |

Table 1-12 Temperature Notification Event

| Information | Description |
|--------------|--|
| Notification | entSensorThresholdNotification |
| Objects | entSensorThresholdValue, entSensorValue |
| Description | The sensor value crossed the threshold listed in entSensorThresholdTable. This notification is generated once each time the sensor value crosses the threshold. The agent implementation guarantees prompt, timely evaluation of threshold, and generation of this notification. |
| OID | 1.3.6.1.4.1.9.9.91.2.0.1 |
| MIB | CISCO-ENTITY-SENSOR-MIB |
| Severity | Information |
| Message | “OID Query Result” exceeded the threshold value \$1. Current Value is \$2. |

Table 1-13 Fan Tray Status Notification Event

| Information | Description |
|--------------|---|
| Notification | cefcFanTrayStatusChange |
| Objects | cefcFanTrayOperStatus |
| Description | This notification generated when the value of cefcModuleOperStatus changes. |
| OID | .1.3.6.1.4.1.9.9.117.2.0.6 |
| MIB | CISCO-ENTITY-FRU-CONTROL |
| Severity | Warning |
| Message | Fan Tray Status: \$1 (unknown(1), up(2), down(3), warning(4)) |

Table 1-14 Power Status Change Notification Event

| Information | Description |
|--------------|--|
| Notification | cefcPowerStatusChange |
| Objects | cefcFRUPowerOperStatus, cefcFRUPowerAdminStatus |
| Description | The cefcFRUPowerStatusChange notification indicates that the power status of a field replaceable unit (FRU) has changed. The varbind for this notification indicates the entPhysicalIndex of the FRU, and the new operational status of the FRU. |
| OID | .1.3.6.1.4.1.9.9.117.2.0.2 |
| MIB | CISCO-ENTITY-FRU-CONTROL-MIB |
| Severity | Warning |
| Message | Power status change: Operational Status \$1 (2 - on, 3 - Off) |

Table 1-15 Module Status Change

| Information | Description |
|--------------|--|
| Notification | cefcModuleStatusChange |
| Objects | cefcModuleOperStatus, cefcModuleStatusLastChangeTime |
| Description | This notification is generated when the value of cefcModuleOperStatus changes. It can be used by an NMS to update the status of the module it is managing. |
| OID | .1.3.6.1.4.1.9.9.117.2.0.1 |
| MIB | CISCO-ENTITY-FRU-CONTROL-MIB |
| Severity | Warning |
| Message | Module Status Changed: \$1 (2-OK, 3-Disabled, 5-Boot, 6-Self Test, Other-Misc) |

Table 1-16 Redundancy

| Information | Description |
|--------------|--|
| Notification | cciscoRFSwactNotif |
| Objects | cRFStatusUnitId, sysUpTime, cRFStatusLastSwactReasonCode |
| Description | A SWACT notification is sent by the newly active redundant unit whenever a switch activity occurs. Where a SWACT event may be indistinguishable from a reset event, a network management station should use this notification to differentiate the activity. sysUpTime is the same sysUpTime defined in the RFC-1213 MIB. |
| OID | .1.3.6.1.4.1.9.9.176.2.0.1 |
| MIB | CISCO-RF-MIB |
| Severity | Warning |
| Message | Supervisor switchover notification. Reason \$3 (No Action(0), Peer Reload(1), Reload (2), Switch Activity (3), Force Switch Activity(4)) |

Advanced Cisco MDS Monitoring

As mentioned earlier, the list of events and thresholds identified as part of the standard monitoring are a subset of the overall set of events and threshold parameters. Customers interested in customizing monitoring capabilities to meet specific needs can do so by identifying the events and customizing their NMS to recognize the events. For a complete list of MIBs supported by the MDS, refer to the [Cisco MDS 9000 Family MIB Quick Reference Guide](#).

Cisco Fabric Services

Starting with Cisco SAN-OS Release 2.0, Cisco MDS 9000 switches are able to propagate and synchronize the configuration of an application on multiple switches across the fabric. This infrastructure, Cisco Fabric Services (CFS), provides the underlying transport for applications such as NTP, device aliases, and IVR to distribute configurations to other switches in the fabric. This feature provides a central point of management for any of the supported applications.

Before Cisco SAN-OS 2.0, on each switch in the fabric, the administrator either had to configure an application manually, using host-based scripting, or using Fabric Manager. With CFS, the administrator executes commands from one switch and they are distributed to the rest of the switches in the fabric. In addition, the CFS protocol provides application locking so that two administrators cannot simultaneously perform configuration changes to the same application.

CFS uses common terminology across its supported applications:

- **Pending Database:** When configuration changes are made to a CFS application, they are first made to the pending database and then distributed to all switches in the fabric. To activate these changes into the switch's running configuration, execute an explicit **commit** command. Alternatively, you can clear the application's pending database by entering an explicit **abort** command.

- a. **Locking:** Before modifying the pending database, the application uses the CFS transport to obtain a lock, preventing other users and switches from modifying the pending database. Applications outside the scope of the lock can still be modified.
 - When initializing the configuration, the application first attempts to obtain a lock. The CFS infrastructure knows which switch and user has obtained the lock.
- **Scope:** The scope of an application can be either physical or logical. This scope determines whether multiple users can simultaneously modify the same application.
 - A physical scope encompasses all the switches in the physical fabric such as NTP. While an NTP lock is active, no other user can modify NTP within the physical fabric. There are two types of physical scope:

Physical-fc: This scope encompasses all MDS switches connected through Fibre Channel or FCIP.

Physical-all: This scope encompasses all MDS switches connected through Fibre Channel or IP. (In SAN-OS 3.x, CFS can be configured to traverse IP.)
 - A logical scope encompasses only the VSAN being configured. For example, port security could be locked in a VSAN. While that port security lock is active, no other user can modify port security for that particular VSAN. However, port security could be modified for another VSAN since it is outside of the scope of the lock.
- **Merge Control:** If two fabrics are merged, each application is responsible for merging its configuration with that of the same application in the other physical fabric. The basic rule for merging is that a union of the two configurations is produced. However, conflicting entries are not merged. Conflicting entries must be manually created in the merged configuration.

**Note**

Failure to fully merge a CFS application when merging two fabrics, will *not* isolate the ISL.

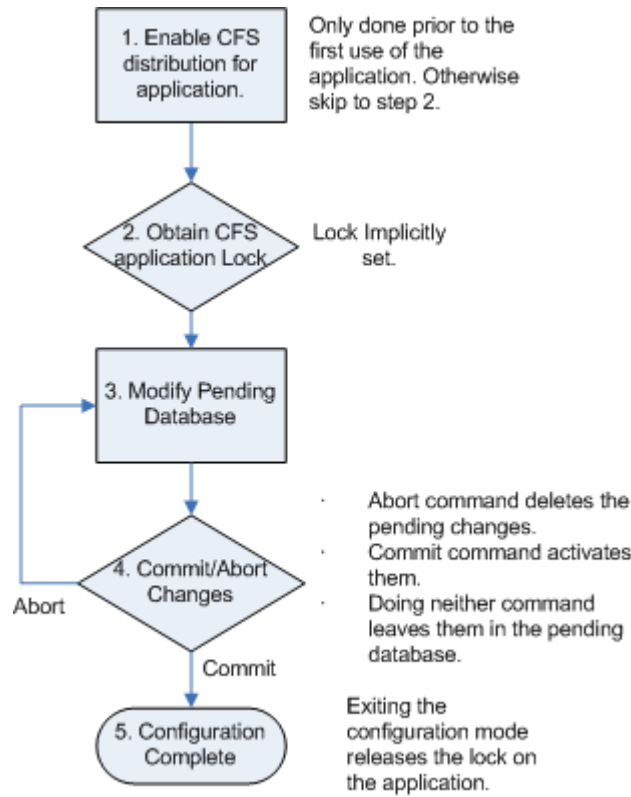
**Tip**

If CFS is used with an application, all the switches in the fabric should be configured to use CFS for that application. For example, if there are five switches in a fabric, and Network Time Protocol (NTP) will be configured leveraging CFS, all five switches should have NTP leveraging CFS.

As [Figure 1-1](#) shows, a CFS application works as follows:

1. Before the first configuration, CFS enables distribution for the application, then enters the configuration mode for the specified application.
2. The local switch requests an application lock from the other switches in the fabric according to the scope of the application (VSAN or physical). If available, other switches grant the lock to the local switch. If the lock is not available, access to the application's pending database is denied.
3. Changes are made to the pending database. The changes are then either explicitly committed or aborted.
4. The local switch informs the other switches in the scope to commit the changes then the lock is released. Until the lock is released, other users on other switches cannot make changes to the locked application. However, other applications can still be modified.

Figure 1-1 CFS Application Flow



Fabric Manager and CFS

Before version Cisco SAN-OS 2.0, Fabric Manager had the ability to configure multiple switches simultaneously by sending configuration commands to all selected switches. Fabric Manager still has this ability, but optionally can use the underlying transport of CFS to do the same thing. You still need to commit the changes, as committing is an explicit activity.



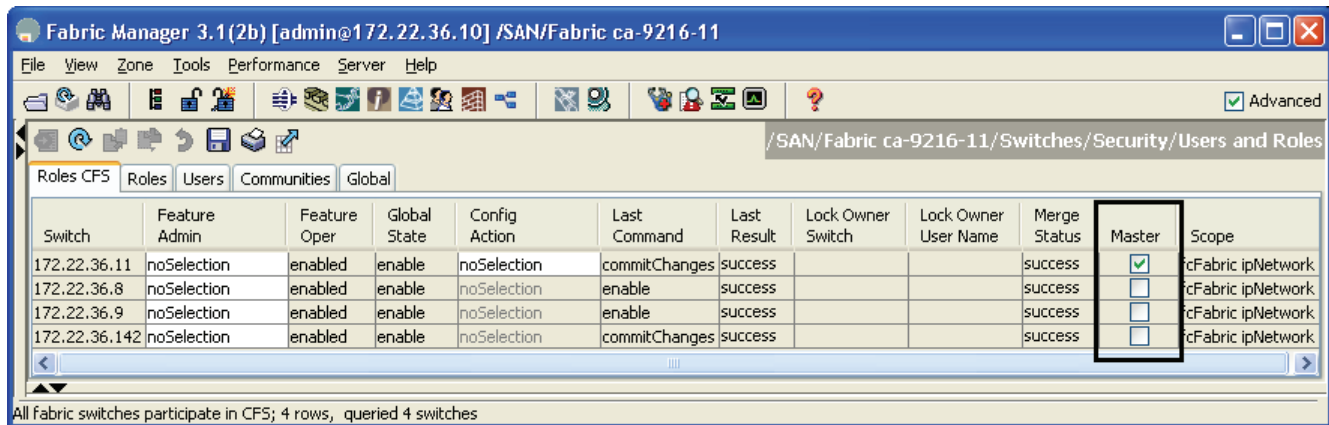
Tip

If an application is configured to use CFS in the fabric, CFS should be enabled for that application on all switches in the fabric. Fabric Manager can use either CFS or the legacy method, but not both.

How Does This Work?

If Fabric Manager uses CFS to distribute a configuration, one switch performs the locking and distribution. This switch is referred to as the master switch (see [Figure 1-2](#)). The master switch is determined by its WWN: the switch with the lowest WWN becomes the master switch.

Figure 1-2 CFS Master in Fabric Manager



CFS CLI Commands

You do not interact with CFS directly because it is an underlying structure. Instead, use applications that leverage CFS, for example NTP or DPVM. It is more important to know the status of an NTP merge or commit than to know how CFS is set up. However, there are some situations when only CFS can provide the required information.

Which Switches are CFS Capable?

The **show cfs peers** command lists switches that can use CFS.

```
172.22.36.9# show cfs peers
```

Physical Fabric

```
-----
Switch WWN                IP Address
-----
20:00:00:05:30:00:86:9e  172.22.36.9    [Local]
20:00:00:05:30:00:68:5e  172.22.36.11
20:00:00:0d:ec:02:1d:40  172.22.36.8
20:00:00:0c:85:e9:d2:c0  172.22.36.142
```

Total number of entries = 4



Note

The CFS protocol is enabled by default. However, most applications are not enabled by default to leverage CFS. You can later enable an application to use CFS for locking and synchronization.

What CFS Applications Do I Have and What Is Their Scope?

The **show cfs application** commands shows the Cisco applications using CFS.

```
172.22.36.9# show cfs application
```

```
-----
```

| Application | Enabled | Scope |
|---------------|---------|--------------|
| ivrr | Yes | Physical-fc |
| ntp | Yes | Physical-all |
| fscm | Yes | Physical-fc |
| islb | No | Physical-fc |
| role | Yes | Physical-all |
| rscn | No | Logical |
| radius | No | Physical-all |
| tacacs | No | Physical-all |
| fctimer | No | Physical-fc |
| syslogd | No | Physical-all |
| callhome | No | Physical-all |
| fcdomain | No | Logical |
| device-alias | Yes | Physical-fc |
| port-security | No | Logical |

Total number of entries = 14



Note

- Remember that a physical scope spans all switches physically connected together, regardless of VSAN configuration. Logical scope applies only to the VSAN for a configuration.
- The SCSI Flow Manager (SFM) monitors SCSI flows with the Storage Services Module (SSM).
- The Fabric Startup Configuration Manager (FSCM) enables the startup **copy running-config startup-config fabric** command.

Why Am I Locked Out of An Application by CFS?

CFS provides locking (physical or logical). If the lock is already in use, you see the error **Failed to acquire lock**.

```
172.22.36.9(config)# ntp peer 172.22.36.99
Failed to acquire Lock
```

To find out which user (on which switch) has the lock, enter the **show cfs lock** command.

```
172.22.36.9# show cfs lock
```

```
Application: ntp
Scope       : Physical
```

| Switch WWN | IP Address | User Name | User Type |
|-------------------------|---------------|-----------|-------------|
| 20:00:00:0c:85:e9:d2:c0 | 172.22.36.142 | admin | CLI/SNMP v3 |

Total number of entries = 1

Until the current user either commits changes to the database or their lock expires, you cannot modify the pending database unless you break the lock. The **clear ntp session** command clears the pending database and all pending changes for the specified application are lost.

```
172.22.36.9# clear ntp session
```

Command Scheduler

This section provides recipes for using the switch command scheduler.

Automated Switch Configuration Backup

Before SAN-OS 2.0, the only method for automated backup of a switch configuration was to set up a management station to periodically log into the switch and issue appropriate scripting commands to copy the configuration to a TFTP server. The drawback of that method is that, if the management station goes down, the configuration is not backed up. By enabling the MDS switch to “back itself up,” its configuration is pushed to a TFTP server. Additionally, in SAN-OS 3.0, the ability to timestamp a filename was provided so that new iterations of the script do not overwrite the previous backups.

Command Scheduler can now be used to regularly back up switch configuration to a TFTP server.

In this example, the following resources are used:

- Switch: 172.22.36.142
- TFTP Server: 171.71.58.69
- Schedule: “nightly_10pm” Every night at 10 PM.

To back up a switch configuration to a TFTP server, follow these steps:

Step 1 Enable the command scheduler with the **scheduler enable** command.

```
ca-9506# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
ca-9506(config)# scheduler enable
```

Step 2 Define the job to be run. Do this by saving the running configuration and then copying it to a TFTP server. The (config-job) prompt works the same as the switch exec-mode prompt. Therefore, any command on the switch can also be executed.

```
ca-9506(config)# scheduler job name backup_config
ca-9506(config-job)# copy running-config startup-config
ca-9506(config-job)# copy startup-config tftp://171.71.58.69/ca-9506_config_($TIMESTAMP)
```



Tip

The \$(TIMESTAMP) portion of the destination filename gets replaced with the date and time that the command was run. For example, the file could read ca-9506_config_2007-11-07-15.40.18 if the script was executed on November 7, 2007 at 15:40:18.

Step 3 Display the defined job with the **show scheduler** command.

```
ca-9506# show scheduler job name backup_config
Job Name: backup_config
-----
copy running-config startup-config
copy startup-config tftp://171.71.58.69/ca-9506/ca-9506_config_2007-11-07-15.40.18
=====
```

Step 4 Create the schedule. Assign the time (20:00) and the job that will be assigned to it (backup_config).

```
ca-9506# conf terminal
Enter configuration commands, one per line. End with CNTL/Z.
ca-9506(config)# scheduler schedule name nightly_10pm
ca-9506(config-schedule)# time daily 20:00
ca-9506(config-schedule)# job name backup_config
```

Step 5 Display the schedule with the **show scheduler** command.

```
ca-9506# show scheduler schedule name nightly_10pm
Schedule Name      : nightly_10pm
-----
```

```

User Name           : admin
Schedule Type       : Run every day at 20 Hrs 0 Mins
Last Execution Time : Yet to be executed

```

```

-----
Job Name           Last Execution Status
backup_config      n/a
-----

```

Step 6 After the job runs, examine the status of the job and the details of the execution with the **show scheduler** command.

```

ca-9506# show scheduler schedule name nightly_10pm
Schedule Name      : nightly_10pm

```

```

-----
User Name           : admin
Schedule Type       : Run every 0 Days 0 Hrs 1 Mins
Start Time          : Fri Apr 22 20:00:00 2005
Last Execution Time : Fri Apr 22 20:00:00 2005
Last Completion Time: Fri Apr 22 20:00:15 2005
Execution count     : 1

```

```

-----
Job Name           Last Execution Status
backup_config      Success (0)
-----

```

Detailed log:

```

ca-9506# show scheduler logfile

```

```

=====
Job Name           : backup_config           Job Status: Success (0)
Schedule Name      : nightly_10pm           User Name : admin
Completion time: Fri Apr 22 20:00:15 2005
----- Job Output -----

```

```

`copy running-config startup-config `
[####] 7%
[#####] 14%
[#####] 23%
[#####] 30%
[#####] 37%
[#####] 46%
[#####] 53%
[#####] 60%
[#####] 69%
[#####] 76%
[#####] 84%
[#####] 92%
[#####] 100%

```

```

`copy startup-config tftp://171.71.58.69/ca-9506_config_2007-11-07-15.40.18`
Trying to connect to tftp server.....

```

```

TFTP put operation was successful
=====

```

Copying Files to and from a Switch

You can move files to and from an MDS switch. These files can be log, configuration, or firmware files. There are two methods for copying files to and from the switch, using the command-line interface (CLI) and using Fabric Manager.

Copying Files Using the CLI

The CLI offers four protocols for copying files to or from the switch, FTP, SCP, SFTP, and TFTP. Because the switch always acts as a client, a session originates at the switch. The switch either pushes files to an external system or pulls files from an external system.

In this example, the following resources are used:

- File server: **172.22.36.10**
- File to be copied to the switch: **/etc/hosts**

The switch's **copy** command supports four transfer protocols and twelve different sources for files.

```
ca-9506# copy ?
 bootflash:      Select source filesystem
 core:           Select source filesystem
 debug:          Select source filesystem
 ftp:           Select source filesystem
 licenses        Backup license files
 log:            Select source filesystem
 modflash:       Select source filesystem
 nvram:          Select source filesystem
 running-config  Copy running configuration to destination
 scp:          Select source filesystem
 sftp:         Select source filesystem
 slot0:          Select source filesystem
 startup-config  Copy startup configuration to destination
 system:         Select source filesystem
 tftp:         Select source filesystem
 volatile:       Select source filesystem
```

Secure Copy Protocol

Secure copy protocol (SCP) transfers use this syntax:

scp:[//[username@]server][/path]

To copy the file /etc/hosts from the server 172.22.36.10 to the switch destination file hosts.txt (using the user user1) enter:

```
switch# copy scp://user1@172.22.36.10/etc/hosts bootflash:hosts.txt
user1@172.22.36.10's password:
hosts                               100% |*****| 2035    00:00
```

Secure File Transfer Protocol

To back up the switch start up configuration to a Secure File Transfer Protocol (SFTP) server, enter:

```
switch# copy startup-config sftp://user1@172.22.36.10/MDS/startup-configuration.bak1
Connecting to 172.22.36.10...
User1@172.22.36.10's password:
switch#
```



Tip

Backing up the startup-configuration to a server should be done on a daily basis and before any changes. A short script can be written to be run on the switch to save, then back up, the configuration. The script needs to contain only two commands: **copy running-configuration startup-configuration** and **copy startup-configuration tftp://server/name_\$(TIMESTAMP)**. To execute the script use the **run-script filename** command.

Managing Files on the Standby Supervisor

To copy to or from a file, or to delete a file from the supervisor, follow these steps:

- Attach to the standby supervisor.
- Use the **dir** and **delete** commands.



Note

This recipe is used when a firmware upgrade fails because there is not enough free bootflash capacity on the standby supervisor for the firmware images.

Delete a File from the Standby Supervisor

To delete a file from the standby supervisor, follow these steps:

- Step 1** Determine which supervisor is the standby with the **show module** command. In this example, the standby is module 6.

```
switch# show module
```

| Mod | Ports | Module-Type | Model | Status |
|-----|-------|----------------------------|-----------------|------------|
| 1 | 16 | 1/2 Gbps FC Module | DS-X9016 | ok |
| 2 | 16 | 1/2 Gbps FC Module | DS-X9016 | ok |
| 3 | 8 | IP Storage Services Module | DS-X9308-SMIP | ok |
| 4 | 0 | Caching Services Module | DS-X9560-SMAP | ok |
| 5 | 0 | Supervisor/Fabric-1 | DS-X9530-SF1-K9 | active * |
| 6 | 0 | Supervisor/Fabric-1 | DS-X9530-SF1-K9 | ha-standby |

- Step 2** Connect to the standby supervisor using the **attach module** command. The prompt now displays the word “standby.”

```
ca-9506# attach module 6
Attaching to module 6 ...
To exit type 'exit', to abort type '$.'
Cisco Storage Area Networking Operating System (SAN-OS) Software
TAC support: http://www.cisco.com/tac
```

Copyright (c) 2002-2004, Cisco Systems, Inc. All rights reserved.
The copyrights to certain works contained herein are owned by
Andiamo Systems, Inc. and/or other third parties and are used and
distributed under license. Some parts of this software are covered
under the GNU Public License. A copy of the license is available
at <http://www.gnu.org/licenses/gpl.html>.
ca-9506(standby)#

Step 3 List the files on the boot flash with the **dir** command.

```
ca-9506(standby)# dir bootflash:
12330496 Jun 30 21:11:33 2004 boot-1-3-4a
2035 Jun 17 16:30:18 2004 hosts.txt
43705437 Jun 30 21:11:58 2004 isan-1-3-4a
12288 Dec 31 17:13:48 1979 lost+found/
12334592 Jun 23 17:02:16 2004 m9500-sflek9-kickstart-mz.1.3.4b.bin
43687917 Jun 23 17:02:42 2004 m9500-sflek9-mz.1.3.4b.bin
99 Apr 07 19:28:54 1980 security_cnv.log

Usage for bootflash://sup-local
126340096 bytes used
59745280 bytes free
186085376 bytes total
```

Step 4 Delete the file with the **delete** command.

```
ca-9506(standby)# delete bootflash:hosts.txt
```

Step 5 Enter the **exit** command, and the prompt returns to the active supervisor prompt:

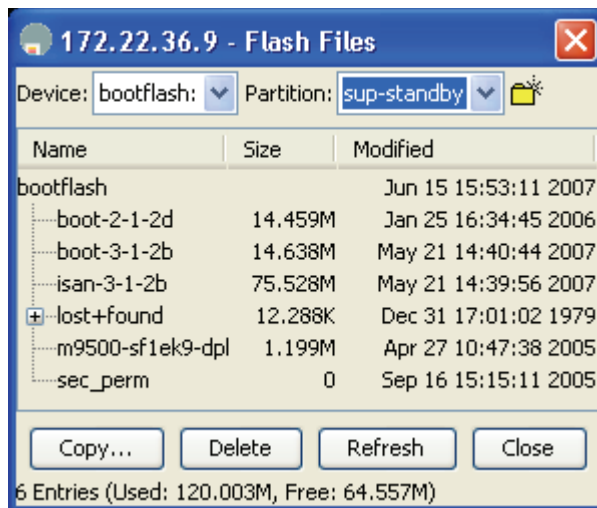
```
ca-9506(standby)# exit
rlogin: connection closed.
ca-9506#
```

Deleting a File using Device Manager

To delete a file using Device Manager, follow these steps:

-
- Step 1** Select the **Admin** menu.
 - Step 2** Select **Flash Files**.
 - Step 3** Select the supervisor that you want to delete the file from. It can be either sup-local or sup-standby. See [Figure 1-3](#).

Figure 1-3 Deleting a File with Device Manager



Step 4 Select the file to be deleted.

Step 5 Click **Delete**.

Firmware Upgrades and Downgrades

Upgrading has not changed from SAN-OS 1.x to SAN-OS 2.x. However, downgrading from SAN-OS 2.x to 1.x requires special attention.

Upgrading Firmware with the CLI

You can upgrade to Cisco SAN-OS 3.x using either the **install all** command or the Firmware Upgrade wizard in Fabric Manager.



Tip

- Always carefully read the output of the compatibility check of the **install all** command. This tells you exactly what needs to be upgraded (BIOS, loader, firmware) and what modules are not hitless. If there are any questions or concerns about the results of the output, select **n** to stop the installation and contact the next level of support.
- Verify before starting the download that there is sufficient space on the bootflash of both supervisors.
- Verify that an Ethernet cable is plugged into the standby supervisor, as that will become the new active supervisor after the upgrade is complete.

The following example below demonstrates upgrading from SAN-OS 3.1(2b) to 3.1(3) using the **install all** command with the source images located on a SCP server:

Upgrade firmware from SAN-OS 3.1(2b) to 3.1(3) using the **install all** command.


```
172.22.36.9# install all system scp://testuser@dcbu-dev1/tftpboot/rel/isan-3-1-3 kickstart
scp://testuser@dcbu-dev1/tftpboot/boot-3-1-3
For scp://testuser@dcbu-dev1, please enter password:
For scp://testuser@dcbu-dev1, please enter password:
```

```
Copying image from scp://testuser@dcbu-dev1/tftpboot/boot-3-1-3 to
bootflash:///boot-3-1-3.
```

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

```
Copying image from scp://testuser@dcbu-dev1/tftpboot/isan-3-1-3 to
bootflash:///isan-3-1-3.
```

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

```
Verifying image bootflash:///boot-3-1-3 for boot variable "kickstart".
```

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

```
Verifying image bootflash:///isan-3-1-3 for boot variable "system".
```

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

```
Extracting "slc" version from image bootflash:///isan-3-1-3.
```

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

```
Extracting "bios" version from image bootflash:///isan-3-1-3.
```

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

```
Extracting "system" version from image bootflash:///isan-3-1-3.
```

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

```
Extracting "kickstart" version from image bootflash:///boot-3-1-3.
```

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

```
Extracting "loader" version from image bootflash:///boot-3-1-3.
```

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

Compatibility check is done:

| Module | bootable | Impact | Install-type | Reason |
|--------|----------|----------------|--------------|--------|
| 2 | yes | non-disruptive | rolling | |
| 3 | yes | non-disruptive | rolling | |
| 5 | yes | non-disruptive | reset | |
| 6 | yes | non-disruptive | reset | |

Images will be upgraded according to following table:

| Module | Image | Running-Version(pri:alt) | New-Version | Upg-Required |
|--------|-----------|--------------------------|------------------|--------------|
| 2 | slc | 3.1(2b) | 3.1(3) | yes |
| 2 | bios | v1.1.0(10/24/03) | v1.1.0(10/24/03) | no |
| 3 | slc | 3.1(2b) | 3.1(3) | yes |
| 3 | bios | v1.1.0(10/24/03) | v1.1.0(10/24/03) | no |
| 5 | system | 3.1(2b) | 3.1(3) | yes |
| 5 | kickstart | 3.1(2b) | 3.1(3) | yes |
| 5 | bios | v1.1.0(10/24/03) | v1.1.0(10/24/03) | no |
| 5 | loader | 1.2(2) | 1.2(2) | no |
| 6 | system | 3.1(2b) | 3.1(3) | yes |
| 6 | kickstart | 3.1(2b) | 3.1(3) | yes |
| 6 | bios | v1.1.0(10/24/03) | v1.1.0(10/24/03) | no |
| 6 | loader | 1.2(2) | 1.2(2) | no |

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

```
Install is in progress, please wait.
```

```
Syncing image bootflash:///boot-3-1-3 to standby.  
[#####] 100% -- SUCCESS
```

```
Syncing image bootflash:///isan-3-1-3 to standby.  
[#####] 100% -- SUCCESS
```

```
Setting boot variables.  
[#####] 100% -- SUCCESS
```

```
Performing configuration copy.  
[#####] 100% -- SUCCESS
```

```
Module 6: Waiting for module online.  
-- SUCCESS
```

```
"Switching over onto standby".
```

To watch the progress of the installation from the new active supervisor, reconnect to the switch and use the **show install all status** command.

```
ca-9506# show install all status  
There is an on-going installation...  
Enter Ctrl-C to go back to the prompt.
```

```
Continue on installation process, please wait.  
The login will be disabled until the installation is completed.  
Trying to start the installer...
```

```
Module 5: Waiting for module online.  
-- SUCCESS
```

```
Module 1: Non-disruptive upgrading.  
-- SUCCESS
```



Note

The installation continues first with the new standby, and then with the modules. If any module fails to upgrade, the entire process stops. You should then contact your next level of support.

Downgrading Firmware with the CLI

Before downgrading firmware, you must turn off or disable any features that are not supported by the older version (see Steps 1-3). Failure to do so can disrupt the downgrade.

To downgrade firmware on the switch from the CLI, follow these steps:

Step 1

Verify there are no features enabled that are not supported in the lower level firmware using the **show incompatibility** command. Always run this command before a downgrade, even if no SAN-OS Release 2.x features were explicitly enabled.



Caution

Failure to disable a feature listed in the incompatibility check can result in a disruptive firmware downgrade.

When downgrading from 3.1(x) to 2.0(x) (because IVR with FC NAT was not available in SAN-OS 2.0(x)), there are several possible errors that you might see:

```
ca-9506# show incompatibility system bootflash:m9500-sf1ek9-mz.2.0.2b.bin
The following configurations on active are incompatible with the system image
1) Service : ivr , Capability : CAP_FEATURE_IVR_FCID_NAT_ENABLED
Description : ivr fcid-nat mode is enabled
Capability requirement : STRICT

2) Service : ivr , Capability : CAP_FEATURE_IVR_AUTO_VSAN_TOPOLOGY_ENABLED
Description : ivr auto vsan-topology mode is enabled
Capability requirement : STRICT
```

Step 2 Disable unsupported features using the configuration commands:

```
ca-9506#
ca-9506# conf t
Enter configuration commands, one per line. End with CNTL/Z.
ca-9506(config)# no device-alias distribute
```

Step 3 Rerun the incompatibility check to verify that all unsupported features are gone:

```
ca-9506# show incompatibility system bootflash:m9500-sf1ek9-mz.1.3.5.bin
No incompatible configurations
```

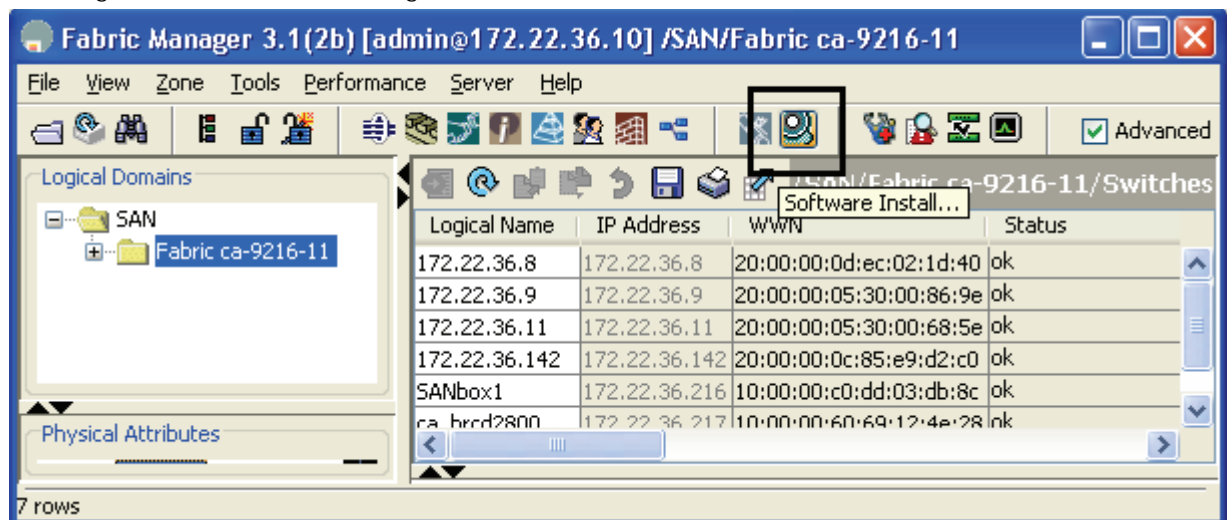
Step 4 Proceed with the downgrade using the **install all** command as described in [Upgrading Firmware with the CLI, page 1-20](#).

Upgrading Firmware with Fabric Manager

To upgrade the firmware of one or more MDS switches with Fabric Manager, follow these steps:

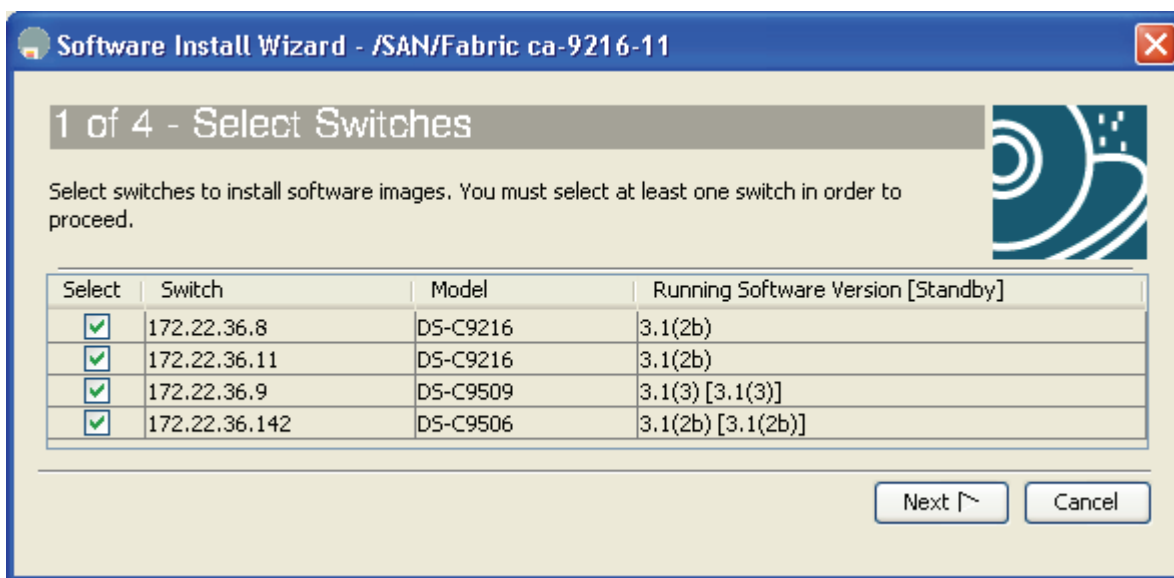
Step 1 Click the **Software Install Wizard** from the tool bar in Fabric Manager.

Figure 1-4 Fabric Manager Software Wizard



Step 2 Select the switches to upgrade and click **Next** (see [Figure 1-5](#)).

Figure 1-5 Select Switches to Upgrade



- Step 3** Specify the location of the firmware images (see [Figure 1-6](#)).
- Provide the file information to transfer the file from the server to the switch. If the files are to be downloaded during the install, provide the path and filename of the images as well.
 - Select **Skip Image Download** to upgrade using images already located on the supervisor's boot flash. L

Figure 1-6 Specify Firmware Images

2 of 4 - Specify Software Image(s) by Model

For each switch model, specify the new images to use. You must specify at least one image for each model by double-clicking on the table cell. The total space required on the bootflash to copy the image is shown in the 'Required Flash Space' column. To use images that are already downloaded, check 'Skip Image Download'. Press 'Verify' to validate remote server settings and filenames (SSHv2 only); be patient, this can take awhile. Please manually copy ssi image to switch.

Transfer files from: ☐ Local FM TFTP ☒ Remote

Remote Options

Copy Files Via: ☐ TFTP ☐ SFTP ☒ SCP ☐ FTP

Server:

UserName:

Password:

Flash Space: 1..512 MB

- Image(s)

| Model | System | Kickstart | Asm-sfn | Ssi |
|----------|------------|------------|---------|-----|
| DS-C9200 | isan-3-1-3 | boot-3-1-3 | | |

☐ Skip Image Download

Step 4 Click Next.

Depending on the installation method (already downloaded to bootflash, or download during the install), the wizard may ask for additional file locations. The fourth and final screen provides a summary and lets you start the actual installation.

During installation, a compatibility screen popup displays the same version compatibility information that was displayed during the CLI upgrade. Click **Yes** to continue with the upgrade.

**Note**

Unlike a CLI upgrade, Fabric Manager maintains a connection to the switch and provides detailed upgrade information. You do not have to manually reestablish connectivity to the switch during the supervisor switchover. If there is a failure, the last screen displays the reasons for a failed upgrade.

Password Recovery

If an admin password is lost and there are no other accounts on the switch with either network-admin or user account creation privileges, you need to recover the password for the admin account.

**Caution**

This procedure requires console access to the switch and requires a reboot of the switch.

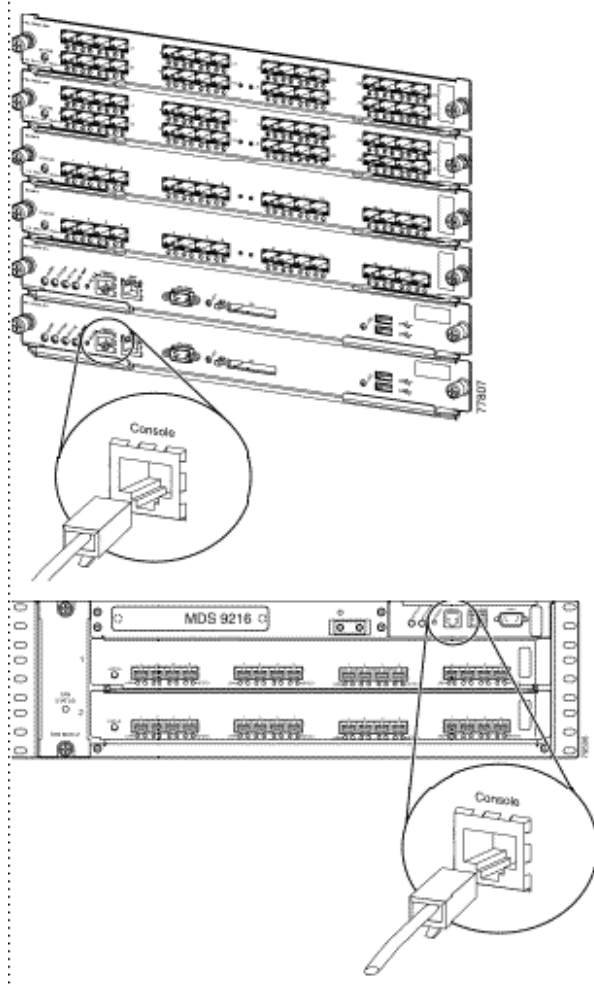
**Tip**

Another CLI user with network-admin privileges can change the password of the admin user without reloading the switch.

To recover the admin password on the switch, follow these steps:

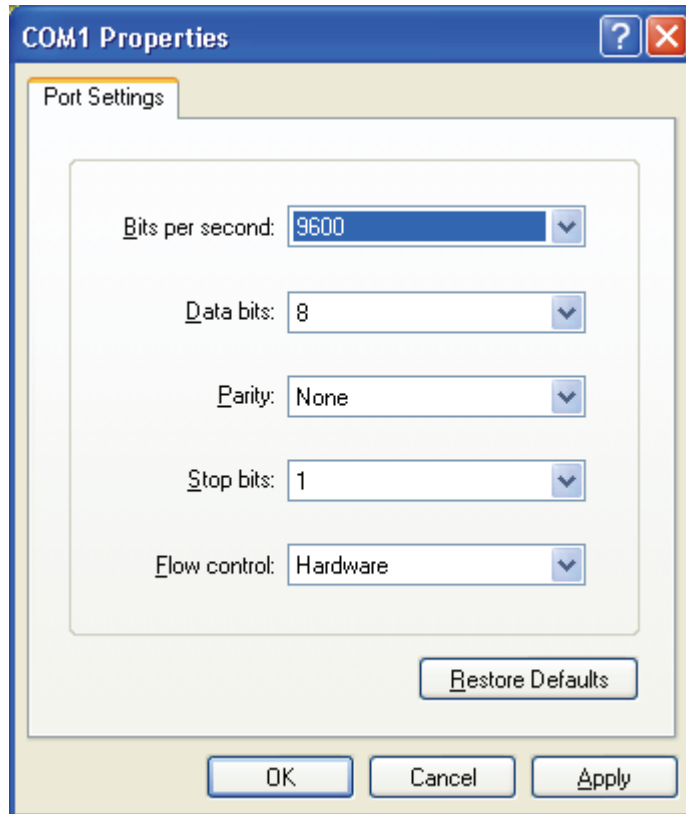
Step 1 Connect a console cable to the active supervisor of the MDS switch:

Figure 1-7 Console Connection on 9500 and 9200 Series MDS switches.



- Step 2** Attach the RS-232 end of the console cable to a PC.
- Step 3** Configure HyperTerm or similar terminal emulation software for 9600 baud, 8 data bits, no parity, 1 stop bit, and no flow control as shown in Figure 1-8.

Figure 1-8 HyperTerm Terminal Settings.



- Step 4** Establish a connection to the switch if possible, or establish enough connection to display the login prompt if no user accounts are available.
- Step 5** For a multisupervisor switch, MDS-9509 switch, or MDS-9506 switch, physically remove the standby supervisor. It is not necessary to remove it from the chassis, just loosen it until it does not make contact with the backplane.
- Step 6** Reboot the switch either by cycling the power or entering the **reload** command from the PC hyper terminal.
- Step 7** Press the **Ctrl-]** key sequence (when the switch begins its SAN-OS software boot sequence) to switch to the switch(boot)# prompt.
- Step 8** Enter configuration mode.
- ```
switchboot# config terminal
```
- Step 9** Issue the **admin-password new password** command.

```
switch(boot-config)# admin-password temppassword
switch(boot-config)# exit
```

**Step 10** Load the system image to finish the boot sequence.

```
switch(boot)# load bootflash: m9500-sf1ek9-mz.3.1.2b.bin
```

- Step 11** Log on to the switch using the admin account and the temporary password.

```
switch login: admin
Password:
Cisco Storage Area Networking Operating System (SAN-OS) Software
TAC support: http://www.cisco.com/tac
Copyright (c) 2002-2004, Cisco Systems, Inc. All rights reserved.
The copyrights to certain works contained herein are owned by
Andiamo Systems, Inc. and/or other third parties and are used and
distributed under license. Some parts of this software are covered
under the GNU Public License. A copy of the license is available
at http://www.gnu.org/licenses/gpl.html.
switch#
```

- Step 12** Change the admin password to a new permanent password.

```
ca-9506# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
ca-9506(config)# username admin password g05ox
```

- Step 13** Save the configuration that includes the new password.

```
switch# copy running-config startup-config
[#####] 100%
```

## Installing a License

To run an MDS 9000 Family switch, a license key is required after 120 days. The 120-day grace period allows you to try out new features or resume operation if a replacement chassis needs a license key installed.



### Caution

If a license grace period expires, all features that depend on that license are disabled even if they are currently running or in production.

You can install a license key using one of two methods, either from the CLI or using Fabric Manager's License Installation Wizard. These examples use the switch 172.22.36.10.

## Using the CLI to Install a License

To install a license from the CLI, follow these steps:

- Step 1** Copy the license file to the boot flash of the supervisor.

```
switch# copy scp://user1@172.22.36.10/tmp/FM_Server.lic bootflash:FM_Server.lic
user1@172.22.36.10's password:
FM_Server.lic 100% |*****| 2035 00:00
```

- Step 2** Verify the license file with the **show license file** command.

```
switch# show file bootflash:FM_Server.lic
FM_Server.lic:
SERVER this_host ANY
```



```
VENDOR cisco
INCREMENT FM_SERVER_PKG cisco 1.0 permanent uncounted \
 VENDOR_STRING=MDS HOSTID=VDH=FOX0713037X \
 NOTICE="<LicFileID>lic_template</LicFileID><LicLineID>0</LicLineID> \
 <PAK>dummyPak</PAK>" SIGN=D8CF07EA26C2
```

**Step 3** Cross-reference the switch's host ID (VDH=FOX0713037X) with the one listed in the license file.

```
ca-9506# show license host-id
License hostid: VDH=FOX0713037X
```

**Step 4** Install the license file.

```
switch# install license bootflash:FM_Server.lic
Installing license ..done
```

**Step 5** Verify that the license has been installed with the **show license** command.

```
switch# show license
FM_Server.lic:
SERVER this_host ANY
VENDOR cisco
INCREMENT FM_SERVER_PKG cisco 1.0 permanent uncounted \
 VENDOR_STRING=MDS HOSTID=VDH=FOX0713037X \
 NOTICE="<LicFileID>lic_template</LicFileID><LicLineID>0</LicLineID> \
 <PAK>dummyPak</PAK>" SIGN=D8CF07EA26C2
```

To display a summary of the installed licenses, use the **show license usage** command:

```
switch# show license usage
```

| Feature                      | Ins | Lic   | Status | Expiry | Date | Comments |
|------------------------------|-----|-------|--------|--------|------|----------|
|                              |     | Count |        |        |      |          |
| FM_SERVER_PKG                | Yes | -     | In use | never  | -    | -        |
| MAINFRAME_PKG                | Yes | -     | Unused | never  | -    | -        |
| ENTERPRISE_PKG               | Yes | -     | In use | never  | -    | -        |
| SAN_EXTN_OVER_IP             | Yes | 2     | In use | never  | -    | -        |
| PORT_ACTIVATION_PKG          | No  | 0     | Unused |        | -    | -        |
| SAN_EXTN_OVER_IP_18_4        | No  | 0     | Unused |        | -    | -        |
| SAN_EXTN_OVER_IP_IPS2        | No  | 0     | Unused |        | -    | -        |
| SAN_EXTN_OVER_IP_IPS4        | No  | 0     | Unused |        | -    | -        |
| 10G_PORT_ACTIVATION_PKG      | No  | 0     | Unused |        | -    | -        |
| STORAGE_SERVICES_ENABLER_PKG | No  | 0     | Unused |        | -    | -        |

To determine which features within a license package are being used, specify the package name. In this case, QoS is using the Enterprise package:

```
ca-9506# show license usage ENTERPRISE_PKG
Application

Qos Manager

```

## Using Fabric Manager to Install a License

To install the licenses with the Fabric Manager License Wizard, follow these steps:

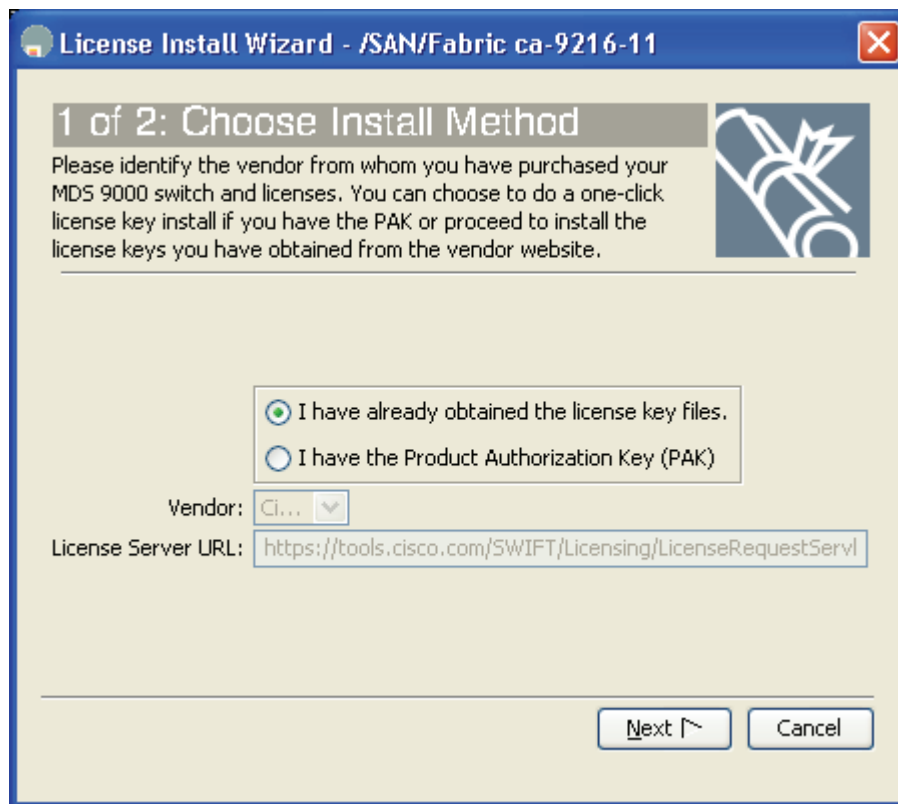
**Step 1** Click the **License Install** icon shown in [Figure 1-9](#) to launch the License Installation Wizard.

Figure 1-9 Launching the License Installation Wizard



- Step 2** Indicate whether you already have license key files or if you have only a Product Authorization Key (PAK) at this time. If you already have the files, you will be asked to indicate their location. If you have a PAK, then the license files will be downloaded and installed from Cisco's website. Click **Next**.

Figure 1-10 Choose License Installation Method



- a. If you indicated that you have the license key files, then you are asked to specify the name and location of the license key files, in Figure 1-11.

Figure 1-11 License File Location

**2 of 2: Install License File**

Enter the license file location for the selected switches. If necessary, the file will be transferred to the switch bootflash prior to install.

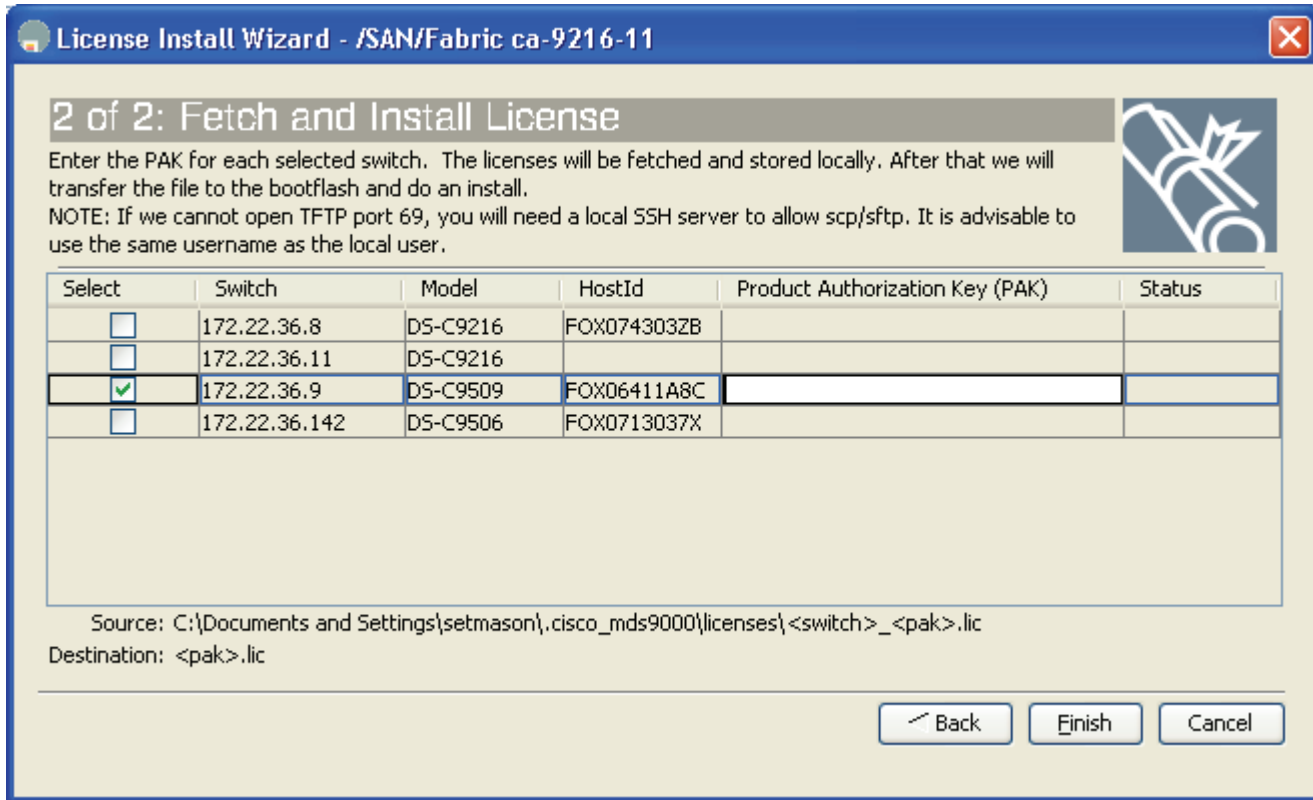
NOTE: If we cannot open TFTP port 69, you will need a local SSH server to allow scp/sftp. It is advisable to use the same username as the local user.

| Select                              | Switch        | Model    | Host ID     | On Bootflash?            | License File Name | Status          |
|-------------------------------------|---------------|----------|-------------|--------------------------|-------------------|-----------------|
| <input type="checkbox"/>            | 172.22.36.8   | D5-C9216 | FOX074303ZB | <input type="checkbox"/> |                   |                 |
| <input type="checkbox"/>            | 172.22.36.11  | D5-C9216 |             | <input type="checkbox"/> |                   | Error - no h... |
| <input checked="" type="checkbox"/> | 172.22.36.9   | D5-C9509 | FOX06411A8C | <input type="checkbox"/> | FMSERVER.lic      |                 |
| <input type="checkbox"/>            | 172.22.36.142 | D5-C9506 | FOX0713037X | <input type="checkbox"/> |                   |                 |

Back Finish Cancel

- b. If you indicated that you have only the PAK numbers, Fabric Manager will obtain the license files directly from Cisco.com. When you see the screen shown in [Figure 1-12](#), provide your PAK.

Figure 1-12 Install License Using PAK



**Step 3** Click **Finish** to complete license installation.

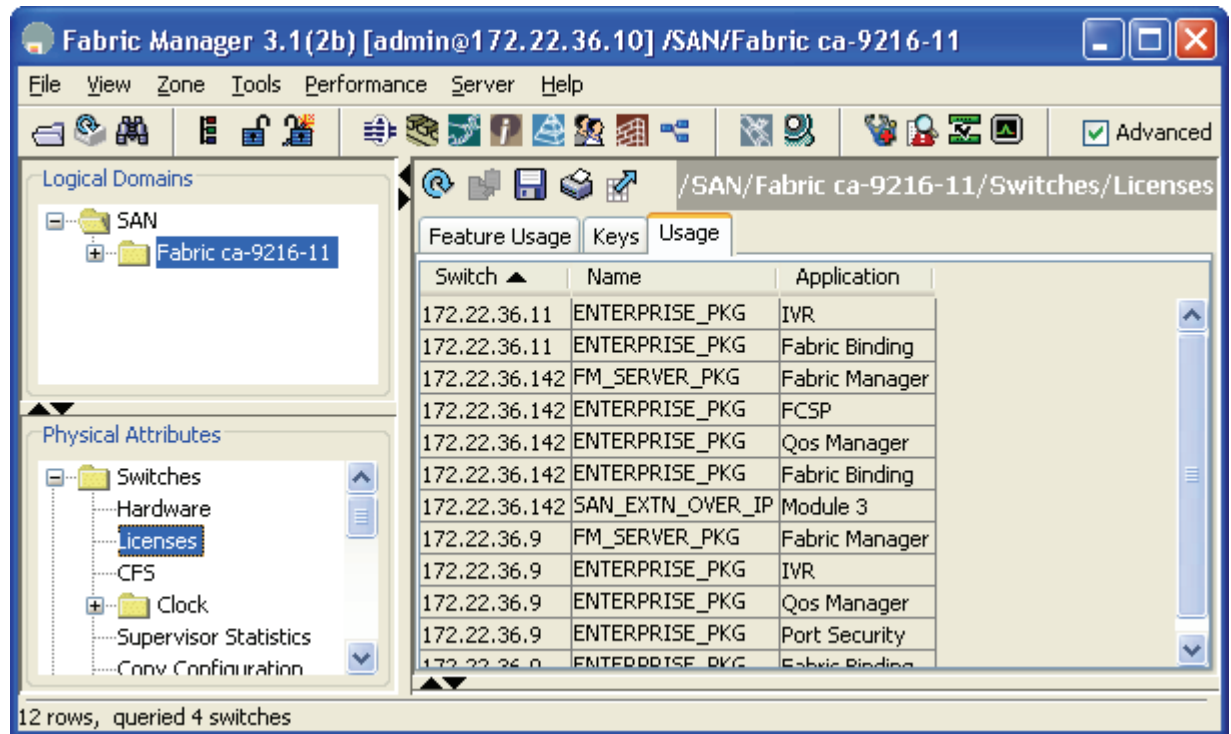
## Which Feature Enables the License Grace Period?

When you enable a feature that is licensed but not covered by your current license, you receive a syslog message, an e-mail or other warnings that the feature, such as Fabric Manager Server, has entered its grace period. To determine what features have enabled a specific license, use either the CLI or Fabric Manager.

## Check License Usage With Fabric Manager

In the Physical Attributes Pane, click **Switches** then **Licenses** to see which feature triggered the warning. In the resulting middle pane, click the **Usage** tab. See [Figure 1-13](#).

Figure 1-13 License Usage



## Check with the CLI

With the CLI, use the **show license usage *license name*** command to see which feature triggered the warning.

```
switch1# show license usage ENTERPRISE_PKG
Application

SFM
Qos Manager
Port Security

```

Once the feature has been identified, either disable it or install a new license.

# Copying Core Files from the Switch

If a switch process crashes, it may create a core file to send to Cisco TAC for further troubleshooting. This procedure explains how to retrieve this core file from the MDS switch.

The resource used in this procedure is FTP server: 172.22.36.10



## Note

Use any of the possible methods to copy, including FTP, TFTP, SFTP, and SCP.

To copy a core file from an MDS switch, follow these steps:

- Step 1** Before copying a core file to another server, identify the PID of the core file with the **show cores** command.

```
switch# show cores
Module-num Process-name PID Core-create-time

5 fspf 1524 Sep 27 03:11
```

- Step 2** Copy the core file (this example uses FTP) with the **copy core** command.

```
"core://<module-number>/<process-id>"
```

```
switch# copy core://5/1524 ftp://172.22.36.10/tmp/fspfcore
```

Send the file to Cisco TAC following the TAC engineer's directions.

# Restoring a Fixed Switch Configuration

This procedure describes backing up and restoring a switch configuration for one of the MDS 9000 Family fixed configuration switches, such as a 9100 or 9200 series switch. Parts of this procedure are disruptive and should only be done during an emergency, such as the chassis or fixed supervisor needing replacement.

This procedure uses the following resources:

- Old switch: switch1: (172.22.36.8)
- New switch: switch2
- File server: host1



## Note

Restore a switch configuration only to a switch with the same firmware version used to create the switch configuration. If an upgrade is required, first restore the configuration then upgrade the firmware.

To restore a switch configuration, follow these steps:

- Step 1** Save the running configuration with the **copy running-config** command.

```
switch1# copy running-config startup-config
[#####] 100%
```

- Step 2** Copy the start up-configuration to the file server using any of the available methods on the switch (FTP, TFTP, SFTP, SCP).

```
switch1# copy startup-config scp://user@host1/switch1.config
user@switch1's password:
sysmgr_system.cfg 100% |*****| 10938 00:00
switch1#
```

- Step 3** Capture the port assignments using the FLOGI database. This will be used to verify that all the cables are placed in their correct locations.

```
switch1# show flogi database

INTERFACE VSAN FCID PORT NAME NODE NAME

fc1/8 600 0x7c0007 50:05:07:63:00:ce:a2:27 50:05:07:63:00:c0:a2:27
fc1/13 1001 0xef0001 50:06:0e:80:03:4e:95:13 50:06:0e:80:03:4e:95:13
fc1/15 600 0x7c0004 50:06:0b:00:00:13:37:ae 50:06:0b:00:00:13:37:af
```

**Note**

At this point the old switch is no longer needed, so you can disconnect its mgmt0 port from the LAN.

- Step 4** Log on to the new switch using the console connection and clear the switch configuration. Do not run the setup script, if prompted. The **write erase** command erases the switch's configuration.

```
switch2# write erase
Warning: This command will erase the startup-configuration.
Do you wish to proceed anyway? (y/n) [n] y
```

- Step 5** Reload the switch.

```
switch2# reload
This command will reboot the system. (y/n)? [n] y
```

- Step 6** The switch comes up in factory default mode and prompts for basic system configuration. You can ignore the prompt by pressing CRL+Z, because all the configuration options are contained in the old switch's start up configuration. Manually configure the IP address.

```
switch2# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch2(config)# int mgmt 0
switch2(config-if)# ip address 172.22.36.8 255.255.254.0
switch2(config-if)# no shut
```

- Step 7** If interface (fc X/Y)-based zoning was done, obtain the WWN of the new switch with the **show wwn switch** command. Otherwise, skip this step.

```
switch2# show wwn switch
Switch WWN is 20:00:00:0d:ec:02:1d:40
```

- Step 8** On the file server, make a copy of the configuration file, then open the copy in a text editor such as Notepad. Make these changes:

- a. Remove the lines that contain the SNMP user accounts, as the encrypted passwords are tied to the MAC address of the chassis.:

```
$ cp switch1.config switch1.config.orig
$ vi switch1.config
```

The user accounts are all grouped together and begin with **snmp-server user** command.

```
snmp-server user admin network-admin auth md5 0x46694cac2585d39d3bc00c8a4c7d48a6
localizedkey
snmp-server user guestadmin network-admin auth md5 0xcae40d254218747bc57ee1df348
```

```
26b51 localizedkey
```

- b. If interface (fc X/Y)-based zoning was done, replace the WWN of the old switch in the zone member commands with the WWN of the new switch. Otherwise, skip this step.

```
zone name Z_1 vsan 9
 member interface fcl/9 swwn 20:00:00:0d:ec:02:1d:40
```

- c. If IVR was configured on this switch, the IVR topology will need to be modified as that is based upon the sWWN and the old sWWN should be replaced with the new sWWN.

```
ivr vsan-topology database
 autonomous-fabric-id 1 switch-wwn 20:00:00:0d:ec:02:1d:40 vsan-ranges 500,3002
 autonomous-fabric-id 1 switch-wwn 20:00:00:0c:85:e9:d2:c0 vsan-ranges 500,3000
```

If the IVR topology is configured for auto and is distributed through CFS, then this step does not need to be done as the switch will learn of the topology via CFS.



**Note** If there are multiple IVR-enabled switches in the fabric, the sWWN from the old switch should be removed from ALL of the IVR topologies in the fabric and replaced with the new sWWN. This step should be done before bringing the new switch online. These modifications can be done on the other switches using either the CLI or Fabric Manager.

- d. Save and exit the configuration file.

- Step 9** From the new switch, copy the modified config file from the file server to the running configuration of the new switch. As the file is copied, it is executed and the configuration is applied. The commands being applied are displayed in single quotes. Any errors resulting from the commands are displayed immediately after the command that caused it. When finished, the prompt changes to reflect the new switch name.

```
switch2# copy scp://user@host1/switch1.config running-config
user@host1's password:
switch1.config 100% |*****| 10938 00:00
```

- Step 10** Save the configuration by copying the startup configuration to the running configuration.

```
switch1# copy running-config startup-config
[*****] 100%:
```

- Step 11** The switch can now be accessed with the CLI. Complete the configuration restoration:

- Recreate SNMP user accounts.
- If the switch is accessed with SSH, remove the MDS switch entry from the host's known\_hosts file because the switch's public key has changed.
- Install any required license keys.

- Step 12** Move the cables from the old switch to the new switch, using the **show flogi database** command output on the old switch as a reference to verify that each cable is in the correct location.

- Step 13** Verify that all devices have logged in and all features are running as they are supposed to be and save the running configuration to the start up configuration with the **copy running-config startup-config** command.

- Step 14** Reload the switch to verify that it boots correctly with the configuration.



# Configuring an NTP Server

Network Time Protocol (NTP) is a protocol used by devices to synchronize their internal clocks with other devices. The switch can only be used as an NTP client and can talk to other NTP systems with a higher stratum (or authority). NTP is hierarchical in nature, so that lower stratum numbers are closer to the source of the time authority. Devices that are at the same stratum can be configured as peers so that they work together to determine the correct time by making minute adjustments. Normally, MDS switches are configured as peers, while a router or other dedicated machine is used as an NTP server.

## Configuring NTP with CFS

CFS (see “[Cisco Fabric Services, page 1-10](#)”) lets you perform a single configuration for NTP and have it propagated to other switches. Also, if a new switch comes online it can be set to inherit the NTP configuration from the existing switches. The new switch merges its configuration (no configuration) with the existing NTP CFS configuration and the result is that the new switch has an NTP configuration.



### Note

NTP does not set the time zone (or offset from UTC) for the switch; it must be set manually, as follows. This example uses Eastern Standard Time and Eastern Daylight-Savings Time:

**clock timezone EST -5.0**

**clock summer-time EDT 1 Sunday Apr 02:00 5 Sunday Oct 02:00 60**

If your fabric spans multiple sites, do not use CFS for an NTP configuration as all MDS switches will end up using the same NTP servers.

For this example, these resources are used:

- Switch #1 IP address: 172.22.36.142
- Switch #2 IP address: 172.22.36.8
- NTP server: 171.69.16.26

To configure NTP for switch 1, follow these steps:

**Step 1** Enter configuration mode and enable CFS distribution for NTP for switches 1 and 2.

a. For switch 1:

```
switch1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
switch1(config)# ntp distribute
```

b. For switch 2

```
switch2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
switch2(config)# ntp distribute
```



### Note

Steps 2 through 5 can be done from configuration mode of either switch 1 or switch 2.

**Step 2** Change to configuration mode with the **conf t** command, and then add the NTP server to the configuration.

```
switch1# conf t
```

```
switch1(config)# ntp server 171.69.16.26
```

**Step 3** Add the NTP peer switches to the configuration.

```
switch1(config)# ntp peer 172.22.36.8
switch1(config)# ntp peer 172.22.36.142
```

**Step 4** Commit the NTP configuration and end configuration mode.

```
switch1(config)# ntp commit
switch1(config)# end
```

At this point, NTP is configured and the switch will slowly adjust to the new time.

To view the NTP configuration on the local switch, use the **show ntp peers** command.

```
switch1# show ntp peers
```

```

Peer IP Address Serv/Peer

172.22.36.142 Peer
171.69.16.26 Server
172.22.36.8 Peer
```

To view the NTP configuration on the remote switch, use the **show ntp peers** command.:

```
switch2# show ntp peers
```

```

Peer IP Address Serv/Peer

172.22.36.142 Peer
171.69.16.26 Server
172.22.36.8 Peer
```

**Step 5** Save the configuration on both switches. CFS can be used to instruct both switches to save their configuration by running the **copy running-config startup-config** command. See [Saving the Configuration Across the Fabric, page 1-41](#).

```
switch1# copy running-config startup-config fabric
[#####] 100%
```

## Configure NTP without CFS

To configure NTP without CFS, log on to each switch in the fabric and configure NTP. This is the same procedure used in a SAN-OS 1.x environment.



### Note

NTP does not set the time zone (or offset from UTC) for the switch. You must set it manually. This example uses Eastern Standard Time and Eastern Daylight-Savings Time:

```
clock timezone EST -5.0
```

```
clock summer-time EDT 1 Sunday Apr 02:00 5 Sunday Oct 02:00 60
```

For this example, these resources are used:

- Switch #1 IP address: 172.22.36.142
- Switch #2 IP address: 172.22.36.9

- NTP server: 171.69.16.26

To configure NTP for switch1, follow these steps:

**Step 1** Change to configuration mode with the **conf t** command, and then add the NTP server.

```
switch1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
switch1(config)# ntp server 171.69.16.26
```

**Step 2** Add the NTP peer switch, and then end configuration mode.

```
switch1(config)# ntp peer 172.22.36.9
switch1(config)# end
```

At this point, NTP is configured and the switch will slowly adjust to the new time.

To view the NTP configuration, use the **show ntp peers** command:

```
switch1# show ntp peers

Peer IP Address Serv/Peer

171.69.16.26 Server
172.22.36.9 Peer

```

## What to Do Before Calling TAC

When you need to contact the Cisco TAC or an OSM for additional assistance, follow these steps: This will reduce the amount of time needed to resolve the issue.



### Caution

**Do not reload the line card or switch until at least Step 1.** Some logs and counters are kept in volatile storage and will not survive a reload.

**Step 1** Collect switch information and the switch configuration both before and after the issue has been resolved. These three collection methods have the same result:

- CLI method:
  - Configure the Telnet/SSH application to log the screen output to a text file and enter the **show tech-support details** command.
  - Alternatively, there are feature-specific versions of the **show tech-support** command details in the syntax, such as the **show tech-support feature** command.
  - Enter the **tac-pac filename** command, for example, **tac-pac bootflash://showtech.switch1**. The **tac-pac** command redirects the output of a **show tech-support details** command to a file which you can then gzip. If no filename is specified, the file created is **volatile:show\_tech\_out.gz**.
  - Copy the file from the switch using the procedure [Copying Files to and from a Switch](#), page 1-17.

- Fabric Manager method:
  - Choose **Tools > Show tech support**. Fabric manager can capture switch configuration information from multiple switches simultaneously. The file can be saved on a local PC.
- For issues with Fabric Manager, capture the fmserver.log log file.

**Step 2** Capture the exact error codes:

- a. The error occurs in Fabric Manager, take a screen shot of the error. In Windows, use **ALT+PrintScreen** to capture the active window and for the entire desktop press **PrintScreen**. Then paste this screen shot into a new **MSpaint.exe** (or similar program) session and save the file.
- b. Copy the error from the message log. Display it using either the **show logging log** command, or to view the last X lines of the log, use the **show logging last #** command.

**Step 3** Ensure that you have answers to the following questions before calling TAC:

1. Which switch, HBA, or storage port is having the problem? List the switch firmware, driver versions, operating systems versions, and storage device firmware.
2. What is the network topology? (From Fabric Manager, select **Tools ->show tech** and save the map.)
3. Were you making any changes to the environment (zoning, adding line cards, upgrades) before or at the time of this event?
4. Are there other similarly configured devices that could have this problem but do not?
5. To what is the problem device connected (MDS switch Z, interface x/y)?
6. When did this problem first occur?
7. When did this problem last occur?
8. How often does this problem occur?
9. How many devices have this problem?
10. Have you examined the syslog (from the **show logging log** command) and accounting log (from the **show accounting log** command) to see if there are any relevant errors or actions that may have caused this condition?
11. Were any traces or debug outputs captured using tools such as:
  - a. Fcanalyzer, PAA-2, Wireshark/Ethereal, local or remote SPAN
  - b. CLI debug commands
  - c. FC traceroute, FC ping
  - d. Fabric Manager/Device Manager SNMP trace
12. What troubleshooting steps have already been done?

## Saving the Configuration Across the Fabric

Rather than logging on to every switch using a script or relaunching Fabric Manager each time, propagate the configuration with CFS.

```
switch# copy running-config startup-config fabric
[#####] 100%
```

This command takes slightly longer than a single **copy running-config startup-config** command takes. When it finishes, all the switches have saved configurations.

## Device Aliases

Device aliases provide a plain text name to port world-wide name (pWWN) mapping. This one-to-one mapping technique was developed to identify devices within the switch environment by labels rather than pWWNs. The device alias can be used in areas such as zoning, QoS, IVR, and throughout Fabric Manager and the Performance Manager utilities. This CFS aware mapping extends to CLI output, where device aliases are used where appropriate.



### Note

Some rules for devices aliases are as follows:

- Mapping is one-to-one of pWWN to a plain text name.
- The CFS scope is physical.
- A device alias database is not tied to a VSAN so if you move an end device (such as an HBA) from one VSAN to another, the device alias still applies.
- The merging of two device alias databases produces a union of the two databases. Conflicts are not imported into the resulting database and must be manually resolved. This will not keep the nonconflicting entries from being merged into the new database. The merge failure will not isolate an ISL.
- The CLI manages device aliases in a central database while Fabric Manager manages them under their respective types (hosts and storage).
- There are two types of device aliases. Regular and enhanced device aliases.



### Tip

- Use the device alias to describe both the host name and HBA instance of the device. For example, host123\_lpf0 or SYMM7890\_FA14ab is a better name than just host123 or SYMM7890.
- The device alias can become the basis for a Fabric Manager Enclosure. Device aliases SYMM7890\_FA14ab and SYMM7890\_FA14ba should be part of the enclosure SYMM7890.

## Standard Device Aliases

With standard device aliases, introduced in SANOS 2.0, the switch performs a world-wide name (WWN) substitution for the device alias, and then passes the WWN to the service or application that is being configured. As a result, changes to the device alias do not reflect changes in the application or service.

For services that leverage device aliasing (zoning, IVR, QOS, etc.), updating the device alias with a new pWWN does not automatically propagate the new information to the services. To update those services, remove the device, update the device alias, and then read the device alias to the service.

## Enhanced Device Aliases

With Enhanced device aliases, introduced in SANOS 3.0, the device alias, and not the WWN it represents, is passed to the application or service being configured. Therefore, changes to the device alias mapping, such as changing the WWN represented by the device alias, is immediately reflected in the application. This advantage is reflected when performing operations that would change the WWN, such as replacing server host bus adapters (HBAs). After changing the WWN that the enhanced device alias represents, you do not need to modify every application that was configured with this enhanced device alias.

Enabling enhanced mode device aliases on a SAN that already has standard device aliases configured, converts the entire device alias database. However, the fabric services or applications, since they contain pWWNs, will not be converted to using device alias as their members.



### Caution

Deleting an enhanced mode device alias will remove it from the applications or services that use it. If you need to rename the device alias or replace the WWN it maps to, do not delete it first.

To enable Enhanced mode device aliases, follow these steps:

#### Step 1 Enter configuration mode.

```
switch# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
```

#### Step 2 Enable enhanced mode device alias.

```
switch(config)# device-alias mode enhanced
```

#### Step 3 If using CFS (highly recommended), commit the change.

```
switch(config)# device-alias commit
```

## Zone Set Output with Enhanced Device Aliases

With Enhanced mode device aliases, the device alias itself, and not the WWN it represents is stored in the zone server. For example:

```
switch# show zoneset active vsan 1000
zoneset name ZS_DNA vsan 1000
zone name ca-aix2_fcs1 vsan 1000
* fcid 0x7f0006 [device-alias ca-aix2_fcs1]
* fcid 0x670005 [device-alias HDS10208-CL5G]
```

Contrast this with the output from a standard device alias:

```
zone name Z_test2 vsan 1000
* fcid 0xef0005 [pwwn 50:06:0e:80:04:27:e0:46] [HDS10208-CL6G]
* fcid 0xec0100 [pwwn 50:06:0b:00:00:13:37:ae] [ca-hpux2_td1]
```

**Note**

It is important to notice that the zone members are a different type. With Enhanced mode device aliases, the members are of type device alias, while in the standard device alias example, the members are pWWN.

## Manipulating Device Aliases with the CLI

Device aliases can be manipulated with the CLI. These recipes demonstrate how device aliases are integrated into the CLI. These procedures are the same if you are using standard device aliases or enhanced device aliases.

### Displaying Device Aliases with the CLI

The CLI can display device aliases in **show** commands such as the following:

- Display the name server

```
ca-9506# show fcns database
```

```
VSAN 1:
```

| FCID     | TYPE | PWWN                    | (VENDOR) | FC4-TYPE:FEATURE |
|----------|------|-------------------------|----------|------------------|
| 0x620000 | N    | 10:00:00:00:c9:32:8b:a8 | (Emulex) | scsi-fcp:init    |
|          |      | [ca-sun1_lpfco]         |          |                  |
| 0x65000a | N    | 10:00:00:00:c9:34:a6:3e | (Emulex) |                  |
|          |      | [ca-aix1_fcs0]          |          |                  |

- Display the active zone set containing IVR and regular zones (Standard device aliases):

```
ca-9506# show zoneset
```

```
zoneset name nozoneset vsan 501
 zone name IVRZ_IvrZone1 vsan 501
 pwwn 50:06:0e:80:03:4e:95:23 [HDS20117-c20-9]
 pwwn 10:00:00:00:c9:32:8b:a8 [ca-sun1_lpfco]

 zone name ca_aix2_HDS vsan 600
 pwwn 10:00:00:00:c9:34:a5:94 [ca-aix2_fcs1]
 pwwn 50:06:0e:80:03:4e:95:23 [HDS20117-c20-8]
```

- Display the flogi database:

```
ca-9506# show flogi database
```

| INTERFACE | VSAN | FCID     | PORT NAME               | NODE NAME               |
|-----------|------|----------|-------------------------|-------------------------|
| fc2/5     | 1000 | 0xef0008 | 50:06:0e:80:03:4e:95:23 | 50:06:0e:80:03:4e:95:23 |
|           |      |          | [HDS20117-c20-9]        |                         |

### Creating Device Aliases with the CLI

In the following examples, these resources are used:

- Host: ca-aix1
- HBA Instance: fcs0
- PWWN: 10:00:00:00:c9:34:a6:3e

To create a device alias using the CLI, follow these steps.

- Step 1** The device alias is enabled by default, so enter configuration mode, and then enter the **device-alias database** command.

```
ca-9506# conf terminal
Enter configuration commands, one per line. End with CNTL/Z.
ca-9506(config)# device-alias database
ca-9506(config-device-alias-db)#
```

- Step 2** Create an entry for the device alias using the **device-alias name** command.

```
ca-9506(config-device-alias-db)# device-alias name ca-aixl_fcs0 pwwn 10:00:00:00
c9:34:a6:3e
```

- Step 3** Display the pending changes with the **show device-alias** command. The plus sign (+) means that the entry is being added to the database, while the minus sign (-) means that it will be removed from the database during CFS commit (Step 4).

```
ca-9506(config-device-alias-db)# do show device-alias database pending-diff
+ device-alias name ca-aixl_fcs0 pwwn 10:00:00:00:c9:34:a6:3e
```

- Step 4** Commit the changes with the **device-alias commit** command.

```
ca-9506(config-device-alias-db)# device-alias commit
ca-9506(config)#
```

## Converting Fibre Channel Aliases to Device Aliases

Fibre Channel aliases in an existing MDS environment can be duplicated to device aliases by importing the FC aliases using the CLI. Only those FC aliases that are also valid device aliases will be imported. An FC alias is eligible to be imported or converted if the following conditions are true:

- The FC alias represents a pWWN.
- The FC alias represents exactly one device and not a group of devices.
- A device alias with the same name does not already exist.
- A device alias with the same pWWN does not already exist.



### Note

- While the device alias database is maintained on all switches, it has a physical scope. FC aliases have a VSAN scope and may not be present or replicated to all switches if Full Zoneset Distribution is not enabled. You may have to log on to multiple switches to import all of the FC aliases.
- Importing FC aliases does not automatically update any zones based on FC aliases. The zones must be manually converted to zones based on device aliases.
- Importing FC aliases does not delete the FC aliases. The FC aliases need to be manually deleted.

In this example the following Fibre Channel aliases are imported and converted into device aliases:

```
fcalias name alias123 vsan 1

fcalias name temphost vsan 1
 pwwn 11:11:11:11:11:11:11:11

fcalias name temphost vsan 2
```



```
pwwn 11:11:11:11:11:11:99:99

fcalias name temphost2 vsan 2
pwwn 11:11:11:11:11:22:22:22
```

Ineligible Fibre Channel aliases are listed in VSAN 1 and 2; these Fibre Channel aliases failed to be imported.

To import and convert Fibre Channel aliases to device aliases, follow these steps:

---

**Step 1** Enter configuration mode.

```
switch# conf terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)#
```

**Step 2** Import the FC aliases with the **device-alias import** command.

```
switch(config)# device-alias import fcalias vsan 1-2
WARNING: Some fc aliases from the specified VSAN range could not be imported due to
conflicts.
```

- a. If a warning is displayed, check the logs to determine which FC aliases did not import:

```
switch(config)# do show device-alias internal errors
1) Event:E_DEBUG, length:111, at 608209 usecs after Wed Sep 14 16:25:13 2005
 [109] ddas_import_fcalias(703): CONFLICT: fcalias temphost on vsan 2 has same name
 as fcalias temphost on vsan 1

2) Event:E_DEBUG, length:127, at 607826 usecs after Wed Sep 14 16:25:13 2005
 [109] ddas_import_getnext_alias_resp_handler(1119): not importing alias alias123
 from vsan 1 as the number of members are not 1.
```

The message tells you that the FC alias **temphost** of VSAN 2 was not imported because the name **temphost** already exists on VSAN 1. Also, **alias123** was not imported. Examining FC alias **alias123** determines that it is not a valid device alias because it does not have any pWWN members.

- b. These conflicts should be resolved by deleting the FC alias **alias123** and renaming the conflicting alias **temphost**.



**Note**

---

FC aliases might be part of an existing zone, so the appropriate zone(s) should be updated accordingly.

---

**Step 3** Display the pending device alias database to see the newly imported device aliases.

```
switch(config)# do show device-alias database pending-diff
+ device-alias name temphost2 pwwn 11:11:11:11:11:22:22:22
```

**Step 4** CFS commit the pending changes.

```
switch(config)# device-alias commit
```

---

## Device Aliases with Fabric Manager

Fabric Manager can use device aliases to provide plain text names in many locations including the map, zoning, and QoS. However, to do this, Fabric Manager needs to be configured to use device aliases instead of FC aliases. Device aliases can be enabled either during installation or afterward. See [Enabling Fabric Manager to use Device Aliases, page 1-46](#). Fabric Manager can leverage either standard or enhanced device aliases.



### Note

We recommend that you do not mix both modes of device aliases within the same fabric.

In [Figure 1-14](#), DPVM is using device aliases to represent the pWWNs in its configuration. HDS20117-c20-8 in VSAN 1000 is plugged into switch 172.22.36.11. The port fc1/1 is easier to understand than the same description using just a pWWN. In this naming scheme the model (HDS), serial number (20117) cluster (20) and port (8) are all used in the name to specifically describe the device. (See [Figure 1-14](#).)

Figure 1-14 DPVM Leveraging Device Aliases

The screenshot shows the Fabric Manager interface. On the left, a tree view under 'Logical Domains' shows a hierarchy: SAN > Fabric 172.22.36.9 > All VSANs > IVR > DPVM (Dynamic Member) > Domain Manager > VSAN0001. On the right, a table titled 'Config Database' displays device aliases. The table has columns: Master, Type, WWN or Name, VSAN Id, and Switch Interface. The data rows show a device alias 'HDS10208-CL5G' mapped to pWWN '50:06:0e:80:03:4e:95:23' in VSAN 1000, connected to switch interface '172.22.36.11 fc1/1'.

| Master      | Type        | WWN or Name                    | VSAN Id | Switch Interface   |
|-------------|-------------|--------------------------------|---------|--------------------|
| 172.22.36.8 | deviceAlias | HDS10208-CL5G                  | 1000    | 172.22.36.11 fc1/1 |
| 172.22.36.8 | pWWN        | 50:06:0e:80:03:4e:95:23        | 1000    |                    |
| 172.22.36.8 | pWWN        | 50:06:0e:80:03:4e:95:33        | 501     |                    |
| 172.22.36.8 | pWWN        | Ologic 21:00:00:e0:8b:09:78:47 | 1000    | 172.22.36.11 fc1/1 |

## Enabling Fabric Manager to use Device Aliases

To enable device aliases in Fabric Manager during installation, check the **Use Global device aliases in place of FC Aliases** check box on the initial installation screen.

To enable device aliases in Fabric Manager after installation, configure Fabric Manager by checking the **Device Alias** check box under **Admin -> Server**.

## Creating a Device Alias for an Existing Device

Before performing any device alias procedures in Fabric Manager, configure Fabric Manager to use device aliases. Fabric Manager can use device aliases to provide plain text names in many locations including the map, zoning, and QoS. However, to do this, Fabric Manager needs to be configured to use device aliases instead of FC aliases. Device aliases can be enabled either during installation or afterward. See the [“Enabling Fabric Manager to use Device Aliases” section on page 1-46](#).

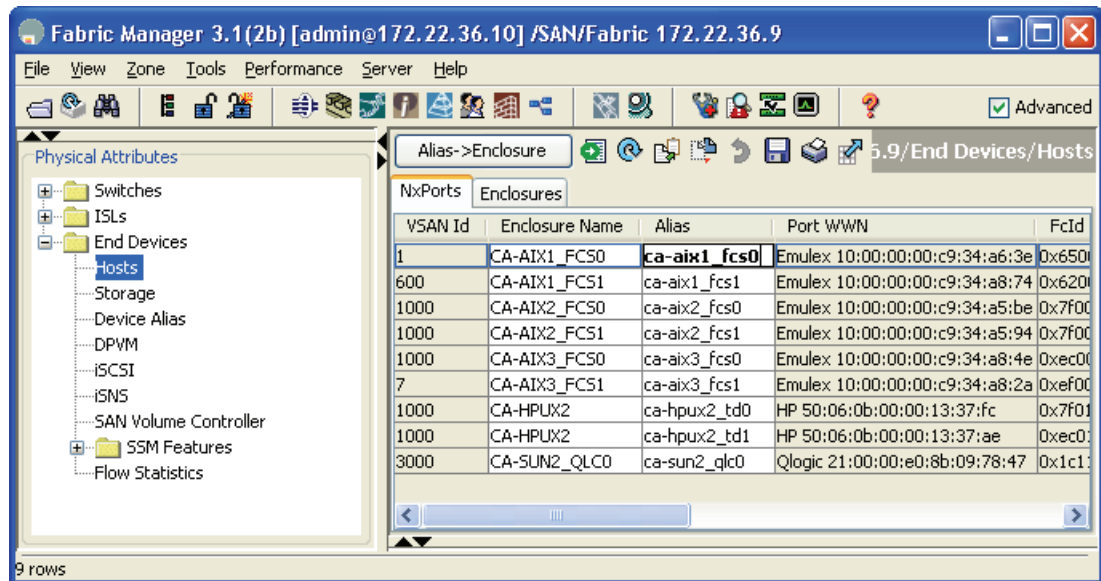
These resources are used in the example:

- Host: ca-aix1
- HBA instance: fcs0
- PWWN: 10:00:00:00:c9:34:a6:3e

To create a device alias in Fabric Manager for a device already logged into the fabric, follow these steps:

- Step 1** In Fabric Manager's **Physical Attributes** pane, expand **End Devices**.
- Step 2** Because the WWN corresponds to a host, choose **Hosts**. (For a storage device, you would choose **Storage**.)
- Step 3** In the device alias column, enter the device alias for the corresponding pWWN.
- Step 4** Click **Apply Changes**. If CFS is enabled for device-alias, this performs an implicit device-alias commit. (See [Figure 1-15](#).)

Figure 1-15 Creating a Device Alias with Fabric Manager



## Creating a Device Alias for a New Device

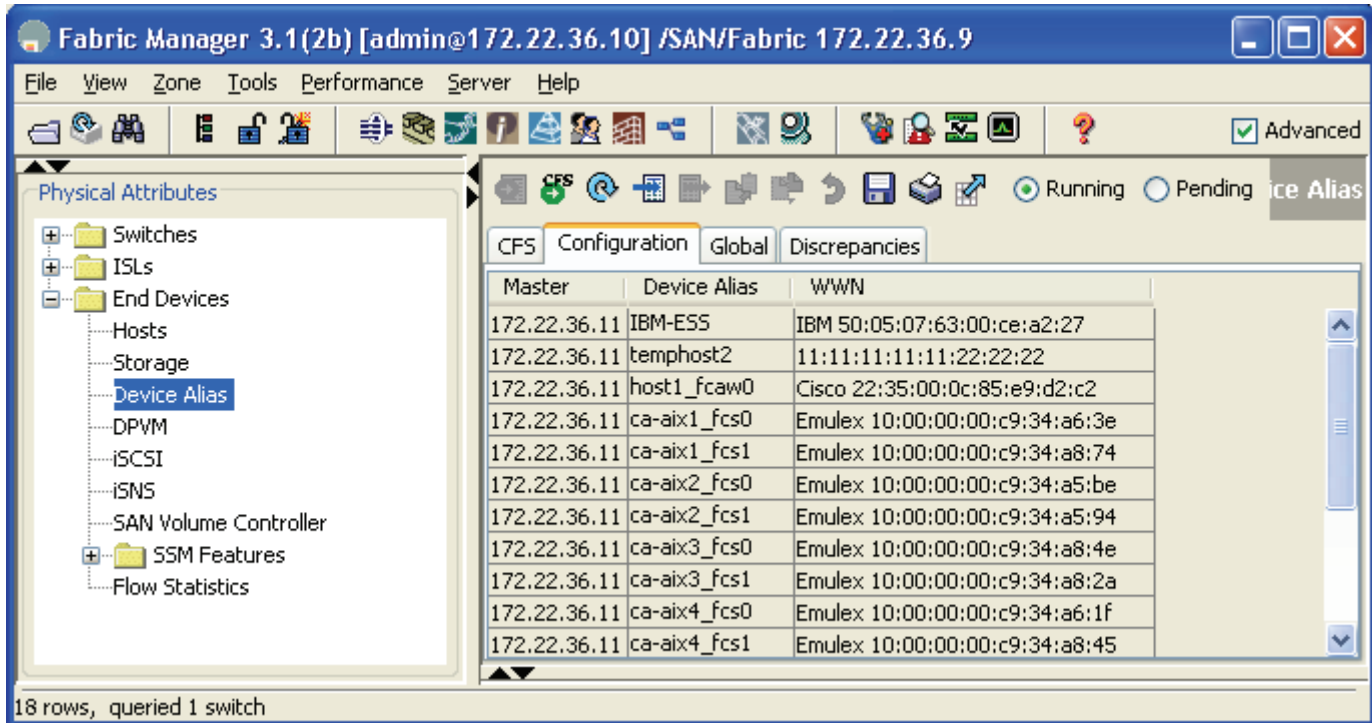
This recipe creates a device alias for a device that is not logged into the switch using CFS. This recipe uses these example resources.

- Device alias HDS10208-LC5G
- Port WWN 50:06:0e:80:04:27:e0:46

To create a device alias for a device that is not logged into the switch using CFS, follow these steps:

- Step 1** In the Physical Attributes pane, expand **End Devices** (see [Figure 1-16](#)).
- Step 2** Choose **Device Alias** (see [Figure 1-16](#)).
- Step 3** Choose the **Configuration** tab (see [Figure 1-16](#)).

Figure 1-16 Creating a New Device Alias



- Step 4** Click **Create Row...**
- Step 5** Enter the device alias and WWN in the corresponding fields.
- Step 6** Click **Create**.
- Step 7** When all device aliases have been created, click **Close**.
- Step 8** Click **Apply Changes**.
- Step 9** If CFS is enabled click **Commit CFS Pending Changes**.

## Implementing Syslog

The syslog facility allows the MDS 9000 Family of switches to send a copy of the message log to a host for more permanent storage. This can be useful if the logs need to be examined over a long period of time or if the MDS switch is not accessible.

This example configures an MDS switch to use the syslog facility on a Solaris platform. Although a Solaris host is being used, syslog configuration on all UNIX and Linux systems is very similar.

Syslog can discriminate between messages of different severity and handle them differently. For example, messages can be logged to different files or sent by e-mail to a particular person. Specifying a level of severity determines that all messages of that severity level and greater (lower numbers are higher severity) are acted upon.

**Tip**

MDS messages should be logged to a different file than the standard syslog file so they are not confused with nonMDS syslog messages. The log file should not be located on the / filesystem to prevent log messages from filling up the / filesystem.

In this example, the following resources are used:

- Syslog client: switch1
- Syslog server: 172.22.36.211 (Solaris)
- Syslog facility: local1
- Syslog severity: notifications (level 5, the default)
- File to log MDS messages to: /var/adm/MDS\_logs

To configure an MDS switch to use the syslog facility on a Solaris platform, follow these steps:

**Step 1** Enter configuration mode and configure the MDS switch.

```
switch1# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch1(config)# logging server 172.22.36.211 6 facility local1
```

**Step 2** Display the switch configuration with the **show logging server** command.

```
switch1# show logging server
Logging server: enabled
{172.22.36.211}
 server severity: notifications
 server facility: local1
```

**Step 3** Configure the Syslog server.

- a. Modify /etc/syslog.conf to handle local messages. For Solaris, there must be at least one tab between the facility.severity and the action (/var/adm/MDS\_logs)

```
#Below is for the MDS 9000 logging
local1.notice /var/adm/MDS_logs
```

- b. Create the log file.

```
#touch /var/adm/MDS_logs
```

- c. Restart the syslogd with the commands **syslog stop** and **syslog start**.

```
/etc/init.d/syslog stop
/etc/init.d/syslog start
syslog service starting.
```

- d. Verify that the syslog started.

```
ps -ef |grep syslogd
root 23508 1 0 11:01:41 ? 0:00 /usr/sbin/syslogd
```

**Step 4** Test the syslog server by creating an event on the MDS switch. In this example, port fc1/2 is reset and the information is listed on the syslog server. Notice that the IP address of the switch is listed in brackets.

```
tail -f /var/adm/MDS_logs
Sep 17 11:07:41 [172.22.36.142.2.2] : 2004 Sep 17 11:17:29 pacific:
%PORT-5-IF_DOWN_INITIALIZING: %$VSAN 1%$ Interface fc1/2 is down (Initializing)
Sep 17 11:07:49 [172.22.36.142.2.2] : 2004 Sep 17 11:17:36 pacific: %PORT-5-IF_UP:
%$VSAN 1%$ Interface fc1/2 is up in mode TE
```

```
Sep 17 11:07:51 [172.22.36.142.2.2] : 2004 Sep 17 11:17:39 pacific:
%VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from pts/0
(dhcp-171-71-49-125.cisco.com)
```

## Configuring Call Home

The following recipes configure MDS switches to send call home e-mail messages for notification of an issue with the switch. These messages can be directed either to a pager service or an e-mail account.

## What are Alert Groups?

Alert groups determine which events are sent to specific destinations by using profiles. For example, a profile may be configured to include the facilities team. When an environmental alert is triggered, all members receive a text message on a pager. (See [Table 1-17](#).)

Table 1-17 Default Alert Group Definitions

| Alert Group         | Description                                                                                                                      | Executed Commands                                  |
|---------------------|----------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------|
| System              | Events generated by failure of a software system critical to unit operation.                                                     | show tech-support<br>show system redundancy status |
| Environmental       | Events related to power, fan, and temperature.                                                                                   | show module<br>show environment                    |
| Line Card Hardware  | Events related to standard or intelligent line cards.                                                                            | show tech-support                                  |
| Supervisor Hardware | Events related to supervisor or fabric cards.                                                                                    | show tech-support                                  |
| License             | Events related to unlicensed use of licensed features.                                                                           | show license all<br>show running-config            |
| Inventory           | Inventory is a noncritical event. Status should be provided whenever a unit is cold booted or when FRUs are inserted or removed. | show version                                       |
| RMON                | Events related to RMON, triggered by Threshold Manager to set alerts.                                                            |                                                    |
| Syslog-group-port   | Events related to syslog messages filed by Port Manager when a port goes up or down.                                             |                                                    |
| Test                | User-generated test messages.                                                                                                    | show version                                       |
| Avanti              | Events related to the IBM Caching Service Module.                                                                                |                                                    |
| Cisco-TAC           | Events intended for only Cisco TAC.                                                                                              |                                                    |

## Configure Call Home to Send All Notifications to a Single E-Mail Address

The simplest MDS notification strategy is to send an e-mail for all events. E-mail is a better choice than a pager notification because e-mail is not space-limited and can contain full details of the event.

In this recipe, CFS will not be enabled for Call Home.

**Note**

---

If all the switches use the same Call Home configuration, then CFS should be enabled.

---

In this example, these assumptions are made:

- Contact: Storage Admins
- Phone number: 123-456-7890
- Mail address: storageadmins@acme.com
- Street address: 123 Main Street
- Switch's e-mail address: mds-callhome@acme.com
- Destination e-mail address (who to mail the error to): NOC@acme.com
- SMTP server: 192.168.1.2

To Configure Call Home to Send All Notifications to a Single E-Mail Address, follow these steps:

- 
- Step 1** In Device Manager, from the **Admin** pull-down menu, choose **Events**.
- Step 2** Choose **Call Home...** You see the screen in [Figure 1-17](#).

Figure 1-17 Call Home General Tab

The screenshot shows a web-based configuration window titled "172.22.36.142 - Call Home". It has a blue header bar with a close button (X) in the top right corner. Below the header is a tabbed interface with five tabs: "General" (selected), "Destinations", "Email Setup", "Alerts", and "Profiles". The "General" tab contains two main sections: "Contact Information (Required)" and "Ids".

**Contact Information (Required)**

- Contact: StorageAdmins
- PhoneNumber: +1-123-456-7890
- EmailAddress: storageadmins@acme.com
- StreetAddress: 123 Main Street

**Ids**

- CustomerId: (empty field)
- ContractId: (empty field)
- SiteId: (empty field)

**DeviceServicePriority:**

A grid of radio buttons for selecting the priority level:

|                                             |                                 |                                   |                                |
|---------------------------------------------|---------------------------------|-----------------------------------|--------------------------------|
| <input type="radio"/> emergency(1)          | <input type="radio"/> alert(2)  | <input type="radio"/> critical(3) | <input type="radio"/> error(4) |
| <input checked="" type="radio"/> warning(5) | <input type="radio"/> notice(6) | <input type="radio"/> info(7)     | <input type="radio"/> debug(8) |

Below the radio buttons is a checkbox labeled "Enable" which is checked.

At the bottom right of the window are four buttons: "Apply", "Refresh", "Help", and "Close".

- Step 3** Complete the appropriate fields (see Figure 1-17). The Device Service Priority you select will be included in e-mail notifications.
- Step 4** Check the **Enable** check box (see Figure 1-17).
- Step 5** Click **Apply** (see Figure 1-17).
- Step 6** Click the **Email Setup** tab (see Figure 1-18).
- Step 7** Complete the **From** and **SMTP** fields (see Figure 1-18).
- Step 8** Click **Apply** (see Figure 1-18).



Figure 1-18 Call Home E-mail Setup

- Step 9** Choose the **Profiles** tab (see [Figure 1-19](#)). A profile determines what types of notifications are sent.
- Step 10** Click **Create...** (see [Figure 1-19](#))
- Step 11** Enter a name for the profile (see [Figure 1-19](#)).
- Step 12** Select the message level **debug** (see [Figure 1-19](#))
- Step 13** Select MaxMessageSize **0** (zero limit to message size) See [Figure 1-19](#).
- Step 14** Check all of the Alert Groups. (See [What are Alert Groups?](#), page 1-50 for more information on Alert Groups). See [Figure 1-19](#).

**Note**

- If you do not have an IBM Caching Service Module installed, uncheck **Avanti**.
- If you do not want to receive a Call Home message every time a port goes up or down, uncheck **syslogGroupPort**.

Figure 1-19 Call Home Profile

| Profile      | Msg...    | MaxM... | MsgL... | AlertGroups                                                                  |
|--------------|-----------|---------|---------|------------------------------------------------------------------------------|
| xml          | xml       | 500000  | debug   | ciscoTac                                                                     |
| full_txt     | fullText  | 500000  | debug   | system environmental linecard supervisor inventory test avanti ciscoTac s... |
| short_txt    | shortText | 4000    | debug   | system environmental linecard supervisor inventory test avanti ciscoTac s... |
| MDS_CallHome | fullText  | 0       | debug   | system environmental linecard supervisor inventory test ciscoTac RMON li...  |

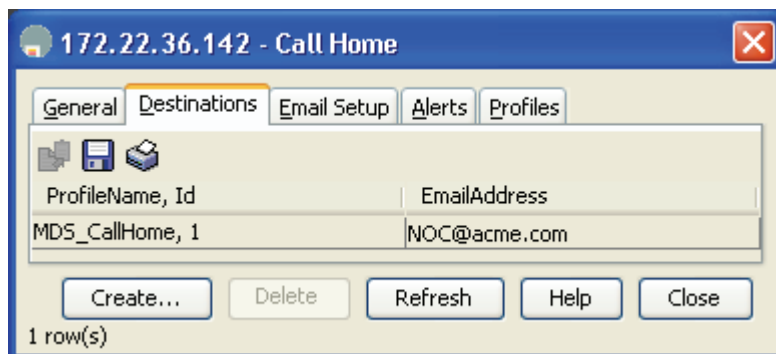
- Step 15** Click **Create...** (see [Figure 1-19](#))
- Step 16** Choose the **Destinations** tab (see [Figure 1-20](#)).
- Step 17** Click **Create...** (see [Figure 1-20](#))
- Step 18** Change the profile to the one you just created in the Profiles tab.

**Note**

- Use **XML** if the Call Home message's destination is Cisco TAC.
- Use **full\_txt** if the Call Home message will be read as an e-mail.
- Use **short\_txt** if the Call Home message is destined for a pager or similar device.

- Step 19** Enter the e-mail address for the Call Home message (see [Figure 1-20](#)).
- Step 20** Click **Create...** (see [Figure 1-20](#))
- Step 21** Click **Close** (see [Figure 1-20](#)).
- Step 22** Test the configuration by choosing the **Alerts** tab, the **Test** action, then clicking **Apply** (see [Figure 1-20](#)). If errors occur, they are displayed in the Failure Cause text box.

Figure 1-20 Call Home Destinations





## CHAPTER 2

# Managing Fabric Manager Server

---

## Managing Fabric Manager

The recipes in this section are for configuring Fabric Manager Client and Server applications rather than for configuring objects that reside only on the switch.

## Optimizing Fabric Manager Server Performance

When deploying Fabric Manager Server, further optimizations can be done to the platform to enhance the performance and stability of the platform. As your fabric increases in size, the number of ports being monitored by both the Fabric Manager Server and Performance Manager are the primary factors when determining the size of the server that is hosting the application.

Optimizing the performance of Fabric Manager Server often requires tuning the Java Runtime Environment, Fabric Manager Server, and the database that it uses.

## Installing the Correct Java Runtime Environment

Before installing Fabric Manager Server, a Sun Java Runtime Environment (JRE) must be installed. On any system that will be running only the Fabric Manager client (GUI) or only Device Manager, the standard JRE can be installed from <http://www.java.com>. However, for any host that will exclusively run the Fabric Manager Server application, a different JRE should be installed, that is only available through the Java Software Development Kit (SDK).

The Sun JRE comes in two versions: a client version and a server version. The difference between the two is that these two systems contain different binaries. They are essentially two different compilers (JITs) interfacing to the same runtime system. The client system is optimal for applications that need fast start up times or small footprints (such as the Fabric Manager Client or Device Manager), while the server system is optimal for applications where the performance is most important (Fabric Manager Server). In general, the client system is better for GUIs, while the server system is optimized for applications that run continuously, such as services. Some of the other differences include the compilation policy used, heap defaults, and inlining policy. Most Windows machines are not considered to be server class, as described at this URL:

<http://java.sun.com/j2se/1.5.0/docs/guide/vm/server-class.html>. For Windows systems, the default installation is the client version of the JRE.

To run Fabric Manager using the server JIT, perform the following steps on the Fabric Manager Server (client machines do not need to be changed):

**Note**

If Fabric Manager Server is already installed on your server and you want to enable the server JDK for Windows, skip [Step 5](#).

- 
- Step 1** Download the latest update of **JDK 5.0** (Java EE or NetBeans versions are not needed) from [http://java.sun.com/javase/downloads/index\\_jdk5.jsp](http://java.sun.com/javase/downloads/index_jdk5.jsp). This is the third download button from the top.
- Step 2** Accept the license agreement on the following page.
- Step 3** If the host has access to the Internet, download the online version, otherwise download the offline version.
- Step 4** While installing the JDK, you have the opportunity to not install certain packages. You do not need to install Demos or Source Code. The installation path is slightly different from the other JREs because it contains *jdk version* in the path.
- Step 5** Install Cisco Fabric Manager Server.
- Step 6** Verify that the **FMServer.conf**, **FMWebClient.conf** (and if you are using the pre-3.2 packaged database (HSQLdb), and the **FMPersist.conf** files are in the `c:\program files\cisco systems\mds 9000\conf` directory for the new path to the `javaw.exe` file (for example, `wrapper.java.command=C:\Program Files\Java\jdk1.5.0_12\bin\javaw.exe`). If you are using an Oracle or PostgreSQL database as the backend database for Fabric Manager Server, you do not need to modify the `FMPersist.conf` file. It should look like this:
- ```
*****
# Wrapper Properties
*****
# Java Application
wrapper.java.command=C:\Program Files\Java\jdk1.5.0_12\bin\javaw.exe
```
- Step 7** Uncomment the `-server` option in both files.
- ```
#Uncomment the following line if you use the Windows JDK
#instead of JRE
wrapper.java.additional.9=-server
```
- Step 8** Restart Fabric Manager Server, the web server, and the database if using the pre-3.2 packaged HSQLdb database. If you are using Oracle Express or PostgreSQL, you do not need to restart the database.
- 

## Performance Manager Database Sizing

Fabric Manager Server provides persistent monitoring, which means that it continues to monitor or poll the fabric even after the Fabric Manager Client has closed. When setting up Fabric Manager for persistent monitoring, be aware of the disk space that the Fabric Manager database uses, even if Performance Manager is enabled. Performance Manager provides Fabric Manager Server with the ability to collect performance statistics on ports, ISLs, or flows (end device-to-end device communication).

Fabric Manager Server stores its data in a round-robin database (RRD). An RRD is a database that contains a fixed number of records. When the last record in the database has been written, the next record to be written overwrites or updates the first record, which keeps the size of the database under control. Also, as data points get older, the granularity of those points becomes less relevant and the trend of those points becomes more important. Performance Manager keeps its RRD files in `C:\Program Files\Cisco Systems\MDS 9000\pm\db` directory for Windows and `/usr/local/cisco_mds9000/pm/db` for Unix/Linux.

Table 2-1 lists the default polling values for Fabric Manager and Performance Manager. To understand the table, read it in the following manner: “X days of polling, every Y (interval), results in Z samples.” Or you can read it this way: “If I take Z samples every Y interval, I’ll have that granularity for X days.” Increasing the number of days to keep a sample will increase the size of the database. These values can be modified through the Performance Manager web GUI, under the **Admin** tab. Follow the resulting screen on the left menu to **Performance, Database**.

Table 2-1 Fabric Manager Server Polling Values

| Day  | Interval | Samples |
|------|----------|---------|
| 2.0  | 5 min    | 600     |
| 15.4 | 30 min   | 700     |
| 64   | 2 hour   | 775     |
| 300  | 1 day    | 300     |

For the purpose of this chapter, default values are used to calculate how much disk space is required. The default values for Performance Manager require the following amount of space:

- **77KB** per port/ISL/flow traffic monitored.
- **39KB** per error/discards if enabled.

A 1000 port fabric with 10 ISLs, with 4:1 hba:target fanout (which is used to determine the number of flows) would use the following amount of disk space:

- 800 zones, and if configured, 800 flows.  $(800 \times 77 \text{ KB}) = 61600 \text{ KB}$ .
- 1000 port traffic and errors/discards  $(1000 \times 77 \text{ KB} + 1000 \times 39 \text{ KB}) = 116000 \text{ KB}$ .
- 10 ISL traffic and error files  $(10 \times 77 \text{ KB} + 10 \times 39 \text{ KB}) = 1160 \text{ KB}$ .

This Fabric Manager Server would require 74360 KB (~74 MB of disk space) consumed in 1820 RRD files. Although this is a trivial amount of storage in disk drives of today, the number of RRD files could create a bottleneck for the Fabric Manager Server because it would need to update all 1820 RRD files every five minutes. It is important to locate the RRD files on fast storage. It is not the throughput (MB/s) of the storage device that is important but the number of I/O /sec that can be performed. A RAID-1 array for writes, performs exactly the same as a single disk device. Just mirroring the internal disk for the Fabric Manager Server may not increase the write performance of this task.

## Configuring Fabric Manager Server to Use an External Oracle Database

Introduced in Fabric Manager 3.1(2), Fabric Manager Server supports the ability to connect to an external Oracle 11i database server to host its database. This is a different database from the one used for Performance Manager as discussed in [Performance Manager Database Sizing, page 2-2](#). This database, instead of holding performance data, contains the inventory, fabric configuration, and the states of the ports within the fabric.

To configure Fabric Manager Server to use an external Oracle 11i server, follow these steps:

- 
- Step 1** Install Fabric Manager Server and verify that you can discover the fabrics.
- Step 2** Give your database administrator the file **c:\Program Files\Cisco Systems\mds9000\db\admin\oraclesetup.sql**. The database administrator will run the file on the database server to create the database schemas.

**Step 3** Shut down Fabric Manager Server, web services, and the Cisco database from within Windows services. The Cisco database can be disabled as it is no longer needed.

**Step 4** In the **server.properties** file (typically found in C:\Program Files\Cisco Systems\MDS 9000), uncomment “Database Property for Oracle” (remove the # from before the **db.<variable name>**) for all the entries in Section 10.



**Note** Your database administrator can provide you with the information for **db.url**, **db.user** and **db.password**. Only these three variables should be modified; the rest can remain uncommented.

**Step 5** Start the Fabric Manager Server Service and discover the fabrics.

**Step 6** Let the Fabric Manager Server run for about 15 minutes.

To preserve the data from previously defined Performance Manager collections, follow these steps:

**Step 1** Shut down Fabric Manager Server.

**Step 2** From a Windows command prompt, run **C:\Program Files\Cisco Systems\MDS 9000\bin\pm.bat syncDB** to import and index the old Performance Manager data into the new Oracle database.

**Step 3** Restart Fabric Manager Server.

**Step 4** Recreate your Performance Manager collections as described in [Creating a Collection in Performance Manager, page 2-10](#). It can take at least 15 minutes for new flow data to appear within the web GUI; however, there is some historical data because you just imported your old .rrd files.

## Adjusting Memory Usage of Fabric Manager

By default, Fabric Manager Server can use up to 256 MB of RAM; however, for larger installations, you might start to notice that Fabric Manager Server approaches that amount of memory. When this level is reached, you will start to see Out of Memory messages displayed in the fmserver.log log file. To adjust the maximum amount of memory, modify the file **C:\Program Files\Cisco Systems\MDS 9000\conf\FMServer.conf**:

```
:# Initial Java Heap Size (in MB)
wrapper.java.initmemory=6
wrapper.java.maxmemory=512
```



**Note** Fabric Manager Server only uses memory if it needs it. So, for example, allocating 512 MB of memory when the Fabric Manager Server only needs 96 MB will not increase performance.

## Authenticating Fabric Manager Through TACACS

With the introduction of Fabric Manager Release 3.1, Fabric Manager requires authentication outside of the authentication provided by the switch. While it is possible for Fabric Manager Server to manage a username and password database locally, we recommend that if you have an existing TACACS+ or RADIUS environment that provide authentication services for your switches, that the same TACACS+ or RADIUS servers be used to provide authentication for the Fabric Manager Server.

In this recipe, the following information came from the team that manages the TACACS+ environment:

- Primary TACACS+ IP address: 11.7.20.7
- Primary server TACACS+ secret: g0s0x!
- Authentication method: pap
- Secondary TACACS+ IP address: 11.7.20.8
- Secondary server TACACS+ secret: f@z3d@z3
- Authentication method: pap



#### Tip

We recommend that you have at least two AAA servers. If you have just one server and it is not reachable, you will not be able to log into Fabric Manager Server unless you switch back to local authentication.

While you can perform the following procedure using the Fabric Manager web client to make the changes, this recipe modifies the configuration file.

To configure TACACS+ authentication services, follow these steps:

**Step 1** Open the file **C:\Program Files\Cisco Systems\MDS 9000\AAA.properties**.

**Step 2** Change **authentication.mode=local** to **authentication.mode=tacacs**.

**Step 3** Configure the primary AAA server:

```
#Primary AAA server
aaa.server.primary.name=11.7.20.7
aaa.server.primary.secret=g0s0x!
Authentication method. PAP and CHAP are only allowed for Radius
Authentication method. PAP,CHAP,MSCHAP and ASCII are only allowed for TACACS+
aaa.server.primary.auth-method=pap
```

**Step 4** Configure the secondary AAA server.

```
#Secondary AAA server
aaa.server.secondary.name=11.7.20.8
aaa.server.secondary.secret=f@z3d@z3
aaa.server.secondary.auth-method=pap
```

**Step 5** Restart the Fabric Manager Server process.

At this point, when a user logs into Fabric Manager Server, you will see entries in **C:\Program Files\Cisco Systems\MDS 9000\fmserver.log** logging their success or failure to log into Fabric Manager Server through the TACACS+ server.

## Operating Fabric Manager Through a Firewall Using SNMP Proxy

With the Fabric Manager Server, a storage administrator can connect to an MDS switch behind a firewall. The Fabric Manager Client application encapsulates the SNMP protocol in TCP which traverses the firewall and connects to the Fabric Manager Server. The Fabric Manager Server then forwards the SNMP packets to the switch.

For the Fabric Manager Client to connect to the Fabric Manager Server, the following TCP ports must be open on the firewall:

- TCP port 9198 — The Fabric Manager Client uses this TCP port to send encapsulated SNMP packets to the Fabric Manager Server.
- TCP ports 9099-9200 — TCP port 9099 is used for the `java.rmi.registry.port` while the other ports are used for `java.rmi.server.remoteObjectPort`.
- TCP port 80 — This port is used to view the Performance Manager's web based statistics.

Additionally, the Fabric Manager Client uses the CLI to obtain information on the switch directly for some features. One of the following two options should be used to allow the Fabric Manager Client to access the switch:

- TCP port 23 when the Fabric Manager Client has been configured to use Telnet for CLI access into the Cisco MDS switch. This is the default setting for the Fabric Manager Client.
- TCP port 22 when the Fabric Manager Client has been configured to use SSH for CLI access into the Cisco MDS switch. This is the preferred setting for this environment, and the switch must have SSH enabled.

**Note**

---

All ports that the Fabric Manager Client and Server use can be changed in the `server.properties` file in the directory `C:\Program Files\Cisco Systems\MDS 9000`.

---

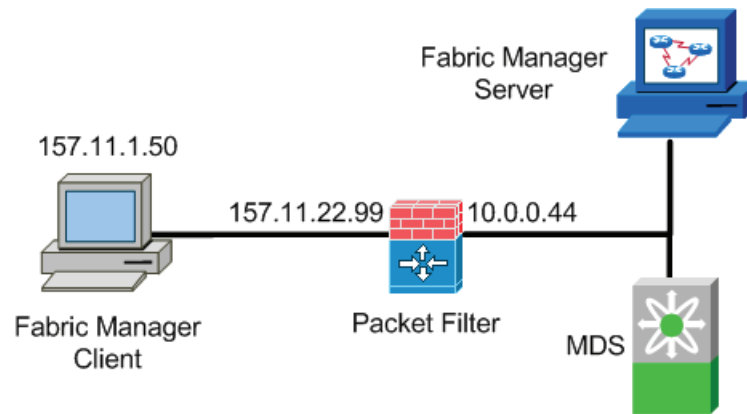
## Configuration Using a Non-NAT Packet Filter

In this recipe, the Fabric Manager Client is separated from the Fabric Manager Server and Cisco MDS switch by a packet filtering device that does not perform network address translation (NAT). Additionally, the packet filter allows unrestricted access to the external network to any internal device. Last, SSH has been configured on the switch and Telnet has been disabled.



This example uses the topology shown in [Figure 2-1](#).

Figure 2-1 SNMP Proxy Topology



To separate the Fabric Manager Client from the Fabric Manager Server and Cisco MDS switch by a packet-filtering device that does not perform NAT, follow these steps:

- 
- Step 1** Configure the firewall to allow the following TCP connections:
- TCP port 9198 — The Fabric Manager Client uses this TCP port to send the encapsulated SNMP packets to the Fabric Manager Server.
  - TCP ports 9099-9200 — TCP port 9099 is used for java.rmi.registry.port while the other ports are used for java.rmi.server.remoteObjectPort.
  - TCP port 23 — Some features of the Fabric Manager client require Telnet access into the switch. However, if SSH is enabled on the switch, the Fabric Manager Client can leverage SSH instead. If SSH is used, then this port is not required.
  - TCP port 80 — Can be used to view the Performance Monitor's web-based statistics.
  - TCP port 22 — Should be enabled if Telnet on the switch is disabled and SSH is enabled.
- Step 2** Configure the host running the Fabric Manager Client to reach the network behind the firewall where the Fabric Manager Server resides. The 10.0.0.0 network is behind the firewall, while 157.11.22.99 is the external IP address of the firewall. This is not the mgmt0 address of the firewall. On a windows host this configuration can be accomplished with the following command:
- ```
C:\>route add 10.0.0.0 mask 255.255.255.0 157.11.22.99
```
- Step 3** Configure the switch to reach the Fabric Manager Client (157.11.1.50) which is outside of the firewall. The internal IP address (the side facing the switch) of the firewall is 10.0.0.44.
- ```
switch(config)# ip route 157.11.1.50 255.255.255.0 10.0.0.44
```
- Step 4** Launch the Fabric Manager Client, and in the login screen, check the **Use SNMP Proxy** check box and enter the IP addresses of the Fabric Manager Server and the MDS switch.
-

## Performance Manager Using Fabric Manager Server

Performance Manager is a licensed feature that is part of Fabric Manager Server. Performance Manager provides historical analysis of SAN statistics and is displayed graphically to a web browser.



### Note

Performance Manager requires the Fabric Manager Server License. This license is required for all switches on which performance data is gathered.

Fabric Manager Server must be configured to run Performance Manager. Select a host that is always up and has enough storage to gather and store the performance data.

A Performance Manager data collection configuration is shown in the recipe.



### Tip

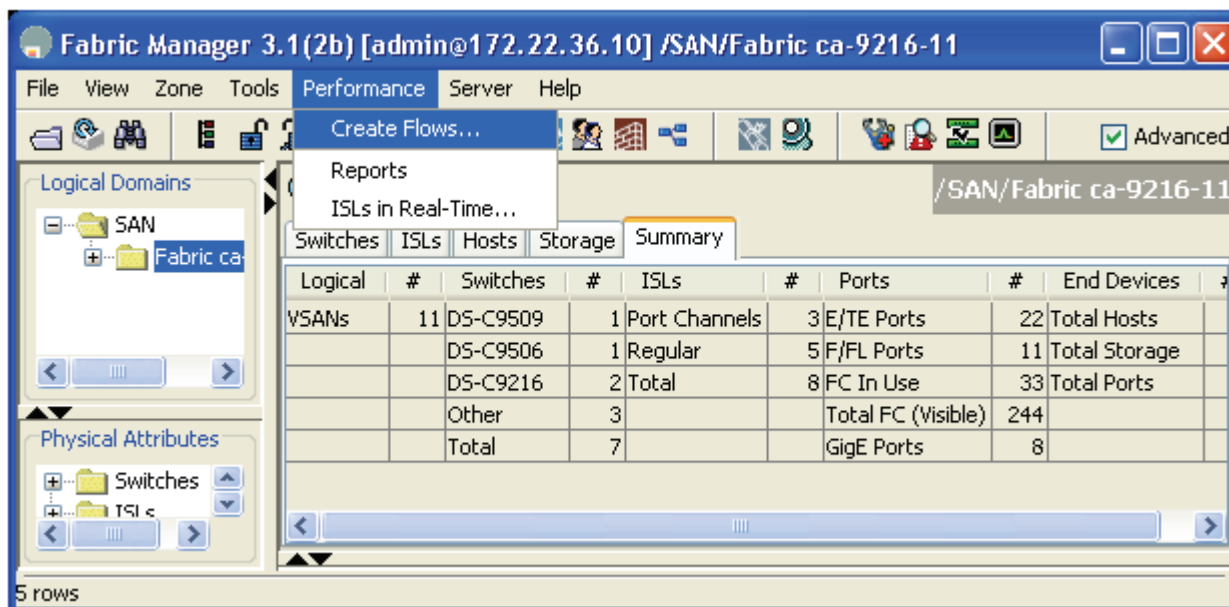
If you are upgrading to FabricManager 3.1(x), you should recreate your flows. Recreating them will remove old flows and by not choosing to “Create flows on all cards” you can optimize your flow collection and reduce the load on both the switch and the Fabric Manager Server.

## Creating Flows Within Fabric Manager

To create a flow with Fabric Manager, follow these steps:

- Step 1** Within Fabric Manager, select the **Performance** menu and choose **Create flows...** as shown in Figure 2-2.

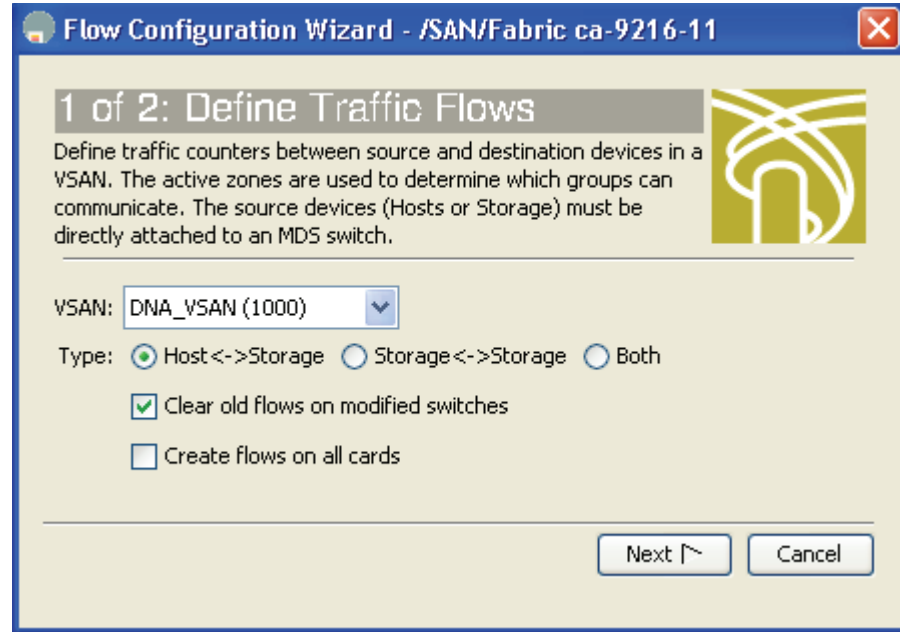
Figure 2-2 Fabric Manager Performance --> Create Flows



This selection launches the Define Traffic Flows screen shown in Figure 2-3. From this screen, select the types of flows to be gathered. You can gather flows from Host to Storage, Storage to Host, Storage to Storage, or All flows.

In [Figure 2-3](#), all flows on VSAN 1000 (DNA\_VSAN) are being set up for data collection. To change the VSAN selected, click the drop-down arrow to see and select configured VSANs in the fabric.

Figure 2-3 Define Traffic Flows



**Step 2** Click **Next** to see all the possible flows based on your selections in [Figure 2-3](#).

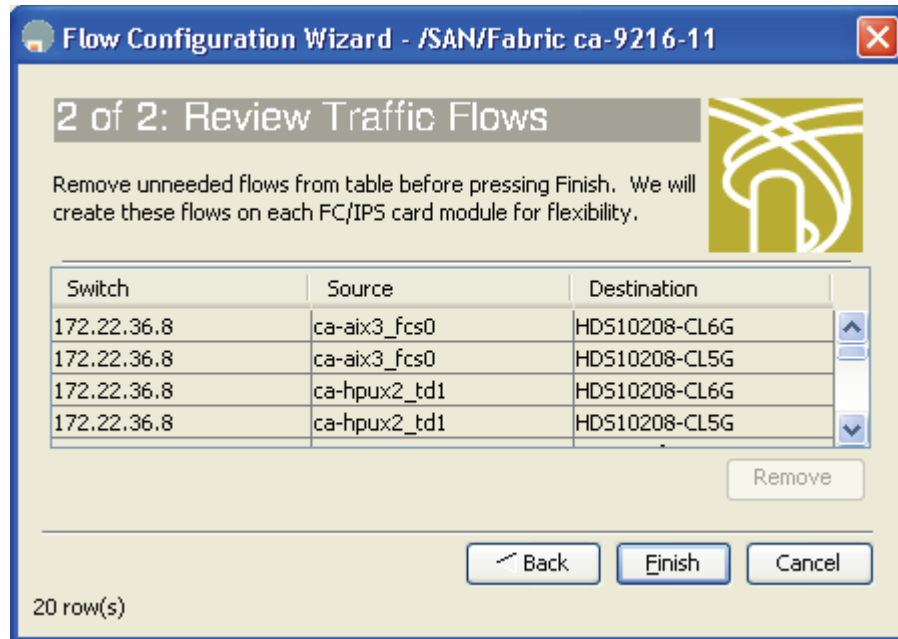
**Step 3** The Review Traffic Flows screen shown in [Figure 2-4](#) lists all possible flows in VSAN 1000 (PM\_VSAN). Remove any unneeded flows.



**Note**

If you are using device aliases, the device alias will be displayed instead of the world-wide name (WWN).

Figure 2-4 Review Traffic Flows



- Step 4** Click **Finish** to create the flows for all the entries listed in the Review Traffic Flows on the appropriate switches.

Repeat this procedure for every VSAN in which you want flows to be created. Or you can add new devices to a VSAN.

In Cisco SAN-OS 3.x, creating collections was moved from the Fabric Manager Client to the Performance Manager web GUI.

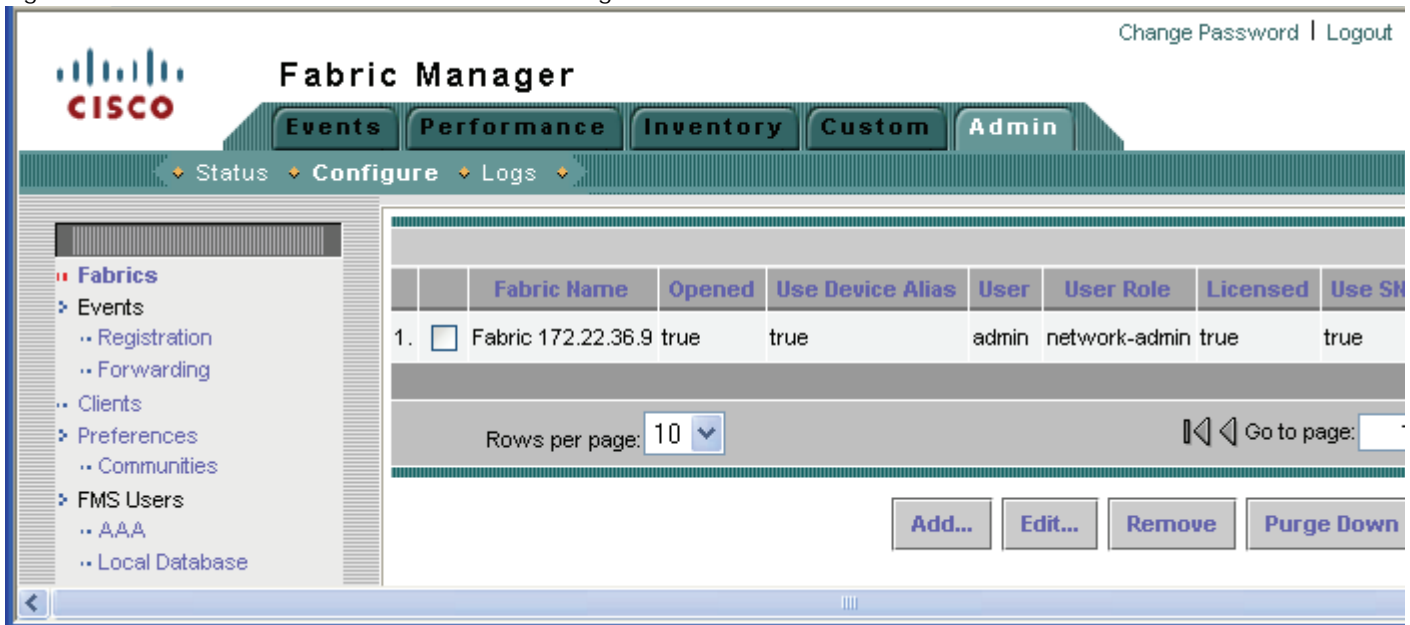
## Creating a Collection in Performance Manager

Perform the following procedure after creating flows in the recipe “[Creating Flows Within Fabric Manager](#)” on page 8. The procedure instructs Performance Manager to begin to populate its database with performance data for flows, end devices, and ISLs.

To create a collection, follow these steps:

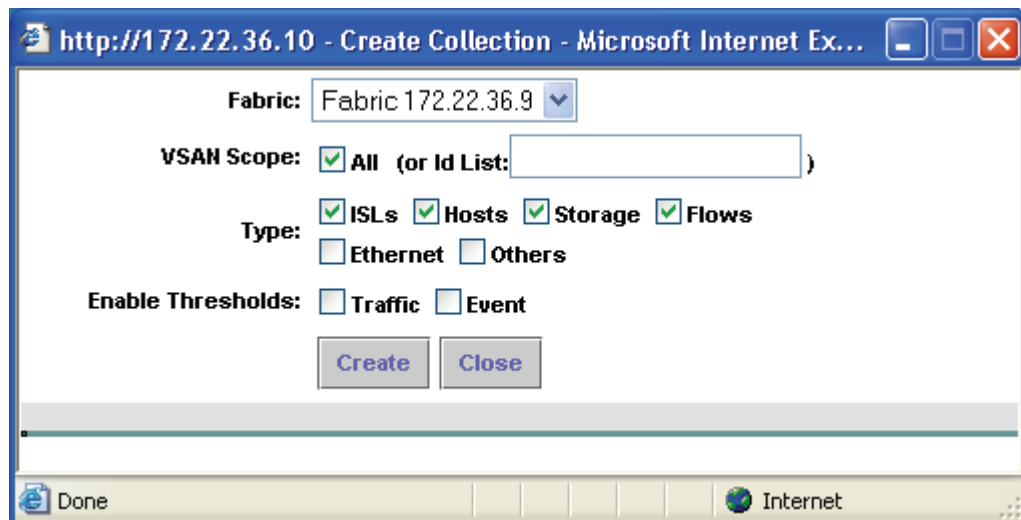
- Step 1** Log into Performance Manager by pointing your web browser at the Fabric Manager Server.
- Step 2** Log into Performance Manager. The default username/password is admin/password.
- Step 3** Select the **Admin** tab.
- Step 4** Select **Configure**.

Figure 2-5 Fabrics Listed in Performance Manager



- Step 5** Ensure that all the fabrics that this Fabric Manager Server is continuously monitoring are listed in Figure 2-5.
- Step 6** On the left menu, under Performance, select **Collections**.
- Step 7** Click the **Add...** button.
- Step 8** In the resulting dialog box, Figure 2-6, select the fabric for which you want to create a collection, the VSANs to monitor, and the type of data you want to collect. At a minimum, ISLs, Hosts, Storage and Flows should be checked.

Figure 2-6 Create Collection



- Step 9** Click **Create**.



**Note**

If you need to create multiple collections because of having multiple fabrics, do not restart Performance Manager when you are prompted. Instead, create the rest of your collections and then restart Performance Manager.

- Step 10** Once all the collections are created, click **Close** to dismiss the Create Collections dialog box.
- At this point, the Fabric Manager Server is starting to collect performance data, on the collection(s) you defined. It can take some time for the flows to start to appear in the web page.

Figure 2-7 List Collections

Showing 1-1 of 1 records

|    | Fabric                                      | VSAN Scope | ISLs | Hosts | Storage | Flows | Ethernet | Others | Traffic Threshold | Event Threshold |
|----|---------------------------------------------|------------|------|-------|---------|-------|----------|--------|-------------------|-----------------|
| 1. | <input type="checkbox"/> Fabric 172.22.36.9 | all        | true | true  | true    | true  | false    | false  | false             | false           |

Rows per page: 10 Go to page: 1 of 1 Pages Go

Add...Remove

Depending on the size of the fabric and the number of objects to be collected (hosts, storage and ISLs), it can take some time for the performance data to appear in the various web pages. By default, the shortest period of time collected is five minutes.



## CHAPTER 3

# Security and Access Management

---

Due to the criticality of the role of the Storage Area Network (SAN), care must be taken to prevent end devices from accessing each other's storage. However, switch security itself is often overlooked, such that in many environments default usernames and passwords are used or all storage administrators use the same user account to perform their tasks, which makes it impossible to create an accounting log of who did what task and at what time.

Switches are no different from servers in that they must adhere to an Authentication, Authorization, and Accounting (AAA) policy. This AAA policy can be further defined as:

- **Authentication**—Is this user allowed to access a resource?
- **Authorization**—What level of rights or privileges does the user have?
- **Accounting**—The user's activity is logged on the resource.

An MDS switch should be a self-defending resource, such that it must be able to prevent access from unauthorized users and prevent authorized users from making configuration mistakes. With the critical role of the SAN in the larger picture of the data center, small SAN mistakes can lead to much larger impacting events.

Within an organization, there are many different roles that work with the SAN. While some just monitor it from within a Network Operations Center (NOC) and do not make changes, others perform the daily routine of enabling servers to access their storage with zones, while still others may be creating VSANs when a new switch is deployed. It is important to identify these different roles within your organization and create a security policy, enforced by the MDS, that enables a user to have enough privileges on an MDS switch to perform their job. For example, an engineer within the NOC should be assigned read-only access to an MDS switch, because their job does not require making configuration changes.

Identifying and using the right features and functionality of the MDS switch in creating a secure deployment within your environment can significantly increase uptime.

This chapter provides recipes for managing users and their accounts. In MDS SAN-OS versions before SAN-OS Release 2.0, a separate account was required for both SNMP and CLI access. Starting with SAN-OS Release 2.0(1) and continuing in SAN-OS Release 3.0, a single username grants access to both CLI and SNMP.



### Note

- A new switch that has SAN-OS Release 2.x or higher preinstalled does not have an existing admin password. You have to choose one the first time you run the setup script. Existing accounts are not forced to change passwords.
- Upgrading from SAN-OS Release 1.x to 2.x results in the 1.x CLI password being applied to both CLI and SNMP accounts.

- SAN-OS Release 2.x enforces strong passwords for all accounts created after installing and upgrading to SAN-OS Release 2.x. A strong password must have the following characteristics:
  - At least eight characters long
  - Not too many consecutive characters
  - Not too many repeated characters
  - No easily-guessed dictionary words

**Tip**

- Use the admin account only during initial setup. After setup, create other user accounts. Each administrator should have their own individual account.
- Always change the admin password from the factory default value.
- Grant users the minimum rights or abilities needed for their job function.
- Implementing TACACS+ (see [Configuring TACACS+ with Cisco SecureACS, page 3-8](#)) eliminates the need for password recovery on the switch.

## Creating a User Role

The MDS switch has two default roles: network-admin and network-operator. Network-admin has write privileges to all parts of the switch, while network-operator has read-only access to the switch. Later, you may want to create a user with write access to only specific areas of the switch (see the example in the “[Creating a Role with Device Manager](#)” section on page 3-2).

The MDS switch has two predefined roles:

- **Network-admin** is the role assigned to an administrator. A network-admin can perform any modification to an MDS switch. There are no restrictions on this user.
- **Network-operator** is a read-only role. A network operator cannot make modifications to the switch.

**Tip**

- Provide each user with a role that provides them with the minimum abilities needed.
- Use the network-operator role for users who do not need to modify the switch.
- VSAN- based roles allow administrators to have complete control over their VSANs while having read-only or no access to other VSANs.

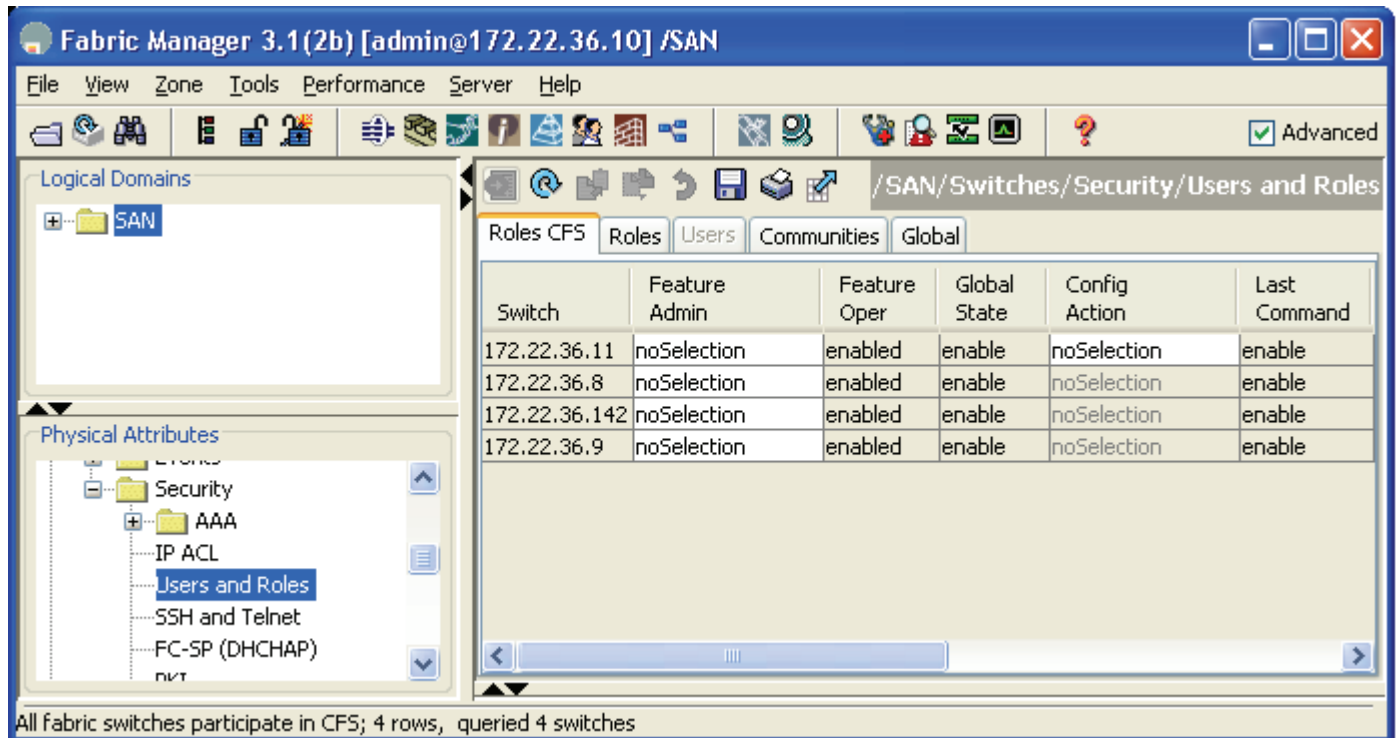
## Creating a Role with Device Manager

This example shows how to create a role with the ability to modify only the zoning configuration on the switch. To create this role, follow these steps:



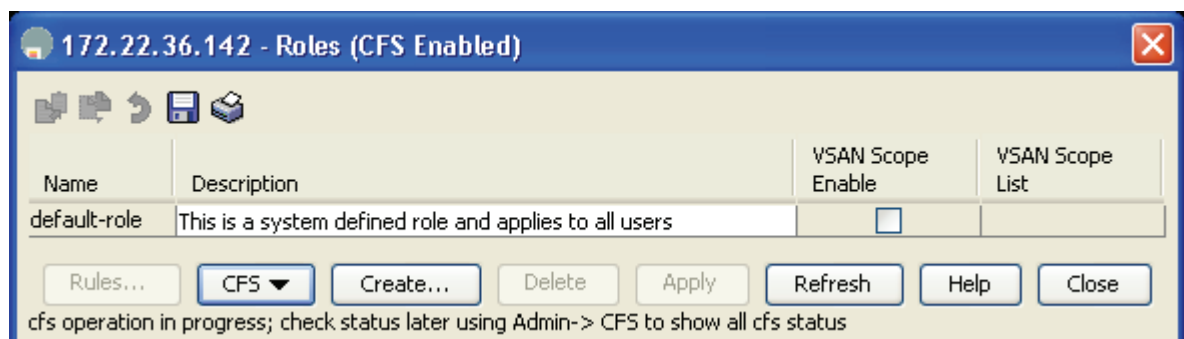
- Step 1** In Fabric Manager, enable role distribution for all switches in the fabric. In the **Physical Attributes** pane, by choose **Switches > Security > Users and Roles** (see [Figure 3-1](#)). In the Feature Admin column, change the cells to **enable**, and then click **Apply**. For the rest of this procedure, Fabric Manager is no longer needed.

Figure 3-1 Enabling CFS Distribution for Roles



- Step 2** Open Device Manager (DM) from any of the switches that are enabled for Cisco Fabric Services (CFS) distribution of roles.
- Step 3** Choose **Security > Roles**. You see the screen in [Figure 3-2](#).
- Step 4** When Device Manager informs you that CFS is enabled, click **Continue**.

Figure 3-2 Initial Roles Screen



- Step 5** Click **Create**.

Figure 3-3 Create Common Roles

**Step 6** Provide a name and description (no spaces) for the role (see [Figure 3-3](#)).

**Note**

This example does not specify a VSAN scope, but you could optionally create a VSAN scope limiting this specific role to a subset of VSANs. For example, a zoning admin role could be created for zone admins who can only modify VSANs 1–10. Adding a VSAN scope requires the installation of the Enterprise Package license.

**Step 7** Click **Rules...** to define what this role can and cannot do within the optional VSAN scope (see [Figure 3-3](#)). The Create Common Rules screen shown in [Figure 3-4](#) appears.

Figure 3-4 Create Common Role Rules

| CLI Command          | Clear                               | Config                              | Debug                               | Show                                | Exec                                |
|----------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| zoneset              | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| zone-attribute-group | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/>            |
| zone                 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| wwn                  | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/>            |
| write                | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            |
| vsan                 | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/>            |
| vrrp                 | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input checked="" type="checkbox"/> | <input type="checkbox"/>            |
| wni                  | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            | <input type="checkbox"/>            |

**Step 8** Choose the CLI Command column to sort the table by CLI command.

**Step 9** Check the **Show** check box at the top of the screen to enable show for all commands.

**Step 10** Scroll down and check all of the **zone**, **zone-attribute-group**, and **zoneset** options.

**Step 11** Check **copy** so that the zoning admin can save the configuration.

- Step 12** Click **Apply** (see [Figure 3-4](#)).
- Step 13** Click **Create** (see [Figure 3-5](#)). This saves the role configuration to the CFS pending database. Not until the CFS **commit** command is executed will this role become part of the running configuration of the switches in the fabric.

Figure 3-5 Create Common Roles

172.22.36.142 - Create Rol...

Name: ZoningAdmin

Description: Zoneset\_Administrator

**Vsan Scope**

☐ Enable

List:

Rules... Create Close

- Step 14** Click **Close** (see [Figure 3-5](#)) to return to the original Roles screen (see [Figure 3-6](#)). The ZoningAdmin role now exists only in the pending database.

Figure 3-6 Display Roles

172.22.36.142 - Roles (CFS Enabled)

| Name         | Description                                            | VSAN Scope Enable                   | VSAN Scope List |
|--------------|--------------------------------------------------------|-------------------------------------|-----------------|
| default-role | This is a system defined role and applies to all users | <input type="checkbox"/>            |                 |
| ZoningAdmin  | Zoneset_Administrator                                  | <input checked="" type="checkbox"/> |                 |

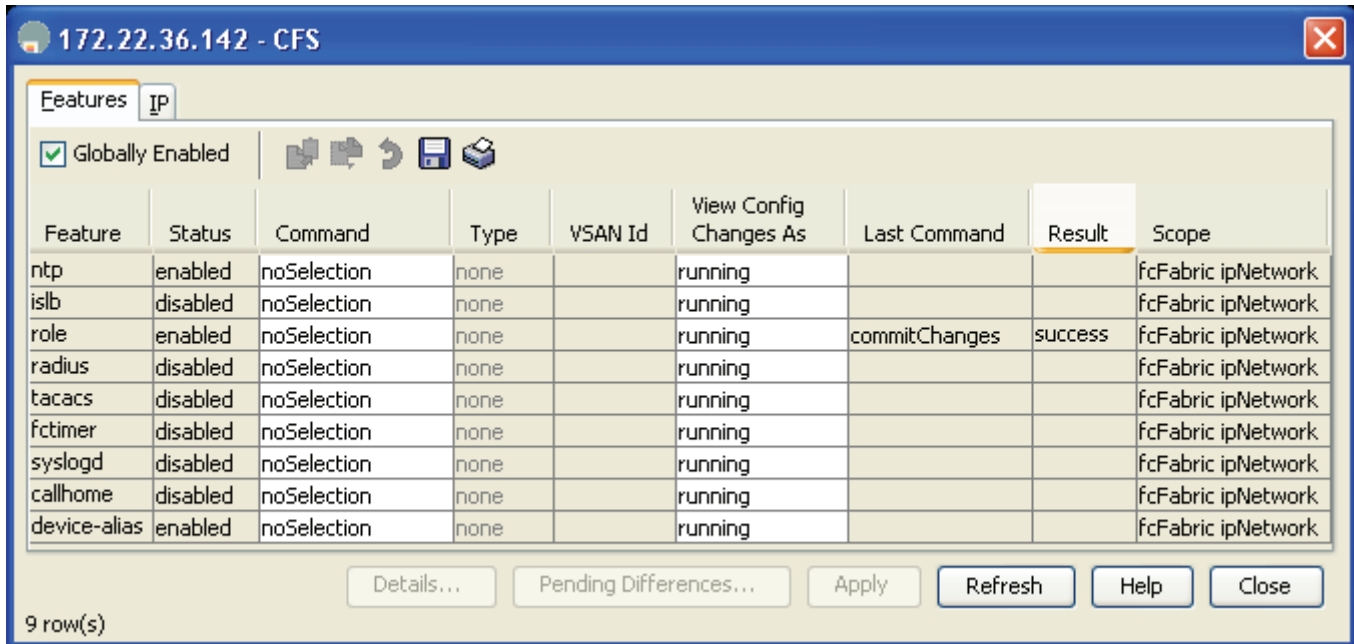
Rules... CFS ▼ Create... Delete Apply Refresh Help Close

cfs operation in progress; check status later using Admin-> CFS to show all cfs status

- Step 15** Commit the changes by expanding **CFS** (see [Figure 3-6](#)) and selecting **Commit**.
- To abort the changes and flush the pending database, expand **CFS** (see [Figure 3-6](#)) and select **Abort**.

To see the status of the CFS operation, from the main window click **Admin > CFS**. You see the Created Roles screen shown in Figure 3-7.

Figure 3-7 Created Roles



## Creating a Role with CLI

The CFS CLI can distribute roles throughout the physical fabric to provide access to one or more switches.

To use the CLI to distribute roles throughout the physical fabric, follow these steps:

- Step 1** Enter configuration mode and enable CFS distribution (**role distribute**) for roles on each switch that should receive this role. This is the only step that must be done individually for all switches. The other steps do not need to be repeated on each switch.

```
switch# conf t
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# role distribute
```

- Step 2** Create the ZoningAdmin role with the **role name** command, describe the role with **description** command, and add rules to it with **rule** command.

```
switch(config)#role name ZoningAdmin
switch(config-role)# description Zoneset_Administrator
switch(config-role)# rule 1 permit show
switch(config-role)# rule 2 permit config feature zoneset
switch(config-role)# rule 3 permit exec feature zoneset
switch(config-role)# rule 4 permit clear feature zone
switch(config-role)# rule 5 permit config feature zone
switch(config-role)# rule 6 permit debug feature zone
switch(config-role)# rule 7 permit exec feature zone
```

```
switch(config-role)# rule 8 permit exec feature copy
```

- Step 3** CFS commit the role and distribute it to the other switches with the **role commit** command.

```
switch(config)# role commit
```

- Step 4** (Optional) Create a user `zoning_user` and assign him the new `ZoningAdmin` role.

```
switch# config terminal
switch(config)# username zoning_user password g0s0x456 role ZoningAdmin
```

- Step 5** Save the configuration fabric-wide with the **copy running-config startup-config fabric** command.

```
switch(config)# copy running-config startup-config fabric
[#####] 100%
```

## Creating User Accounts

To access an MDS switch, create at least one username. This section explains how to create a user.



**Tip**

Cisco recommends using TACACS+ servers to manage user accounts instead of managing them locally, especially if you have multiple users and multiple MDS switches. See [“Configuring TACACS+ with Cisco SecureACS” on page 8](#).

## User Accounts Through Command-Line

To create a user from the CLI, follow these steps:

- Step 1** Enter configuration submode with the **config terminal** command.

```
switch# config terminal
```

- Step 2** Create the user with the syntax **username username password password role role**.

```
switch(config)# username fedona password sox2004ch@mps role network-admin
```

At this point, the user “fedona” can access the switch with the password `sox2004ch@mps`. Access can be with console, SSH, Telnet, or SNMP.

## User Accounts Through Fabric Manager

To create a user from Fabric Manager, follow these steps:



**Note**

Creating a user with Fabric Manager *requires* that you previously logged into Fabric Manager using a privacy password.

- Step 1** In the Physical Attributes pane, select **Switches**.

- Step 2** Select **Security**.
- Step 3** In the top pane, select the **Users** tab. (If the Users tab is grayed out, select the Roles tab first, then select the Users tab.)
- Step 4** Click **Create Row...** See [Figure 3-8](#).

Figure 3-8 Creating a User in Fabric Manager

The screenshot shows a 'Create User' dialog box with the following fields and options:

- Switches:** A list box containing four IP addresses: 172.22.36.11, 172.22.36.142, 172.22.36.8, and 172.22.36.9. All are checked.
- New User:** A text field containing 'fedona'.
- Password:** A text field containing '\*\*\*\*\*'.
- Roles:** A list box containing four roles: default-role, ZoningAdmin, network-admin, and network-operator. 'ZoningAdmin' is checked.
- Digest:** Radio buttons for MD5 (selected), SHA, and None.
- Encryption:** Radio buttons for DES (selected), AES, and None.
- Optional:** A section with two text fields: 'ExpiryDate: (eg. yyyy/mm/dd)' and 'SshKeyFilename: ([bootflash:|volatile:])'.
- Buttons:** 'Create' and 'Close' buttons at the bottom right.

- Step 5** Select the switches on which you want to create the user.
- Step 6** Fill in the username and password. The password will not be displayed.
- Step 7** Select the roles for this user.
- Step 8** Select **Create...**

## Configuring TACACS+ with Cisco SecureACS

Cisco's SecureACS product enhances MDS switch management security and provides centralized authentication, authorization, and accounting of users.

**Tip**

We recommend that a TACACS+ server be used for authentication, authorization and accounting. With TACACS+ implemented, you do not have to perform password recovery on the MDS switch.

## Authentication and Authorization with TACACS+

Configuring an MDS switch to use TACACS+ allows centralized account management of the switch. This centralized management allows an administrator to not have to create and maintain usernames and passwords on individual switches. The SecureACS server provides the authentication to a switch login as well as role assignment for the user. A shared secret key is used to provide encryption and authentication between the TACACS+ client (MDS-9500) and the TACACS+ server (Cisco SecureACS).

In this example, these resources are used:

- The TACACS+ server is Cisco Secure ACS v4.0
- The switch's IP address is 172.22.36.142.
- The TACACS+ server's IP address is 172.22.36.10.
- The TACACS+ shared secret key is WarEagle.

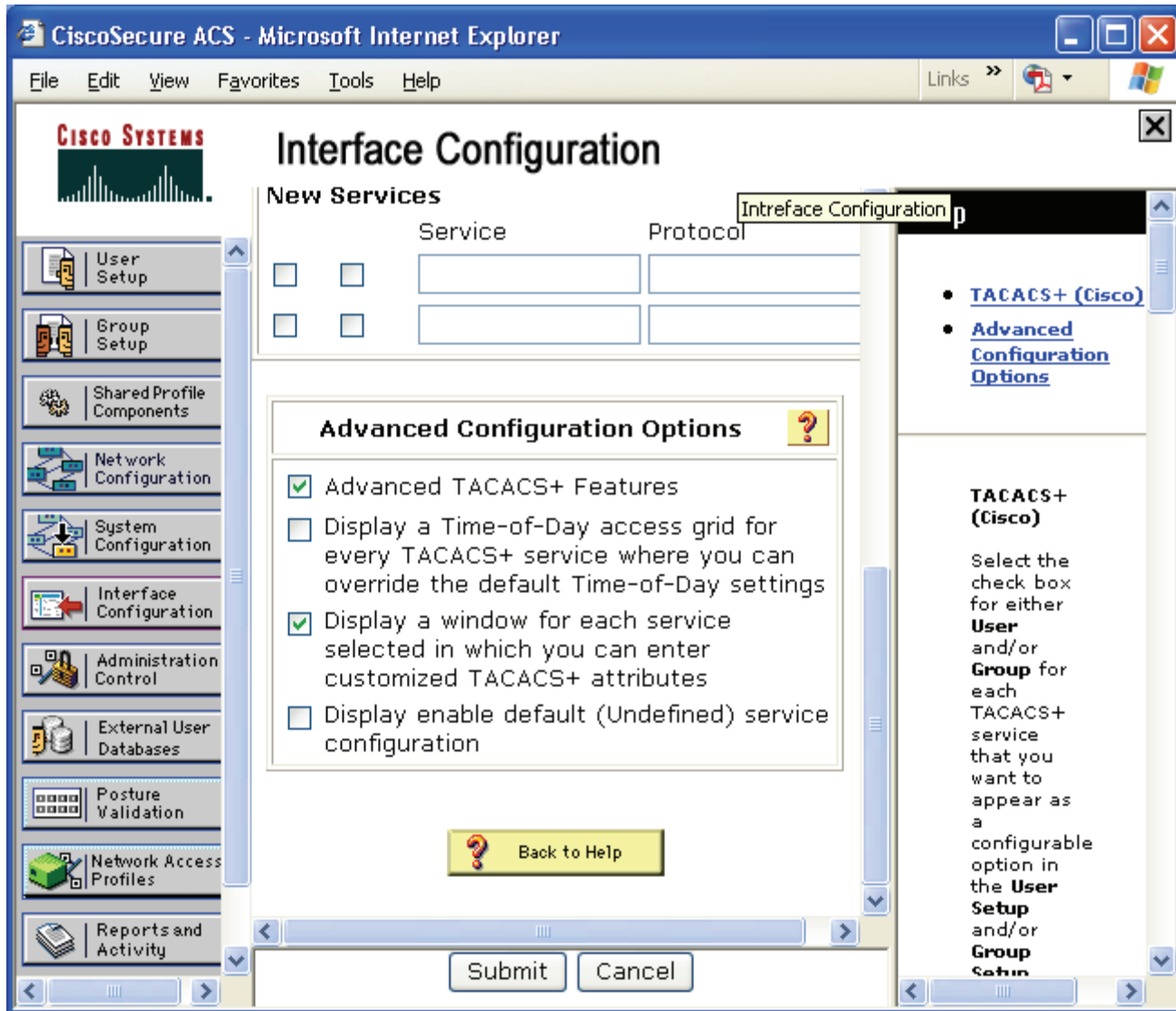
## Configuring the SecureACS Server

Before you configure the switch, configure the SecureACS server with Cisco SecureACS.

To configure the SecureACS server with Cisco SecureACS, follow these steps:

- 
- Step 1** Configure SecureACS to allow modification of advanced TACACS+ settings.
- a. On the left pane of the main screen, choose **Interface Configuration**.
  - b. Select the link **TACACS+ (Cisco IOS)**. You see the Interface Configuration screen shown in [Figure 3-9](#).

Figure 3-9 SecureACS Configure Display



- c. Check both **Advanced TACACS+ Features** and **Display a window...** attributes.
- d. Click **Submit** to save the changes.



- Step 2** Use SecureACS to define the MDS-9506 switch for the TACACS+ server. This allows the MDS switch to be authenticated by the server.
- In the left pane, click **Network Configuration > Add Entry**. You then see the Network Configuration screen shown in [Figure 3-10](#).
  - Provide the MDS switch IP address **172.22.36.142** and the shared secret key **WarEagle** as shown in [Figure 3-10](#).

Figure 3-10 SecureACS Client Setup

The screenshot shows the CiscoSecure ACS web interface in Microsoft Internet Explorer. The browser window title is "CiscoSecure ACS - Microsoft Internet Explorer". The address bar shows "File Edit View Favorites Tools Help" and "Links >>". The main content area is titled "Network Configuration" and "Edit". Below this is a large heading "Add AAA Client". The form contains the following fields and options:

- AAA Client Hostname: 172.22.36.142
- AAA Client IP Address: (empty)
- Key: WarEagle
- Authenticate Using: TACACS+ (Cisco IOS) (dropdown menu)
- ☐ Single Connect TACACS+ AAA Client (Record stop in accounting on failure).

- Click **Submit** to save the information.

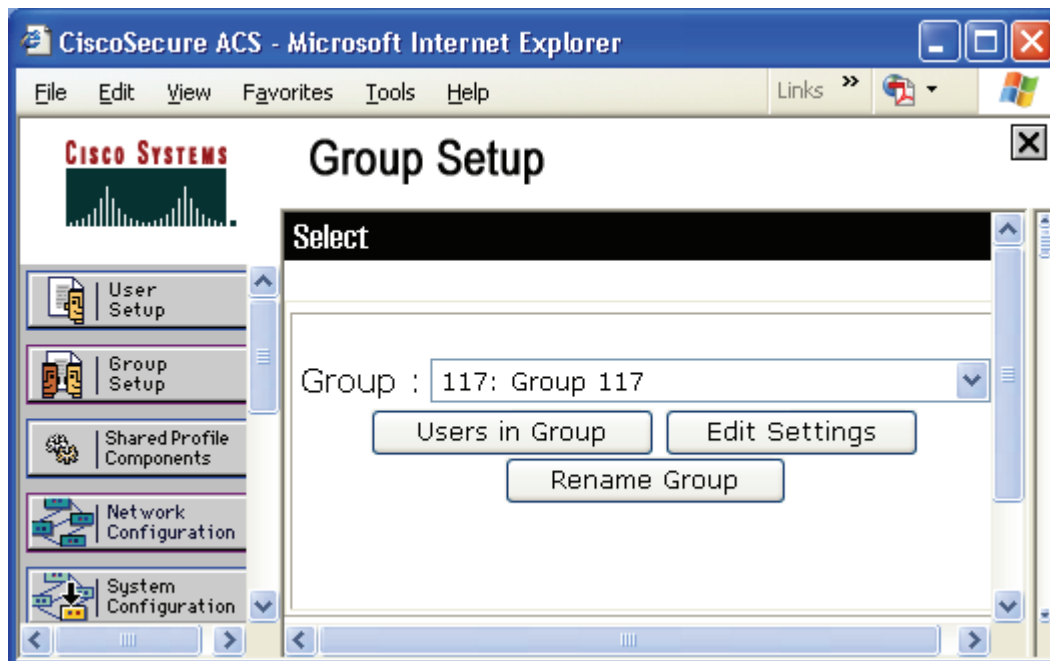
- Step 3** Define a group so you can assign the same role to multiple users without having to modify the attributes of each user individually.

**Note**

The name of the TACACS+ group is not related to the role of the MDS users. The role that is assigned to the MDS users is a **property** of the TACACS+ group or the TACACS+ user.

- a. In the left pane, click **Group Setup**. The Group Setup screen appears (see Figure 3-11).

Figure 3-11 SecureACS: Group Setup



- b. Select an available group and click **Rename Group** (see Figure 3-11).  
c. Enter a new name for this group.

**Tip**

Use the same SecureACS group name as the role name to ease creation of TACACS+-based users. Names such as MDS\_Zoning\_Admin can aid in determining what the role is without looking at the role's rules.

- d. Click **Submit** to save the name change.  
e. Select the newly renamed group and click **Edit Settings**.  
f. Scroll to the section labeled TACACS+ Settings, then check the **Shell** and **Custom attributes** (see Figure 3-12).

Figure 3-12 SecureACS Adding MDS Switch Role



- g. In the Custom attributes field, enter the av-pair string corresponding to the role defined on the switch for users. The syntax is **cisco-av-pair=shell:roles="<role>."** (see Figure 3-12)
- h. Click **Submit + Restart** (see Figure 3-12) to save and apply the configuration.

**Step 4** Define a user using these steps:

- a. Click **User Setup** in the left pane. You then see the User Setup screen.
- b. Enter a new or existing user name.
- c. Click **Add/Edit**.

- d. Provide the information for the fields **Password**, **Confirm Password** and **Group to which user is assigned** (see Figure 3-13).

Figure 3-13 SecureACS Creating TACACS+ User

The screenshot displays the 'User Setup' page of the CiscoSecure ACS web interface. The left sidebar contains various configuration options, with 'User Setup' currently selected. The main panel shows the 'ACS Internal Database' as the selected user source. Below this, there are input fields for 'Password' and 'Confirm Password', both masked with dots. A checkbox labeled 'Separate (CHAP/MS-CHAP/ARAP)' is present and unchecked. Further down, there are additional 'Password' and 'Confirm Password' fields. A descriptive text block explains that a separate CHAP password is useful for token servers. At the bottom, a dropdown menu for 'Group to which the user is assigned' is set to 'MDS\_Zoning\_Admin'. 'Submit' and 'Cancel' buttons are located at the bottom right of the form.

Configuration of the SecureACS server is complete. Next, configure the MDS switch itself.

## Configuring TACACS+ on the MDS Switch

You can configure an MDS switch from the CLI or using SNMP.

To configure the switch from the CLI, follow these steps:

- Step 1** Enter configuration mode, and then enable TACACS+.

```
ca-9506# conf t
```

Enter configuration commands, one per line. End with CNTL/Z.

```
ca-9506(config)# tacacs+ enable
```

- Step 2** Define the TACACS+ server 172.22.36.10 and the corresponding shared secret key WarEagle.

```
ca-9506(config)# tacacs-server host 172.22.36.10 key WarEagle
```

- Step 3** Define a group of authentication servers to use, and then add the TACACS+ server to the group.

```
ca-9506(config)# aaa group server tacacs+ tacacs-group1
ca-9506(config-tacacs+)# server 172.22.36.10
```

- Step 4** Define the authentication method for the switch's Telnet, SSH, or SNMP access.

```
ca-9506(config)# aaa authentication login default group tacacs-group1
```

- Step 5** Use **show** commands to display and check the configuration:

```
ca-9506# show tacacs-server

timeout value:5
total number of servers:1

following TACACS+ servers are configured:
 172.22.36.10:
 available on port:49
 TACACS+ shared secret:*****
ca-9506# show aaa authentication
 default: group tacacs-group1
 console: local
 iscsi: local
 dhchap: local

ca-9506# show user-account
user:admin
 this user account has no expiry date
 roles:network-admin

user:fedona
 expires on Fri Jun 18 23:59:59 2004
 roles:ZoningAdmin
account created through REMOTE authentication
Local login not possible
```



**Note**

The user seth is not available locally on the switch and yet is a member of the group/role network-admin. This means seth was authenticated by the TACACS server and not by the switch.

## Accounting with TACACS+

Cisco's SecureACS server can provide a command history of users and their actions. This information is similar to that provided by the CLI command **show accounting log**. However, by placing the information on a remote system, the logs can be independently examined and are available if the switch is inaccessible. This configuration builds upon the configuration defined in [Authentication and Authorization with TACACS+](#), page 3-9.

## Configuring the MDS Switch to Use TACACS Accounting

Because this procedure builds on the configuration defined in [Authentication and Authorization with TACACS+, page 3-9](#), only small modifications need to be made.

To configure the switch to use a TACACS+ server for accounting, follow these steps:

- 
- Step 1** Enter configuration mode.
- Step 2** Configure the switch to use the tacacs-group1 server group. The local keyword indicates local logging on the switch if all servers listed in the server group are unavailable. If the server group is available, commands and events are **not** logged locally.

```
switch# conf t
switch(config)# aaa accounting default group tacacs-group1 local
```

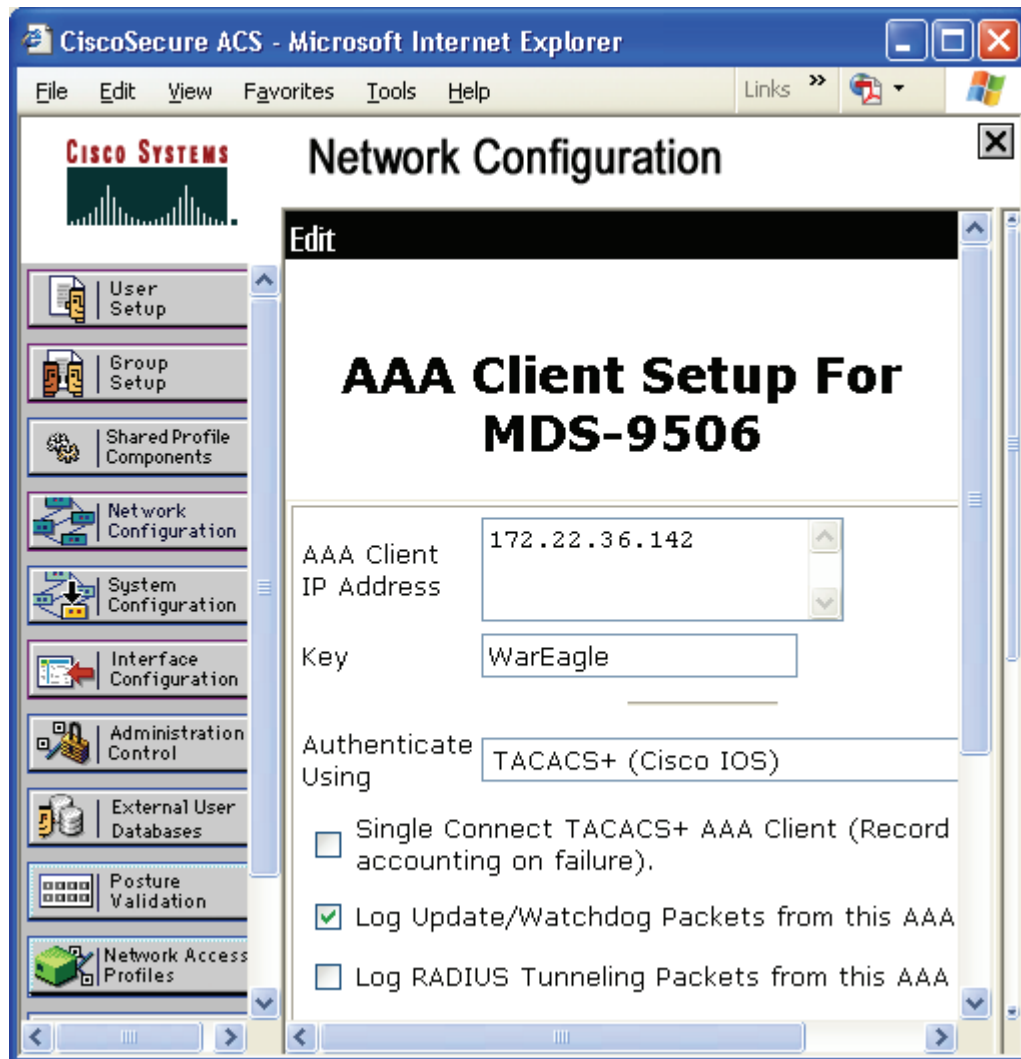
---

## Configuring SecureACS to Receive TACACS+ Accounting

To configure SecureACS to receive TACACS+ accounting, follow these steps:

- 
- Step 1** Configure the SecureACS server to monitor Update/Watchdog packets by modifying the client configuration.
- In the SecureACS left pane, click **Network Configuration** (see [Figure 3-14](#)).
  - Select the client to be modified.
  - Check the **Log Update/Watchdog Packets from this AAA Client** check box (see [Figure 3-14](#)).
  - Click **Submit**.

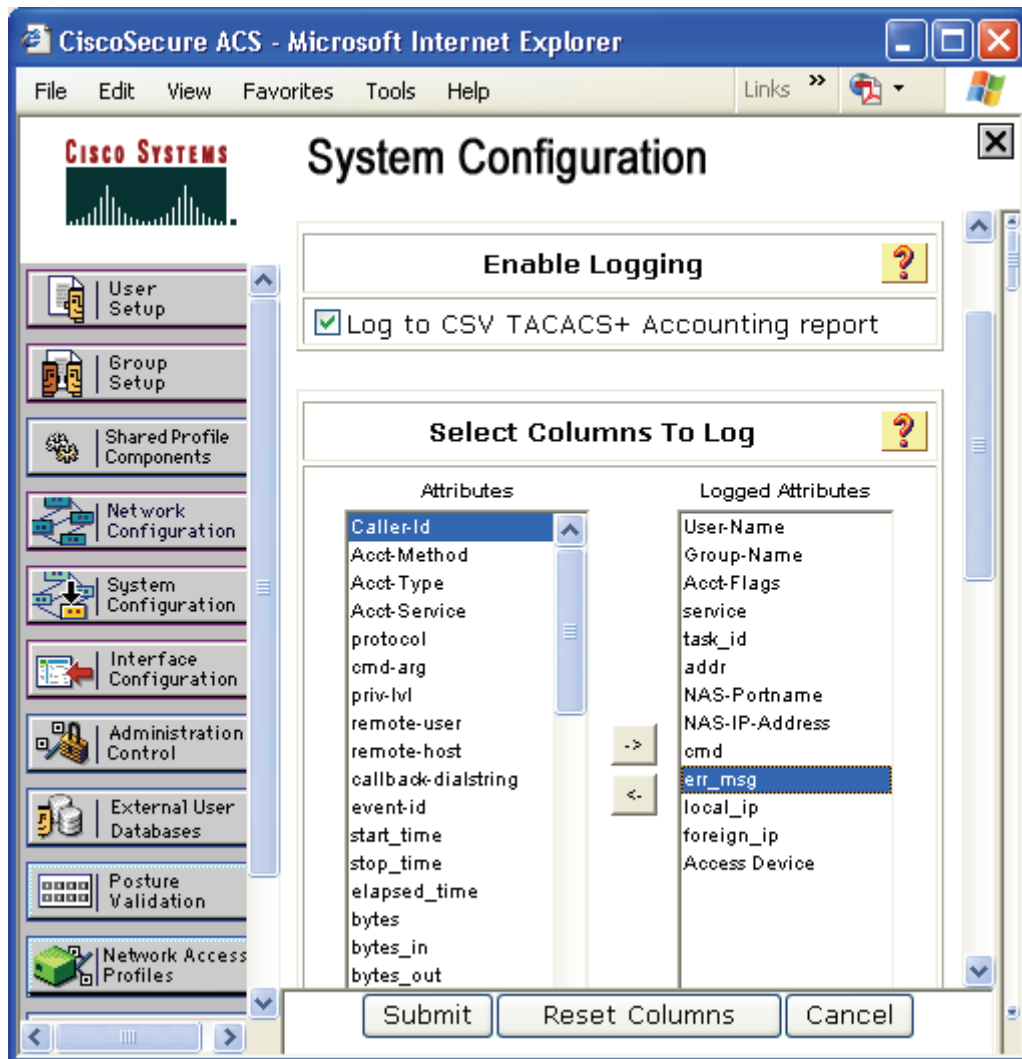
Figure 3-14 Enabling Accounting on the SecureACS server



**Step 2** Configure SecureACS to display commands.

- a. Click **System Configuration** in the left pane.
- b. Click **Logging**.
- c. Select **CSV TACACS+ Accounting**.
- d. Add the column **err\_msg**.
- e. Check the **Log to CSV TACACS+ Accounting report** box (see Figure 3-15).
- f. Click **submit**.

Figure 3-15 Add MDS Command Logging to the Report



**Step 3** View the accounting report.

- a. Click **Reports and Activity** in the left pane (see [Figure 3-15](#)).
- b. Select **TACACS+ Accounting**.
- c. In the right pane, select the day to view (see the result in [Figure 3-16](#)).

The current day is called **TACACS+ Accounting active.csv**.



Figure 3-16 SecureACS Accounting Log

| Date       | Time     | User-Name | Group-Name         | Acct-Flags | service | task_id | NAS-IP-Address | err_msg                                                              | Access Device |
|------------|----------|-----------|--------------------|------------|---------|---------|----------------|----------------------------------------------------------------------|---------------|
| 06/08/2007 | 16:27:47 | admin     | MDS: network-admin | watchdog   | none    | 398550  | 172.22.36.142  | role committed configuration changes done by user admin@172.22.36.11 | MDS-951       |

## Providing Password-Free Access Using SSH

You can allow switch access with no password from automated scripts or agents. Providing a null password or hard-coding the password into the script or agent could be considered a weak security practice. However, using the private/public key infrastructure of SSH maintains a secure environment. SSH uses a private/public key exchange; the switch knows only the public key while the host knows both the public and private keys. Access is only granted if the user comes from a host that knows both the public and private keys.

This procedure includes creating the appropriate key on a host, then adding the key to a new read-only (network-operator) user.



### Tip

Assign password-free logons to either a read-only role like network-operator or to a role with a minimal set of privileges.



### Caution

Having only the public key does not trigger the switch to grant access. The private key must also be on the host. Treat the private key like a password.

To create a key on a host and then add the key to a read-only user, follow these steps:

#### Step 1 Create an SSH RSA1 public/private key on the host.

```
$ /usr/bin/ssh-keygen -t rsa1
Generating public/private rsa1 key pair.
Enter file in which to save the key (/users/testuser/.ssh/identity):
/users/testuser/.ssh/identity already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /users/testuser/.ssh/identity.
Your public key has been saved in /users/testuser/.ssh/identity.pub.
The key fingerprint is:
c2:4d:6d:26:21:9d:79:9b:c3:86:dc:a5:07:d2:62:d4 testuser@host
```

On the host, the file **/users/testuser/.ssh/identity.pub** is the SSH public key that is encrypted using the RSA1 algorithm. The contents of this file are used in the creation of the MDS switch user. In this example, the file looks like this:

```
$ cat /users/testuser/.ssh/identity.pub
```

```

1024 35
139198677264732164858153476357747926024656548233745027006381178621992083524037906211714241
450436547019604214530354070873624269283640613058470615170649963414635036859628344005142227
886318134122126153182906740418449098047827961768214148936752631482459130056603268404256522
191410368204629699075809390037814979061 testuser@host

```

**Step 2** On the switch, create all of the SSH keys, even though in this case the client is using RSA1.

```

172.22.36.11# conf t
Enter configuration commands, one per line. End with CNTL/Z.

172.22.36.11(config)# ssh key rsa1
generating rsa1 key(1024 bits).....
generated rsa1 key

ca-9506(config)# ssh key dsa
generating dsa key(1024 bits).....
generated dsa key

ca-9506(config)# ssh key rsa
generating rsa key(1024 bits).....
generated rsa key

```

**Step 3** Enable SSH on the switch.

```
172.22.36.11(config)# ssh server enable
```

**Step 4** On the switch, create the user, pasting in the contents of the identity.pub file after the SSH key parameter.

```

172.22.36.11# conf t
Enter configuration commands, one per line. End with CNTL/Z.
172.22.36.11(config)# username testuser role network-operator
warning: password for user:testuser not set. S/he cannot login currently
172.22.36.11(config)# username testuser sshkey 1024 35
139198677264732164858153476357747926024656548233745027006381178621992083524037906211714241
450436547019604214530354070873624269283640613058470615170649963414635036859628344005142227
886318134122126153182906740418449098047827961768214148936752631482459130056603268404256522
191410368204629699075809390037814979061 testuser@host
172.22.36.11(config)# end

```

**Step 5** Examine the configuration of the user with the **show user-account** command.

```

172.22.36.11# show user-account testuser
user: testuser
 this user account has no expiry date
 roles:network-operator
no password set. Local login not allowed
Remote login through RADIUS/TACACS+ is possible
 ssh public key: 1024 35 139198677264732164858153476357747926024656548233
74502700638117862199208352403790621171424145043654701960421453035407087362426928
36406130584706151706499634146350368596283440051422278863181341221261531829067404
18449098047827961768214148936752631482459130056603268404256522191410368204629699
075809390037814979061 testuser@host

```

**Step 6** Test the login process from the host with the **testuser** command.

```

$ ssh testuser@172.22.36.11
Warning: Remote host denied X11 forwarding.
Cisco Storage Area Networking Operating System (SAN-OS) Software
TAC support: http://www.cisco.com/tac
Copyright (c) 2002-2004, Cisco Systems, Inc. All rights reserved.
The copyrights to certain works contained herein are owned by
other third parties and are used and distributed under license.
Some parts of this software are covered under the GNU Public
License. A copy of the license is available at

```

```
http://www.gnu.org/licenses/gpl.html.
172.22.36.11#
```

If the same user tries logging in from another host without both the private key file (/users/testuser/.ssh/identity) and the public key file (/users/testuser/.ssh/identity), then access to the switch is denied. The fact that the public key has **testuser@host** included does not tie it to a specific host, but does allow an administrator to determine from which host it was generated.



**Tip**

A simple way to use this feature is to schedule a nightly backup (using cron, for example) for the switch configuration using SSH. The following backup example works as long as the specified user has the privilege to enter the **copy** command:

```
#!/bin/sh
#####
#
/usr/local/bin/backup_mds_config.sh

This is used for a cron entry. No arguments are
allowed in cron. Absolute paths to commands must
be specified to ssh for it to work properly
ssh key exchange must be separately configured
for the account "USER"
#
Adjust the variables for your host and switch
#####

DIR=/mds_config
DATE=`date "+%m%d%y_%H%M%S"`
SWITCH_NAME=beat_bama
FILE=$SWITCH_NAME"_run_cfg_"$DATE
USER=testuser
COMMAND1="copy running-config startup-config"
COMMAND2="show startup-config"

#Copy running to startup-config
/usr/local/bin/ssh -l $USER $SWITCH_NAME $COMMAND1
#Backup MDS config to local file
/usr/local/bin/ssh -l $USER $SWITCH_NAME $COMMAND2 > $DIR/$FILE
```

Set up cron to execute the script. The cron job must be run by the user specified in the script. Configure the crontab for the user. This example runs at 11:00pm every Sunday:

```
#Backup MDS config:
00 23 * * 0 /usr/local/bin/backup_mds_config.sh > /mds_logs/beat_bama1
```

## Disabling the Web Server

On an MDS switch, a built-in web server provides a method of downloading and installing Device Manager and Fabric Manager. After installation of these tools, the web server is no longer needed. For configuration and notification, the MDS switch does not rely on HTTP for communication to the switch.

In fact, it is possible to download a specific version of Fabric Manager and Device Manager directly from Cisco.com (<http://www.cisco.com>) without using the web server at all. However, to increase the security of the MDS platform, the web server can be used to block all access to TCP port 80 of the switch.

The IP access control list (ACL) that is set up denies access to a TCP port 80 on the switch but allows access to all other TCP ports.

**Note**

IP ACLs have an implicit DENY ALL added to the end of an ACL, so at least one permit statement must be used. Otherwise, all traffic is denied.

**Tip**

- Whenever using ACLs on the mgmt0 interface, first test it using a switch that has console access, in case a typo occurs.
- The commands in this recipe can be integrated into a script by copying the bold commands and pasting them into a CLI session.

To create an IP ACL filter and apply it to the mgmt0 interface from the CLI, follow these steps:

- Step 1** Enter configuration mode and then create an access list called **disable\_webserver**. The second entry for the access list is a permit any statement.

```
switch1#conf t
switch1(config)#ip access-list disable_webserver deny tcp any any eq port www
switch1(config)#ip access-list disable_webserver permit ip any any
```

- Step 2** Apply the access list to mgmt0.

```
switch1(config)#interface mgmt0
switch1(config-if)#ip access-group disable_webserver in
```



## CHAPTER 4

# Physical Interfaces

---

The MDS switch is a multiprotocol switch. It can host both Fibre Channel and IP (Gigabit Ethernet interfaces) on the same switch. The Fibre Channel interfaces support Fibre Channel as well as FICON. The IP (Gigabit Ethernet ports) support FCIP as well as iSCSI. In this section, various modes and protocol options that are used to configure the Fibre Channel (FC) and Gigabit Ethernet ports are detailed.

The MDS switch supports Generation 1 modules, including the 16-port and 32-port Fibre Channel switching modules, the SSM, the MPS-14/2, and the IPS-8 and IPS-4 storage services modules. The Fibre Channel ports support 1- or 2-Gbps and the Ethernet speed is 1 Gigabit. In the Generation-1 modules, the 16-port modules support 2 Gigabit line rate on all 16 ports. The port can autonegotiate to either 1 Gbps or 2 Gbps. The 32-port modules are oversubscribed 3.3:1. The oversubscription rate is fixed on Generation 1 modules and cannot be changed. See [Figure 4-1](#) for oversubscription rates.

Figure 4-1 Oversubscription of Various Modules

| Line Cards   | Speeds and Over Subscription |                    |                    |                    |
|--------------|------------------------------|--------------------|--------------------|--------------------|
|              | 1 Gbit/sec                   | 2 Gbit/sec         | 4 Gbit/sec         | 10 Gbit/sec        |
| 16 port LC   | 1:1<br>(line rate)           | 1:1<br>(line rate) | Not Supported      | Not Supported      |
| 32 port LC   | 1:1<br>(line rate)           | 3.3:1              | Not Supported      | Not Supported      |
| 14+2 port LC | 1:1<br>(line rate)           | 1:1<br>(line rate) | Not Supported      | Not Supported      |
| 12 port LC   | 1:1<br>(line rate)           | 1:1<br>(line rate) | 1:1<br>(line rate) | Not Supported      |
| 24 port LC   | 1:1<br>(line rate)           | 1:1<br>(line rate) | 2:1                | Not Supported      |
| 48 port LC   | 1:1<br>(line rate)           | 2:1                | 4:1                | Not Supported      |
| 4 port LC    | Not Supported                | Not Supported      | Not Supported      | 1:1<br>(line rate) |

The Generation 2 modules are the 12-port, 24-port and 48-port Fibre Channel switching modules. All these modules support 1-, 2-, and 4-Gbps on all the interfaces. The 12-port module supports 1-, 2-, and 4-Gbps line rate on 12 ports. The 24- and 48-port modules are oversubscribed. The 24-port module handles traffic at a line rate of 1- and 2-Gbps and is 2:1 oversubscribed at 4-Gbps. The 48-port module handles traffic at a line rate of 1-Gbps and is 2:1 oversubscribed at 2-Gbps and 4:1 oversubscribed at 4-Gbps. The 24-port module and the 48- port module allow the user to manage the oversubscription on a per port basis using the rate limiting feature on the 24- and 48-port modules. This feature is not available on the Generation-1 modules.

Figure 4-1 shows the various throughput for the different modules supported on the MDS switches.

The recipes below show how to configure various parameters and modes for different physical ports supported on the MDS switches.

## Configuring Fibre Channel Ports

This section describes how to configure Fibre Channel ports.

### Port Description

A port description provides a plain text description for the interface of a port on a switch. In this example, the Fibre Channel interface fc 1/1 is given the description “storage array 17 port 1.”

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
```

```

mds-switch-1(config)# interface fc 1/1
mds-switch-1(config-if)# switchport description "storage array 17 port 1"
mds-switch-1(config-if)# end
mds-switch-1#

```

## Port Speed

This example sets the port speed for fc 1/1 to either 1-Gbps, 2-Gbps, or an automatically negotiated speed.



### Note

A port can be set to only one speed at a time. The default is autonegotiate

```

mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface fc 1/1
mds-switch-1(config-if)# switchport speed 1000 <- port speed set to 1 Gbits/sec
mds-switch-1(config-if)# switchport speed 2000 <- port speed set to 2 Gbits/sec
mds-switch-1(config-if)# switchport speed 4000 <- port speed set to 4 Gbits/sec
mds-switch-1(config-if)# switchport speed auto <- port speed set to auto negotiate
mds-switch-1(config-if)# exit
mds-switch-1#

```

## Port Mode Auto



### Note

A Fibre Channel port can be set to only one port mode at a time. The default mode is auto on the 12- and 16-port modules and on the MPS-14/2, and FX on the 24-, 32- and 4-port modules.

Setting port mode to auto allows the port to negotiate to either F port mode, FL port mode or E port mode. It cannot negotiate to ST port mode, SD port mode, or TL port mode.

In this example, fc 1/1 is set to auto port mode. This is the default setting for all ports on a 16-port module.

```

mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface fc 1/1
mds-switch-1(config-if)# switchport mode auto
mds-switch-1(config-if)# end
mds-switch-1#

```

## Port Mode E

Setting port mode to E restricts the port to operating as an E port; the port can be either a trunking or nontrunking port, depending on the trunk port mode. E port mode is used when the port talks to another port of a different switch forming an ISL. In this example, fc 1/1 is set to E port mode.

The following example shows the configuration steps for 12- and 16-port modules.

```

mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface fc 1/1
mds-switch-1(config-if)# switchport mode E
mds-switch-1(config-if)# end

```

## Configuring and E Port on a 32-Port Module

On a 32-port module only one port in every quad can function as an E port. To configure this, disable three of the four ports in each quad. Once the three ports have been disabled the remaining port needs to be configured as an E port. The remaining three ports in the quad cannot be used as long as one of the ports in the quad operates as an E port.

The following example shows the configuration steps on a 32-port module.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# interface fc 2/2-4
mds-switch-3(config-if)# shutdown
mds-switch-3(config-if)# interface fc 2/1 switchport mode E
mds-switch-3(config-if)# switchport mode E
mds-switch-3(config-if)# end
mds-switch-3#
```

## Configuring an E Port on 24- and 48-Port Modules

On 24- and 48-port modules, all ports by default use rate-mode as shared. To make one of the ports on a 24- or a 48-port modules an E port, the rate-mode for the port has to be configured as dedicated. Then the port has to be configured as an E port. These two additional steps are required before the ports on these module can function as an E port. The default speed on 24- and 48-port modules is 4 Gbps.

The following example shows the configuration details for 24- and 48-port modules.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# interface fc 2/1
mds-switch-2(config-if)# switchport rate-mode dedicated
mds-switch-2(config-if)# switchport speed 2000
mds-switch-2(config-if)# end
mds-switch-2#
```

## Configuring Trunking E Ports

A trunking port is used to carry VSAN-enabled frames between switches. The following section shows various configuration options for a trunking port.



### Note

These same commands apply to port channels. Specify the port channel interface **int PortChannel 1** rather than an individual link **interface fc 1/1**.

## Trunk Port Mode

This example sets fc 1/1 trunk port mode to auto, on, and off. The default mode is auto. One end of an ISL should be set to on when connected between two MDS switches, while the other end can be either on or auto. The other end of the ISL needs to be off when talking to nonMDS switches.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface fc 1/1
```



```

mds-switch-1(config-if)# switchport trunk mode auto <- auto negotiates trunk port mode
mds-switch-1(config-if)# switchport trunk mode on <- sets trunk port mode to on
mds-switch-1(config-if)# switchport trunk mode off <- sets trunk port mode to off
mds-switch-1(config-if)# exit
mds-switch-1#

```

## Configuring Trunk Ports to Filter-Specific VSANs

This recipe configures allowed VSAN traffic through the interface fc 1/1. The **all** keyword allows all VSAN traffic to go through the port. **Add 2** adds VSAN 2 to the list of VSANs allowed through the port. **Add 2-4** adds VSANs 2 through 4 to the list of VSANs allowed through the port. Default mode is to allow all VSAN traffic to pass through the port.

```

mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface fc 1/1
mds-switch-1(config-if)# switchport trunk allowed vsan all <- all VSAN traffic
mds-switch-1(config-if)# switchport trunk allowed vsan add 2 <- only VSAN 2 traffic
mds-switch-1(config-if)# switchport trunk allowed vsan add 2-4 <- VSAN 2 to 4 traffic
mds-switch-1(config-if)# ^Z
mds-switch-1#

```

## Port Mode F

Setting port mode to F restricts the port to operating as an F port. F port mode is used for end devices that can only communicate in point-to-point mode. It is a mode used on FC ports on the switch that connect to host HBA or storage port or a tape device. In this example, fc 1/1 is set to F port mode.

```

mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface fc 1/1
mds-switch-1(config-if)# switchport mode F
mds-switch-1(config-if)# end
mds-switch-1#

```

## Port Mode FL

Setting port mode to FL restricts the port to operating as an FL port. FL port mode is used for end devices that is a Fibre Channel arbitrated loop (FCAL) device. In this example, fc 1/1 is set to FL port mode.

```

mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface fc 1/1
mds-switch-1(config-if)# switchport mode FL
mds-switch-1(config-if)# end
mds-switch-1#

```

## Port Mode Fx

Setting port mode to Fx restricts the port to operating as either an F or FL port. Fx port mode is used exclusively for end devices and prevents a port from autonegotiating to an E port. In this example, fc 1/1 is set to Fx port mode.

```

mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.

```

```
mds-switch-1(config)# interface fc 1/1
mds-switch-1(config-if)# switchport mode Fx
mds-switch-1(config-if)# end
mds-switch-1#
```

## Port Mode SD

Setting port mode to SD configures the port as the span destination (SD) port of a span session. This is used in conjunction with the port analyzer adapter (PAA) to span a port and obtain FC traces without a FC analyzer. In this example, fc 1/1 is set to SD port mode.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface fc 1/1
mds-switch-1(config-if)# switchport mode SD
mds-switch-1(config-if)# end
```

## Port Mode ST

Setting port mode to ST configures the port as the span tunnel (ST) port of a remote span session. This is used to set up a remote SPAN session to a remote switch in which a PAA or protocol analyzer is connected. In this example, fc 1/1 is set to ST port mode.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface fc 1/1
mds-switch-1(config-if)# switchport mode ST
mds-switch-1(config-if)# end
mds-switch-1#
```

## Port Mode TL

Setting port mode to TL restricts the port to operating as a TL port. TL port mode is used exclusively for end devices that can only communicate as a private loop device. In this example, fc 1/1 is set to TL port mode.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface fc 1/1
mds-switch-1(config-if)# switchport mode TL
mds-switch-1(config-if)# end
mds-switch-1#
```

## Enabling Port Beacons

Using the **switchport beacon** command shown in this example causes the LEDs below port fc 1/1 to start flashing. This is useful in identifying a port for physical cabling or troubleshooting.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface fc 1/1
```

```
mds-switch-1(config-if)# switchport beacon
mds-switch-1(config-if)# end
```

## Oversubscription Management of Ports and Rate Limiting

The section describes how oversubscription on a 24- and 48-port modules can be improve management using the rate-limiting feature available on these two modules. The 24-port module is oversubscribed 2:1 at 4-Gbps speeds and a 48-port modules is oversubscribed 2:1 at 2-Gbps and 4:1 at 4-Gbps speeds.



### Note

The 3-port modules do not support oversubscription and rate-limit management per port. The 32-port modules supports a fixed 3.3:1 oversubscription on all ports.

On a 24-port module, there are four groups of 6 ports. Each group of 6 ports has 12.8 Gigabits to the crossbar. Similarly, on a 48-port module there four groups of 12 ports each. Each group of 12 ports has 12.8 Gigabits to the crossbar. By default, all the ports on the 24- and 48-port modules use the port rate-mode as share. The rate-limiting feature allows the user to decide which port gets how much of the bandwidth in a port group.

On a 24-port module or a 48-port module, a maximum of 3 ports from each of the 4 port groups can be configured to operate at 4-Gbps each. This essentially permits only 0.8 Gbps for the remaining ports in the port group which is insufficient to operate the other ports in that group. In order to get 3 ports of the 6 ports in a port group of a 24-port module and 3 ports of the 12 ports in the port group of a 48-port modules to get dedicated 4 Gigabits of bandwidth, the remaining ports in the port group need to be set to out-of-service or put into shared mode with a maximum speed of 1-Gbps, or one port could be at 2-Gbps shared and 2 other ports in out-of-service mode.

There are two modes of oversubscription enforcement on the 24- and 48-port modules: strict mode oversubscription and unlimited mode oversubscription.

In the strict mode oversubscription, a 4:1 ratio on the 24-port modules and 5:1 oversubscription ratio on 48-port modules is enforced. In the unlimited oversubscription mode, no specific oversubscription is enforced, which means that if certain ports have been allocated dedicated bandwidth but are not currently using it, the spare bandwidth can be used by the ports that do not have dedicated bandwidth.

The **show port-resource module slot** command lists the current status of the bandwidth allocation and the rate-mode for each port group.

```
mds-switch-2# show port-resources module 2
Module 2
Available dedicated buffers are 5400
```

```
Port-Group 1
Total bandwidth is 12.8 Gbps
Total shared bandwidth is 12.8 Gbps
Allocated dedicated bandwidth is 0.0 Gbps

Interfaces in the Port-Group B2B Credit Bandwidth Rate Mode
 Buffers (Gbps)

fc2/1 16 2.0 shared
fc2/2 16 2.0 shared
fc2/3 16 4.0 shared
fc2/4 16 4.0 shared
fc2/5 16 4.0 shared
fc2/6 16 4.0 shared

Port-Group 2
```

```
Total bandwidth is 12.8 Gbps
Total shared bandwidth is 12.8 Gbps
Allocated dedicated bandwidth is 0.0 Gbps
```

| Interfaces in the Port-Group | B2B Credit<br>Buffers | Bandwidth<br>(Gbps) | Rate Mode |
|------------------------------|-----------------------|---------------------|-----------|
| fc2/7                        | 16                    | 4.0                 | shared    |
| fc2/8                        | 16                    | 4.0                 | shared    |
| fc2/9                        | 16                    | 4.0                 | shared    |
| fc2/10                       | 16                    | 4.0                 | shared    |
| fc2/11                       | 16                    | 4.0                 | shared    |
| fc2/12                       | 16                    | 4.0                 | shared    |

```
Port-Group 3
Total bandwidth is 12.8 Gbps
Total shared bandwidth is 12.8 Gbps
Allocated dedicated bandwidth is 0.0 Gbps
```

| Interfaces in the Port-Group | B2B Credit<br>Buffers | Bandwidth<br>(Gbps) | Rate Mode |
|------------------------------|-----------------------|---------------------|-----------|
| fc2/13                       | 16                    | 2.0                 | shared    |
| fc2/14                       | 16                    | 2.0                 | shared    |
| fc2/15                       | 16                    | 4.0                 | shared    |
| fc2/16                       | 16                    | 4.0                 | shared    |
| fc2/17                       | 16                    | 4.0                 | shared    |
| fc2/18                       | 16                    | 4.0                 | shared    |

```
Port-Group 4
Total bandwidth is 12.8 Gbps
Total shared bandwidth is 12.8 Gbps
Allocated dedicated bandwidth is 0.0 Gbps
```

| Interfaces in the Port-Group | B2B Credit<br>Buffers | Bandwidth<br>(Gbps) | Rate Mode |
|------------------------------|-----------------------|---------------------|-----------|
| fc2/19                       | 16                    | 4.0                 | shared    |
| fc2/20                       | 16                    | 4.0                 | shared    |
| fc2/21                       | 16                    | 4.0                 | shared    |
| fc2/22                       | 16                    | 4.0                 | shared    |
| fc2/23                       | 16                    | 4.0                 | shared    |
| fc2/24                       | 16                    | 4.0                 | shared    |

```
mds-switch-2#
```

## Strict Oversubscription Mode Recipes

The following two recipes describe how strict oversubscription mode is configured in the 24- and 48-port modules. A strict 4:1 oversubscription ration on the 24-port module and a 5:1 oversubscription ration on the 48-port module is enforced in this mode.

The following recipes use 24-port modules. The same procedure applies for configuring a 48-port module also.

To configure three ports in the first port group to have 4-Gbps bandwidth, follow these steps:

- 
- Step 1** Choose three ports that require the dedicated bandwidth (ports (2/2, 2/3, 2/5).
  - Step 2** Set the other three ports (ports 2/1,2/4,2/6) to out-of-service.
  - Step 3** Set the rate-mode on the three ports that require the dedicated bandwidth to dedicated.

- Step 4** Set the bandwidth on the three ports to 4-Gbps.
- Step 5** Enable the ports.

The following example shows the preceding steps. It dedicates 3 ports on the first group of 6 ports on a 24-port modules to have 4-Gbps dedicated bandwidth and the remaining ports are set to out-of-service.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config-if)# interface fc 2/4
mds-switch-2(config-if)# out-of-service force
mds-switch-2(config-if)# interface fc 2/6
mds-switch-2(config-if)# out-of-service force
mds-switch-2(config-if)# interface fc 2/2
mds-switch-2(config-if)# interface fc 2/2
mds-switch-2(config-if)# switchport rate-mode dedicated
mds-switch-2(config-if)# switchport speed 4000
mds-switch-2(config-if)# interface fc 2/3
mds-switch-2(config-if)# switchport rate-mode dedicated
mds-switch-2(config-if)# switchport speed 4000
mds-switch-2(config-if)# interface fc 2/5
mds-switch-2(config-if)# switchport rate-mode dedicated
mds-switch-2(config-if)# switchport speed 4000
mds-switch-2(config)# interface fc 2/2-3, fc 2/5
mds-switch-2(config-if)# no shut
mds-switch-2(config-if)# end
mds-switch-2#
```

The **show port-resource module 2** command shows the status of the ports in port group 1.

```
mds-switch-2# show port-resources module 2
Module 2
Available dedicated buffers are 5478
```

**Port-Group 1**

```
Total bandwidth is 12.8 Gbps
Total shared bandwidth is 0.8 Gbps
Allocated dedicated bandwidth is 12.0 Gbps
```

| Interfaces in the Port-Group | B2B Credit<br>Buffers | Bandwidth<br>(Gbps) | Rate Mode |
|------------------------------|-----------------------|---------------------|-----------|
| fc2/1 (out-of-service)       |                       |                     |           |
| fc2/2                        | 16                    | 4.0                 | dedicated |
| fc2/3                        | 16                    | 4.0                 | dedicated |
| fc2/4 (out-of-service)       |                       |                     |           |
| fc2/5                        | 16                    | 4.0                 | dedicated |
| fc2/6 (out-of-service)       |                       |                     |           |

The following recipe shows how to configure the same port group to operate all six ports in that group with the following combination: three ports have dedicated 4-Gbps bandwidth and the three other ports can operate at 1-Gbps shared instead of being shut down. When the 12-Gbps bandwidth is allocated to three of the six ports, the remaining bandwidth is just 0.8-Gbps with 4:1 over subscription. This is enforced at a maximum of  $0.8 * 4 = 3.2$ -Gbps bandwidth that can be supported on the remaining three ports. If the remaining three ports need to operate, they can do so with a maximum bandwidth of 1-Gbps each in shared mode.

To configure all six ports in the first port group to have a combination dedicated and shared bandwidth as described in the preceding paragraph, follow these steps:

- 
- Step 1** Set the port speed of the three ports to 1-Gbps (2/, 2/4, 2/6)
- Step 2** Set the rate-mode to dedicated on the port that requires dedicated bandwidth.
- Step 3** Enable the port in that port group.
- 

The following example shows the preceding steps.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# interface fc 2/1, fc 2/4, fc 2/6
mds-switch-2(config-if)# switchport speed 1000
mds-switch-2(config-if)# interface fc 2/2-3, fc 2/5
mds-switch-2(config-if)# switchport rate-mode dedicated
mds-switch-2(config-if)# no shut
mds-switch-2(config-if)# exit
mds-switch-2(config)# interface fc 2/1, fc 2/4, fc 2/6
mds-switch-2(config-if)# no shut
mds-switch-2(config-if)# end
mds-switch-2#
```

The **show port-resource module 2** command lists the status of the ports in port groups on module 2.

```
mds-switch-2# show port-resources module 2
Module 2
Available dedicated buffers are 5391
```

```
Port-Group 1
Total bandwidth is 12.8 Gbps
Total shared bandwidth is 0.8 Gbps
Allocated dedicated bandwidth is 12.0 Gbps

Interfaces in the Port-Group B2B Credit Bandwidth Rate Mode
 Buffers (Gbps)

fc2/1 16 1.0 shared
fc2/2 16 4.0 dedicated
fc2/3 16 4.0 dedicated
fc2/4 16 1.0 shared
fc2/5 16 4.0 dedicated
fc2/6 16 1.0 shared
```

## Unlimited Oversubscription Mode Recipe

The next two recipes describe how unlimited oversubscription mode is configured in the 24- and 48-port modules. There are three ports in the first port group with 4-Gbps bandwidth dedicated and the remaining three ports, also configured for 4-Gbps, are in shared mode. The three ports that are configured for shared bandwidth may use the bandwidth in the share pool and any unused bandwidth that is not being used by the dedicated ports.



### Note

Only unused bandwidth from the ports with dedicated bandwidth can be used by the ports configured with the shared bandwidth. This mode of oversubscription is available in Cisco SAN-OS Release 3.1(x) and higher.

---

The following recipes use a 24-port modules, but the same procedure applies for configuring a 48-port module.

The command **no rate-mode oversubscription-limit module slot** enables the unlimited oversubscription mode in that module. In this configuration all the ports can operate at 4-Gbps but only interfaces fc 2/2,2/3 and 2/5 are guaranteed 4-Gbps bandwidth.

To configure three ports in the first port group with 4-Gbps dedicated bandwidth and the remaining three ports with 4-Gbps in shared mode, follow these steps:

- 
- Step 1** Use the **shutdown** command on all the ports in the port group (2/1-2/6).
- Step 2** Set the rate mode dedicated on the interfaces that need dedicated 4-Gbps (2/2, 2/3, 2/5)
- Step 3** Use the **no shutdown** command all the ports in the port group.
- 

The following example shows the preceding steps:

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# interface fc 2/1-6
mds-switch-2(config-if)# shut
mds-switch-2(config-if)# no rate-mode oversubscription-limit module 2
mds-switch-2(config)# interface fc 2/2-3, fc 2/5
mds-switch-2(config-if)# switchport rate-mode dedicated
mds-switch-2(config-if)# exit
mds-switch-2(config)# interface fc 2/1-6
mds-switch-2(config-if)# no shut
mds-switch-2(config-if)# end
mds-switch-2#
```

The **show port-resources module 2** command lists oversubscription related data.

```
mds-switch-2# show port-resources module 2
Module 2
 Available dedicated buffers are 5391
```

```
Port-Group 1
 Total bandwidth is 12.8 Gbps
 Total shared bandwidth is 0.8 Gbps
 Allocated dedicated bandwidth is 12.0 Gbps
```

| Interfaces in the Port-Group | B2B Credit Buffers | Bandwidth (Gbps) | Rate Mode |
|------------------------------|--------------------|------------------|-----------|
| fc2/1                        | 16                 | 4.0              | shared    |
| fc2/2                        | 16                 | 4.0              | dedicated |
| fc2/3                        | 16                 | 4.0              | dedicated |
| fc2/4                        | 16                 | 4.0              | shared    |
| fc2/5                        | 16                 | 4.0              | dedicated |
| fc2/6                        | 16                 | 4.0              | shared    |

## Configuring Gigabit Ethernet Ports

The following section describes how to configure Gigabit Ethernet ports.

## Configuring VRRP

Virtual Router Redundancy Protocol (VRRP) allows two Gigabit Ethernet interfaces to provide failover capability for an IP address. The two interfaces form an active/passive or master/backup state in which one interface services requests for the shared IP address, while the other remains in a backup or standby state. It is ideal for providing port level redundancy in iSCSI configurations. A Gigabit Ethernet port can still have its own IP address while participating in a VRRP configuration.

A VRRP session has an ID assigned to it and the two interfaces use it to communicate to identify its peer. The same ID must be used on both switches. The procedure for having both members of the VRRP pair on the same switch would be the same as if the two members were on different switches.



### Note

To have one interface become the master interface whenever it is online (preemption), set the Gigabit Ethernet interface to have the same IP address as the VRRP IP address.

In this example, the following resources are used:

- VRRP ID: 1
- VRRP IP address: 192.168.1.40
- Switch 1: Interface gige3/3 (192.168.1.20)
- Switch 2: Interface gige4/1 (192.168.1.30)

To configure VRRP, follow these steps:

### Step 1 Configure IP addresses on the two Gigabit Ethernet interfaces.

```
mds-switch-1# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface gigabitethernet 3/3
mds-switch-1(config-if)# ip address 192.168.1.20 255.255.255.0
mds-switch-1(config-if)# no shut

mds-switch-2# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# interface gigabitethernet 4/1
mds-switch-2(config-if)# ip address 192.168.1.30 255.255.255.0
mds-switch-2(config-if)# no shut
```

At this point, it is a good idea to verify that a host on the local subnet can ping both IP addresses (192.168.1.20 and 192.168.1.30). Alternatively, the **ips measure-rtt** command can be used to ping one Gigabit Ethernet port from the other.

```
mds-switch-1# ips measure-rtt 192.168.1.30 interface gigabitethernet 3/3
Round trip time is 172 micro seconds (0.17 milli seconds)
```

### Step 2 Configure the VRRP session on both switches using the VRRP id (1).

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface gigabitethernet 3/3
mds-switch-1(config-if)# vrrp 1
mds-switch-1(config-if-vrrp)# address 192.168.1.40
mds-switch-1(config-if-vrrp)# no shut
mds-switch-1(config-if-vrrp)# end

mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
```



```

mds-switch-2(config)# interface gigabitethernet 4/1
mds-switch-2(config-if)# vrrp 1
mds-switch-2(config-if-vrrp)# address 192.168.1.40
mds-switch-2(config-if-vrrp)# no shut
mds-switch-2(config-if-vrrp)# end

```

**Step 3** Verify that the VRRP session is up and determine which interface has become the master with the **show vrrp vr** command.

```

mds-switch-2# show vrrp vr 1

```

| Interface          | VR | Status |
|--------------------|----|--------|
| GigabitEthernet3/3 | 1  | backup |

```

mds-switch-1# show vrrp vr 1 interface gig3/3 status
vr id 1 status
MAC address 00:00:5e:00:01:01
Operational state: master
Up time 8 sec

```

View the configuration with the **show vrrp vr** command.

```

mds-switch-1# show vrrp vr 1 interface gigabitethernet 3/3 configuration
vr id 1 configuration
admin state up
priority 100
associated ip: 192.168.1.40
no authentication
advertisement-interval 1
preempt no
protocol IP

```

## Implementing WWN-Based VSANs

Dynamic Port VSAN Membership (DPVM) provides the ability to have an interface VSAN assignment determined by the world-wide name (WWN) of the device that is logging in, and not by the configuration of the physical port. The primary advantage of using DPVM occurs when a device is moved from one port on a switch to another port on the same or different switch. The device ends up in the same VSAN, preventing further configuration changes, or the device ends up in the wrong VSAN. It is also useful if the WWN is known for a device, but the interface that it will be plugged into is not yet known.

DPVM can leverage the CFS infrastructure and we recommend using it to maintain database synchronization and locking. The default on the switch is that the DPVM application uses CFS infrastructure. To populate the database, either autolearning can be enabled, which uses the VSAN that each device that is currently logged into, or a VSAN can be manually specified. The second method can be used for devices that are not yet in the fabric.



### Note

- A DPVM configuration overrides the VSAN assigned to the port. Therefore, changing the VSAN membership of an interface that has a DPVM configured device attached has no effect on the VSAN of the device.
- DPVM CFS scope is physical.

- DPVM can work in conjunction with persistent FCIDs. However, if the device moves to another switch, it is assigned a different FCID.

---

After configuring DPVM in Fabric Manager, if the DPVM-assigned VSAN is different from the port-assigned VSAN, the operational value for the VSAN is the DPVM assigned value.

The CLI is different. In the CLI, the operational VSAN assignment is displayed in the **Port vsan** field. The configured VSAN displays the VSAN that the port would belong to if DPVM was not configured.

```
switch# show int fc2/5
fc2/5 is up
 Hardware is Fibre Channel, SFP is short wave laser w/o OFC (SN)
 Port WWN is 20:45:00:0c:85:e9:d2:c0
 Admin port mode is auto, trunk mode is on
 Port mode is F, FCID is 0xef0008
 Configured Port vsan is 1
 Port vsan is 1000
 Speed is 2 Gbps
 Transmit B2B Credit is 7
 Receive B2B Credit is 16
 Receive data field Size is 2112
```

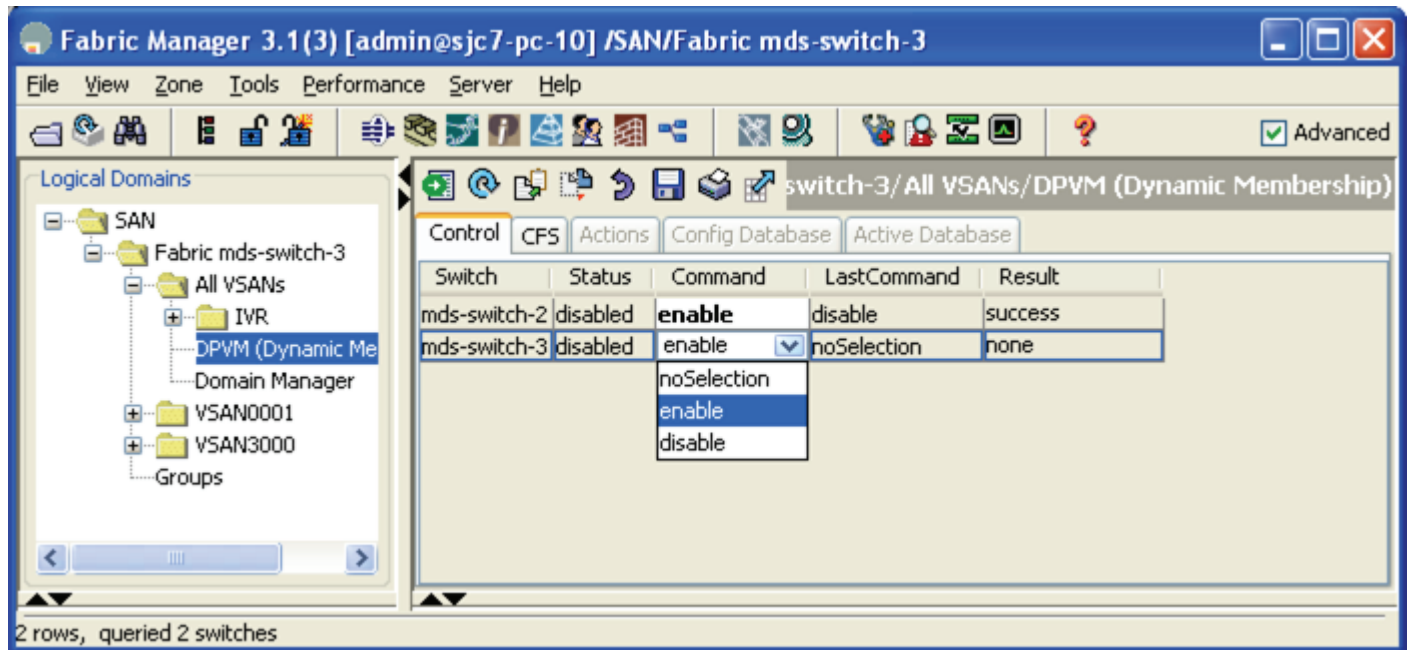
Enable DPVM with either the CLI or Fabric Manager before any actual configuration activities take place. Use the CLI command **dpvm enable** in configure mode to do this.

```
switch# conf terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# dpvm enable
```

To enable DPVM in Fabric Manager, follow these steps:

- 
- Step 1** From Logical Domains, choose **All VSANs, > DPVM**.
  - Step 2** Click the **Control** tab.
  - Step 3** Set the command field to **enable** and click **Apply Changes** (see [Figure 4-2](#)).

Figure 4-2 Enabling DPVM with Fabric Manager



After enabling DPVM, proceed to either [“Adding Existing Devices to DPVM”](#) on page 15 or [“Adding New Devices to DPVM”](#) on page 18 for the rest of the recipe.

## Adding Existing Devices to DPVM

If DPVM is being configured for an environment with existing end devices and ports already assigned to VSANs, use this recipe for the DPVM wizard to import the VSAN configuration into DPVM.

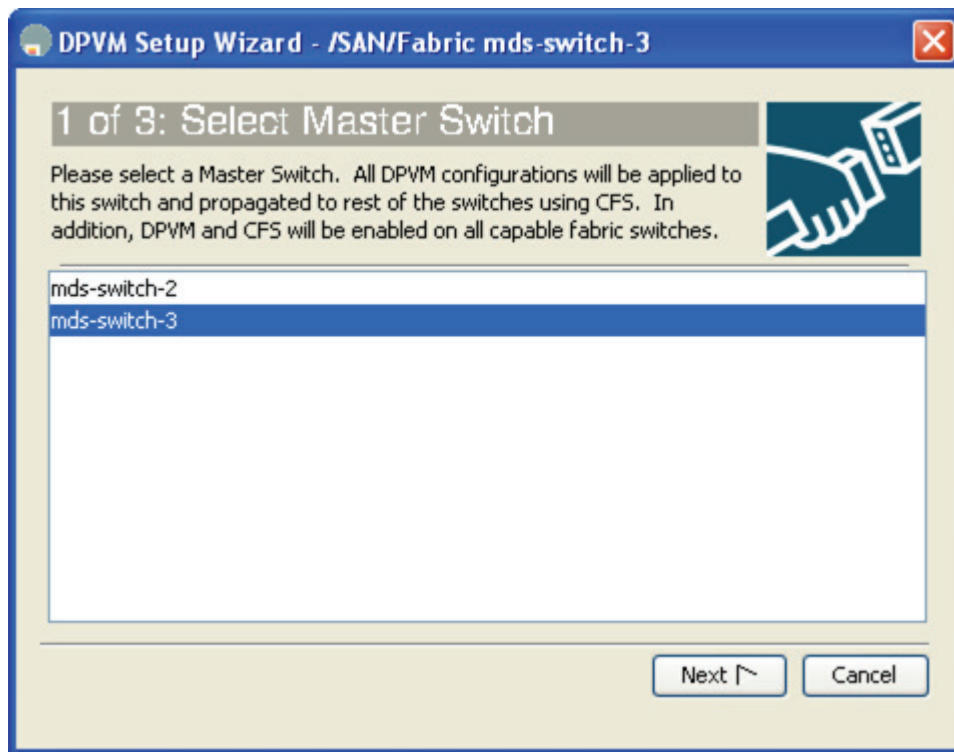
In this recipe, the following resources are used:

- Hosts: 50:06:0e:80:03:4e:95:33 and 10:00:00:00:c9:3b:54:78
- VSAN: 3000

To have the DPVM wizard import the VSAN configuration of existing end devices and port into DPVM, follow these steps:

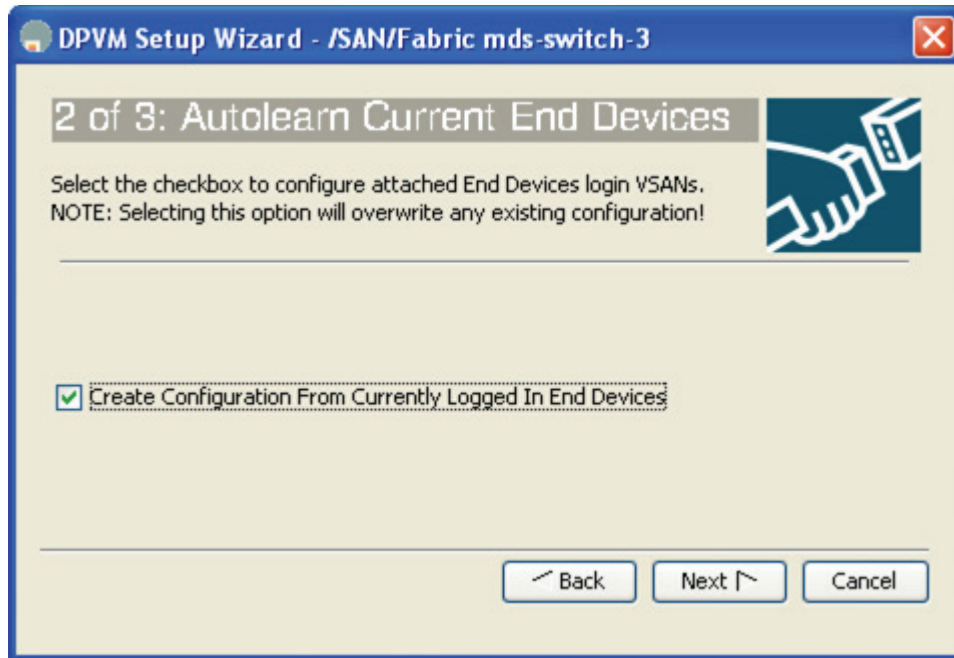
- Step 1** Enable DPVM as per [“Implementing WWN-Based VSANs”](#) on page 13.
- Step 2** In Fabric Manager, select **Tools > Other > DPVM Setup**. You see the DPVM Setup Wizard (see [Figure 4-3](#)).

Figure 4-3 DPVM Setup Wizard



- Step 3** Select any one of the switches enabled for DPVM. Because DPVM is a CFS-aware application, the DPVM configuration is propagated to all switches. If a switch is listed as not having DPVM configured, then the wizard automatically enables DPVM on that switch. Click **Next**.
- Step 4** Because this is a new DPVM configuration, check the Create Configuration From Currently Logged In End Devices check box and click **Next**. The dialog box is shown in [Figure 4-4](#).

Figure 4-4 DPVM Autolearn Current End Devices

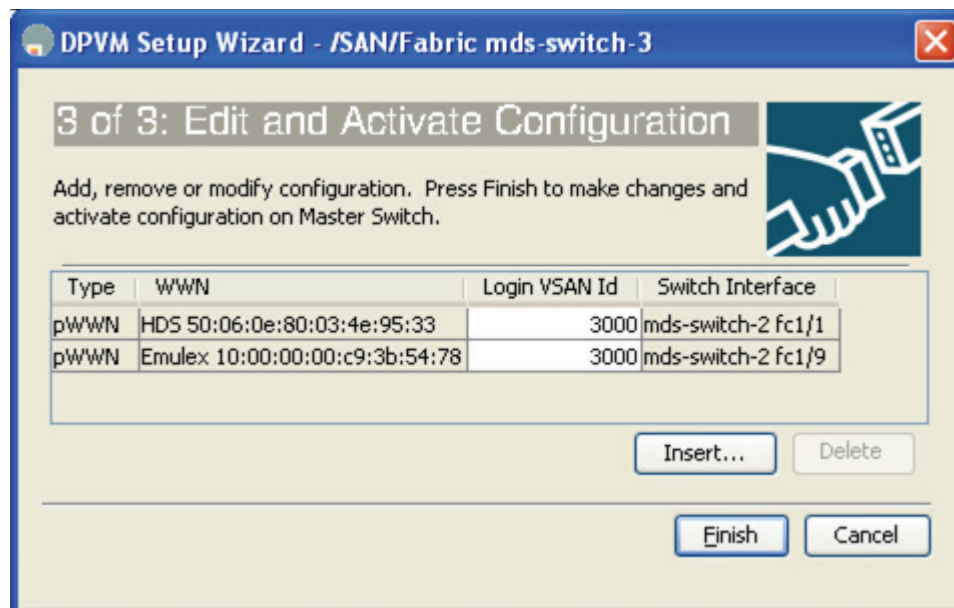
**Caution**

When an existing DPVM configuration exists, do not check the Create Configuration From the Currently Logged In End Devices check box.

**Step 5**

At this point, FM determines the VSAN assignment of all the devices in the fabric and presents a table listing the proposed configuration (see [Figure 4-5](#)).

Figure 4-5 Proposed DPVM Configuration



At this stage, you can do the following to add or remove entries from the new database:

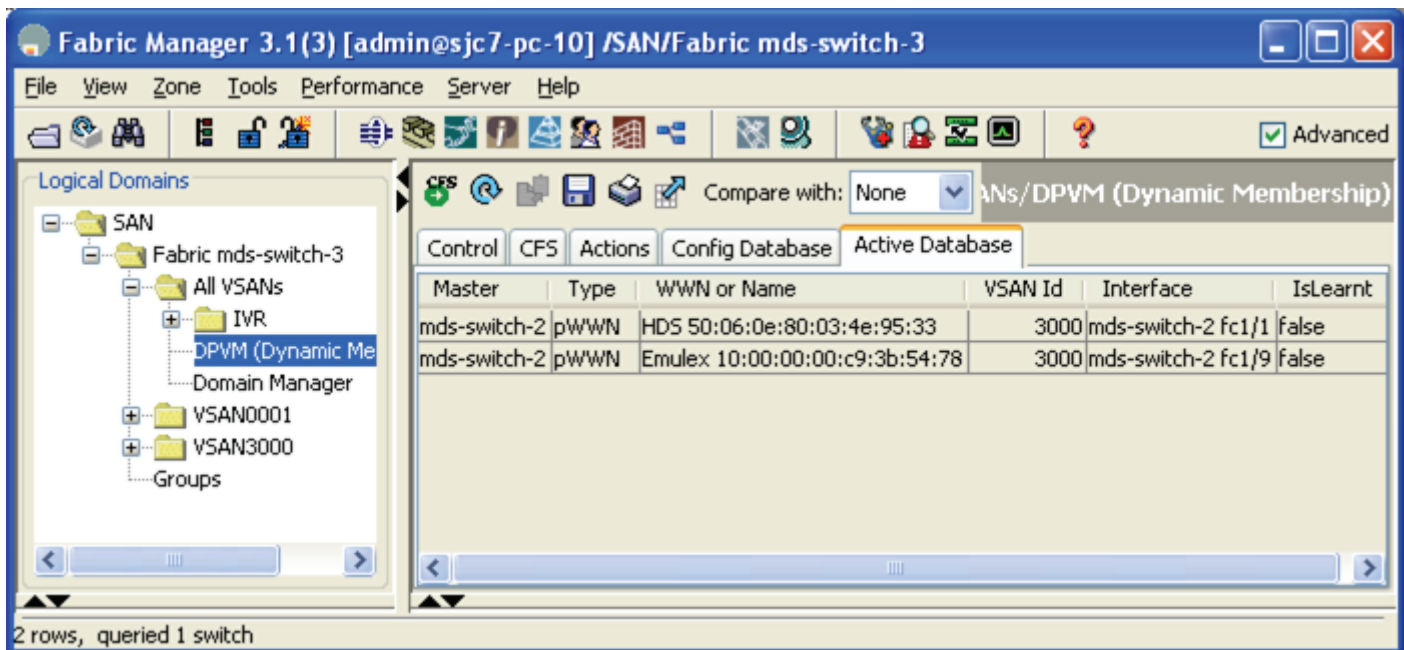
- a. If additional entries are desired, click **Insert...** and provide a WWN and VSAN.
- b. If an entry should be deleted, select the row to be removed and click **Delete**.

**Step 6** Click **Finish** to CFS commit the action.

**Step 7** To view the DPVM configuration in Fabric Manager, in the Logical Domains pane select the Fabric to view and select **All VSANs > DPVM**.

**Step 8** Click the **CFS** tab, which activates the other tabs, then click the **Active Database** tab. The active database screen is shown in Figure 4-6.

Figure 4-6 DPVM Active Database



## Adding New Devices to DPVM

In this recipe, the following resources are used:

- Storage: 50:06:0e:80:03:4e:95:23 and Host 21:00:00:e0:8b:82:99:dc
- VSAN: 3000

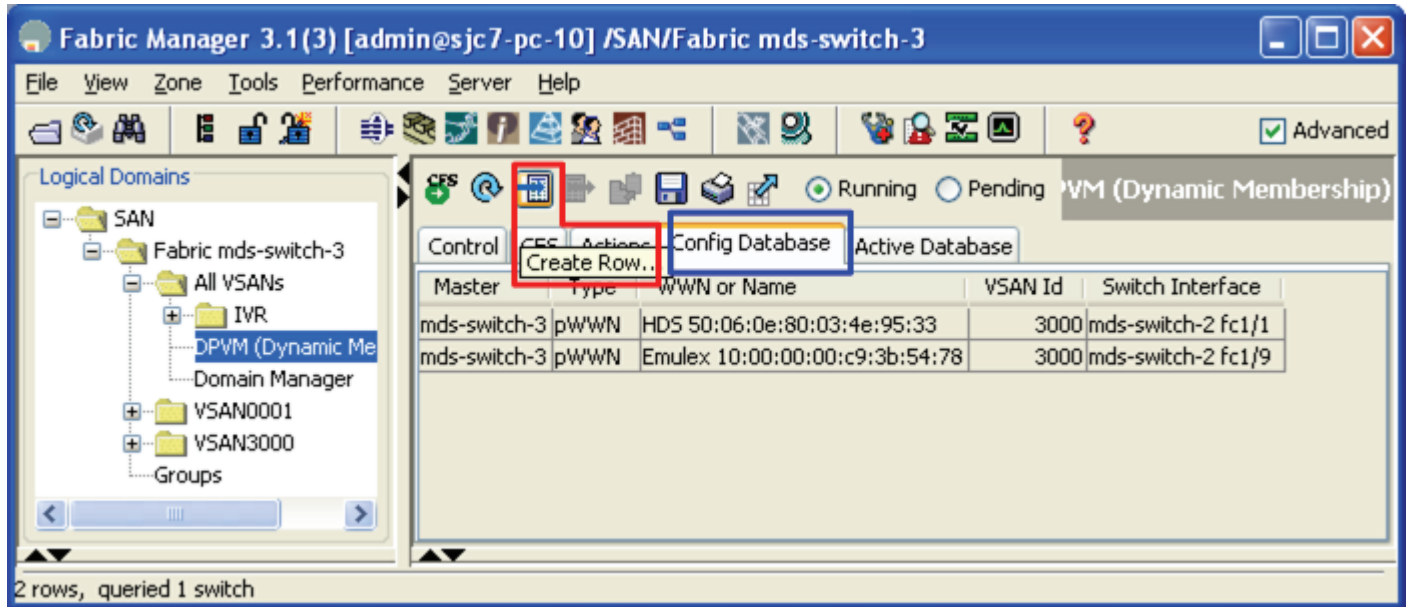
To add new devices to DPVM, follow these steps:

**Step 1** Enable DPVM as per “[Implementing WWN-Based VSANs](#)” on page 13.

**Step 2** Access the DPVM configuration in Fabric Manager from the Logical Domains pane by selecting the **Fabric to view > All VSANs > DPVM**.

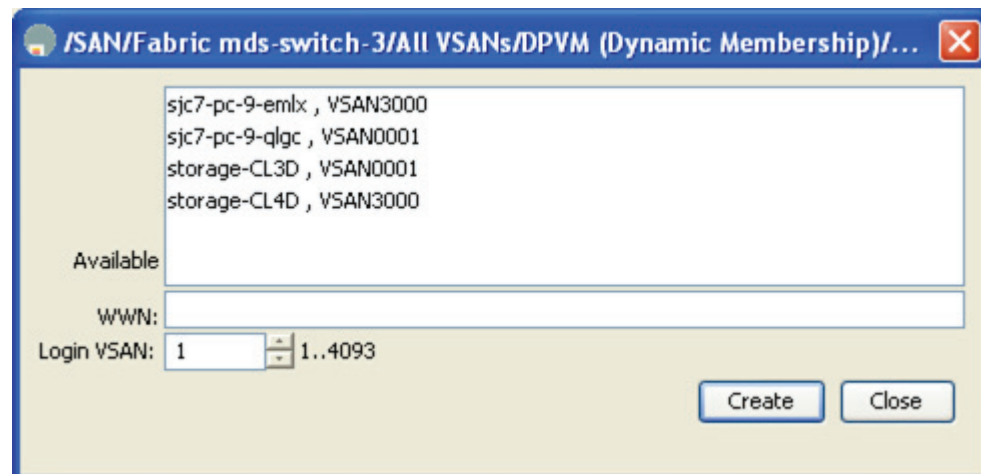
- Step 3** Click the **CFS** tab, which activates the other CFS tabs, and then choose the **Config Database** tab. You see the DPVM Config Database screen as shown in Figure 4-7.

Figure 4-7 DPVM Config Database



- Step 4** Click the **Create Row...** icon (located above the CFS tab). The screen in Figure 4-8 appears.

Figure 4-8 Creating the DPVM entry.



- Step 5** Either select a device from the list or type the port world-wide name (pWWN).
- Step 6** Select the VSAN to be assigned.
- Step 7** Click **Create**.
- Step 8** When all entries have been created, click **Close**.

In this example, device aliases have been configured (see [Device Aliases, page 1-41](#)), and device aliases are displayed instead of the WWNs.

The next step is to activate the new entry.

**Step 9** Click the **Actions** tab.

**Step 10** Change the action to **activate** and click the green **Apply Changes** icon. Using the **activate** action ensures that a device currently logged into the fabric does not get accidentally moved into another VSAN and disrupt I/O.

To changes the VSAN of an actively logged in device using DPVM use the **forceaction** action to force activate the new database.

**Note**

Using the **forceactivate** action is disruptive for ports whose VSAN is being changed. Be careful when you use **forceactivate**.

At this point the configuration and active databases are different. Because DPVM is CFS-enabled, a CFS commit is still required.

**Step 11** Commit the changes, by selecting **CFS > Commit**. If the commit succeeds, you see the message **CFS(dpvm):Committed**.

At this point, the active database contains the new entry.

## Modifying the VSAN Assignment of a DPVM Entry

In this recipe, the VSAN assignment of a device is changed.

This example uses the following resources:

- PWWN: 21:00:00:e0:8b:82:99:dc
- Old VSAN: 3000
- New VSAN: 1000

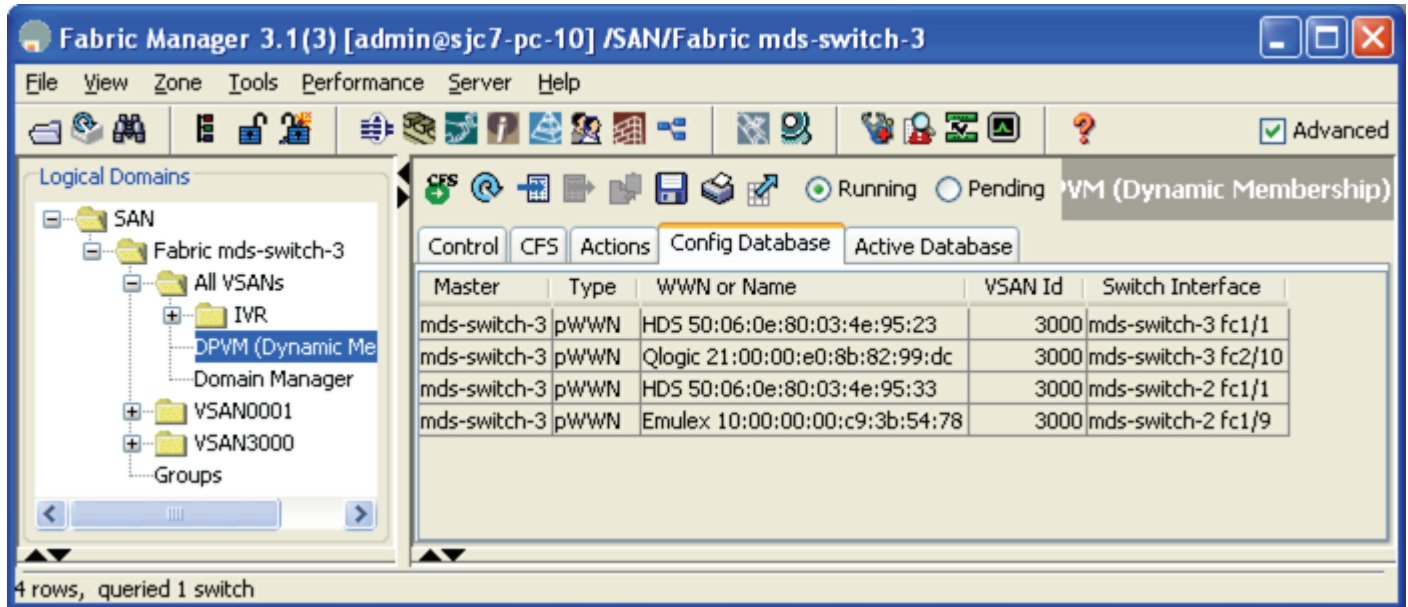
To modify the VSAN assignment of a DPVM entry, follow these steps:

**Step 1** To access the DPVM configuration in Fabric Manager, in the Logical Domains pane select the **Fabric to view > All VSANs > DPVM**.

**Step 2** Click the **CFS** tab to activate the other tabs, then click the **Config Database** tab (see [Figure 4-9](#)).

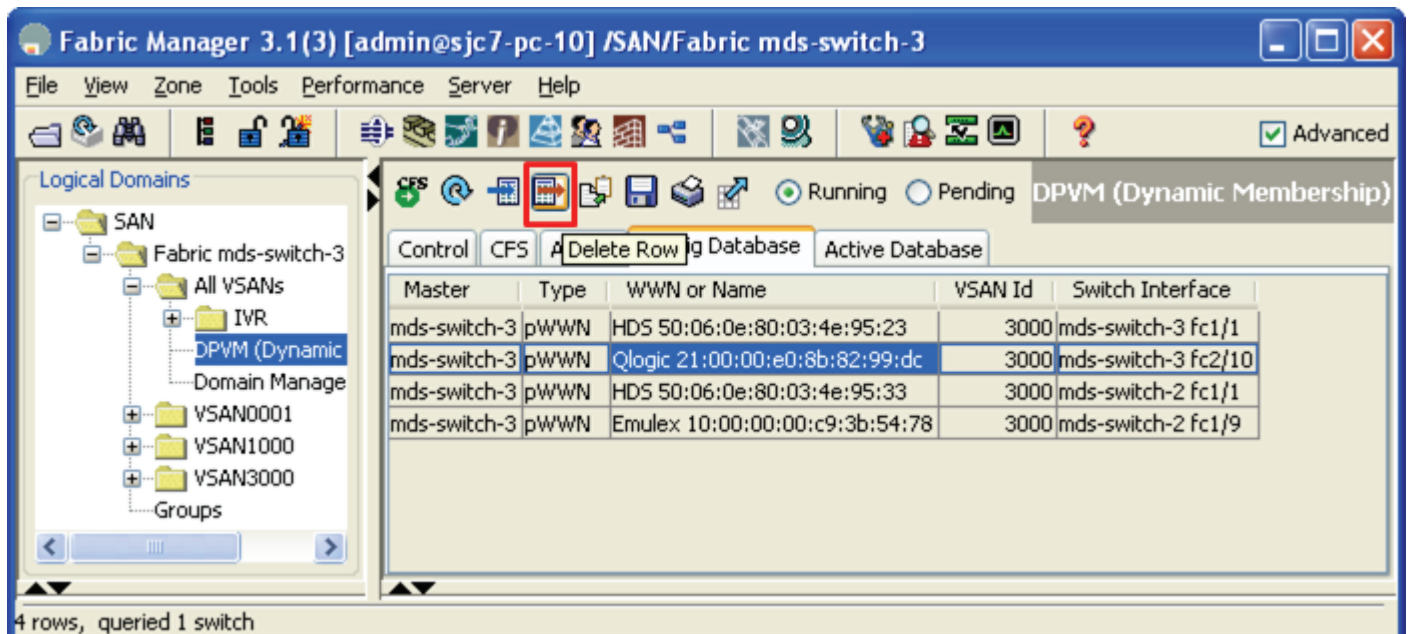


Figure 4-9 DPVM Config Database



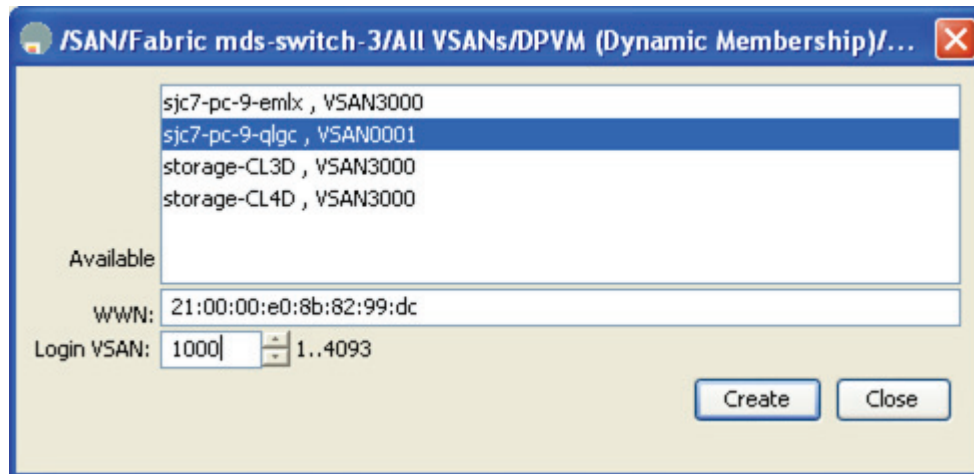
- Step 3** Ensure that the new VSAN 1000 exists. Create the new VSAN if it does not exist.
- Step 4** Select the device whose VSAN has to be updated. This populates the pWWN.
- Step 5** Delete the entry from the DPVM databases seen in Figure 4-10. Click **Yes** to the popup confirming deletion.
- Step 6** In the Actions tab, select the **forceactivate** action and click the green **Apply** icon to save the changes.
- Step 7** Click **CFS** to do a CFS commit.

Figure 4-10 Delete the DPVM Database Entry of the Device that Needs to be Changed



- Step 8** In the Logical Domains pane, select the **Fabric to view > All VSANs > DPVM**.
- Step 9** Click the **CFS** tab to activate the other tabs, and then select the **Config Database** tab.
- Step 10** Click the **Create row** icon (see [Figure 4-7](#)).

Figure 4-11 Add the device into the new VSAN



- Step 11** Select the device and update the VSAN in the login VSAN box (see [Figure 4-11](#)). Click **Create** to add the device.

In this example, device aliases have been configured (see [Device Aliases](#), page 1-41), and device aliases are displayed instead of the WWNs.

Activate the change by proceeding with the next steps:

- Step 12** Choose the **Actions** tab.
- Step 13** Change the Action to **forceactivate** as the device is currently logged in and click the green **Apply Changes** icon.

At this point the configuration and active databases are different. Because DPVM is CFS-enabled, a CFS commit is still required.

- Step 14** Commit the changes by clicking **CFS > Commit**. If the commit succeeds, you see the message **CFS(dpvm):Committed....**

At this point, the active database contains the new entry.

## DPVM Conflicting Entries

When a DPVM configuration change is committed with a device that is already logged into the fabric, the CFS commit does not succeed. The CFS commits fails because performing a DPVM commit under these circumstances would change the device VSAN assignment and potentially cause an I/O disruption. If one of the switches cannot successfully update its configuration, then none of the switches can do it.

Follow the recipe to change the DPVM VSAN (see [Modifying the VSAN Assignment of a DPVM Entry](#), page 4-20.)

## DPVM with the CLI

DPVM can be manipulated with the CLI as well as with Fabric Manager. Both have the same underlying CFS infrastructure. To enable DPVM from the CLI, use the **dpvm enable** command.

```
ca-9506# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
ca-9506(config)# dpvm enable
```

## Adding Existing Devices to DPVM

To add existing devices that are already logged into the fabric, use the procedure outlined in the section [Adding Existing Devices to DPVM, page 4-15](#).

## Adding New Devices to DPVM

In this recipe, a new device is entered into the DPVM database and configured.

This example uses the following resources:

- Hosts: 50:06:0e:80:03:4e:95:33 and 10:00:00:00:c9:3b:54:78
- VSAN: 3000

To enter a new device into the DPVM database and configure it, follow these steps:

---

**Step 1** Log into the switch and enter configuration mode and DPVM database configuration submode.

```
mds-switch-3# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# dpvm database
mds-switch-3(config-dpvm-db)#
```

**Step 2** Enter the pWWN and the VSAN:

```
mds-switch-3(config-dpvm-db)# pwwn 50:06:0e:80:03:4e:95:33 vsan 3000
```

**Step 3** Activate the changes from the current CLI prompt or the previous prompt (enter **exit** to see it) with the command **dpvm activate**.

```
mds-switch-3(config-dpvm-db)# dpvm activate
```

**Step 4** Examine the changes with the **show dpvm pending-diff** command before committing them. Look for conflicting devices that may cause the CFS commit to fail.



### Note

- The + represents devices that are being added to the database.
  - The - represents devices that are being removed from the database.
  - The \* represents devices that are being modified in the database, including those that are currently logged into the fabric and are changing their VSAN assignment.
- 

The **do** keyword is required for exec commands in configuration mode.

```
mds-switch-3(config)# do show dpvm pending-diff
Session is on, Lock Taken
DPVM Pending Status

```

```

Active DB : Activate
Auto Learn : None

Pending Database Diff

Legend: "+" New Entry, "-" Missing Entry, "*" Possible Conflict Entry

+ pwwn 50:06:0e:80:03:4e:95:33 vsan 3000

```

**Step 5** Commit the changes with the **dpvm commit** command.

```
mds-switch-3(config)# dpvm commit
```

---

## Modifying the VSAN Assignment of a DPVM Entry

The procedure for modifying the VSAN assignment of a DPVM entry is the same as [Adding New Devices to DPVM](#), [page 4-23](#). However, it uses the **dpvm activate force** command instead of **dpvm activate** because the device being reassigned to a new VSAN is currently logged into the fabric.



## CHAPTER 5

# Logical Interfaces

---

## PortChannels

PortChannels aggregate multiple FC or FCIP links into a single, high-speed, fault-tolerant Fibre Channel or FCIP Inter-Switch Link (ISL). A PortChannel has the same configuration options as a single link Fibre Channel or FCIP ISL. However, building, modifying and reducing PortChannels is different from working with a single link Fibre Channel or FCIP ISL. This section discusses these differing PortChannel operations.



Tip

- A PortChannel should use interfaces on multiple modules to protect the PortChannel against module failure.
- The same channel group number should be used on both ends of a PortChannel. This aids in troubleshooting and identifying the corresponding channel group on the other switch.
- A PortChannel, like all other interfaces, can have a description. Use the description field to specify exactly where the PortChannel goes.
- PortChannels can use any port on the switch and connect to any other port on a switch.
- Set the initial VSAN Allowed List before bringing up the PortChannel. This prevents VSANs from merging during the initial start up.

## Quiesce a PortChannel or ISL Link

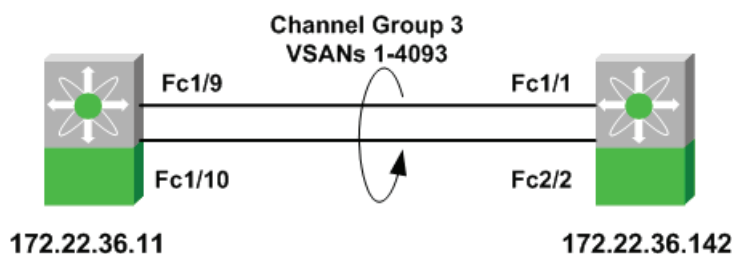
In Cisco SAN-OS Release 1.3, the **quiesce** command existed to explicitly inactivate an ISL or a PortChannel member so that the link could be hitlessly shut down or removed from the PortChannel. In SAN-OS Release 2.x, this function is the default when a **shutdown** command is issued to either an ISL or a PortChannel member. Therefore, the individual **quiesce** command was deprecated.

## Creating a PortChannel Using Fabric Manager

This Fabric Manager recipe creates a PortChannel from two existing ISLs. Because converting all ISLs between two switches into one PortChannel can be disruptive, this procedure first creates a one link PortChannel, and then adds a second link into the PortChannel. If traffic disruption is not a concern, both ISLs can be selected at one time.

The topology shown in [Figure 5-1](#) is used in this example.

Figure 5-1 PortChannel Creation with FM Topology

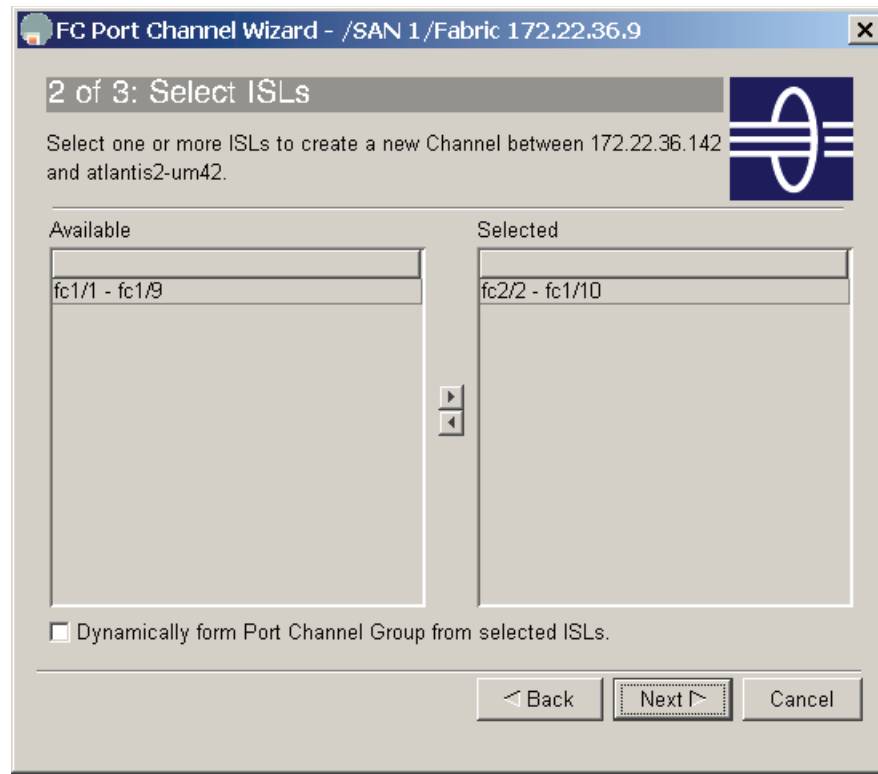


To create a PortChannel from Fabric Manager, open the topology map and follow these steps:

- 
- Step 1** On the map, right-click the first link to be converted to a PortChannel.
  - Step 2** Click **Create port channel...**

This displays Step 2 of 3 in the PortChannel Creation Wizard (see [Figure 5-2](#)). The first step was skipped because the map selection provided the necessary input.

Figure 5-2 Create PortChannel Wizard



The link you selected from the map appears in the Selected column while any other available or candidate links appear in the Available column. Because this PortChannel will be created one link at a time, do not move both links into the Selected column.

- Step 3** If you are creating an FCIP-based PortChannel in which the FCIP tunnels have write acceleration ([Enabling FCIP Write Acceleration, page 9-6](#)) enabled, check the **Dynamically form PortChannel Group from selected ISLs** check box.
- Step 4** Click **Next**.

Figure 5-3 PortChannel Wizard Screen Three

**3 of 3: Create Port Channel**

Please review the Channel attributes before pressing Finish to create.  
 Converting all ISL(s) simultaneously into a port channel may be disruptive.  
 NOTE: the Channel may take time to appear in map.

**Between Switch 172.22.36.142 (fc2/2)**

Channel Id:  1..128

Description:

**And Switch atlantis2-um42 (fc1/10)**

Channel Id:  1..128

Description:

**Channel Attributes**

VSAN List:  (1-4093) e.g. 1-22,29-45

Trunk Mode: ☐ nonTrunk ☒ trunk ☐ auto

☒ Force Admin, Trunk, Speed, and VSAN attributes to be identical

The Channel ID (or Channel Group), the descriptions, and the PortChannel attributes are displayed on this screen.

- Step 5** Modify any of the fields (see [Figure 5-3](#)). If the VSAN list needs to be modified to either add or remove VSANs, do this now.
- Step 6** Click **Finish**.
- Step 7** A warning is displayed concerning converting ISLs into PortChannels (see [Figure 5-4](#)). Because only one ISL is being converted into a PortChannel, the other untouched ISL continues to carry traffic. (Fabric Shortest Path First (FSPF) load balances around this link.) Click **Yes** to continue.

Figure 5-4 PortChannel Creation Warning

**FC Port Channel Wizard - /SAN 1/Fabric 172.22.36.9**

Converting ISL(s) into a port channel may be disruptive.  
 Do you want to continue?

On the Fabric Manager map, the link selected to become a PortChannel momentarily goes down (represented by a red X), and then comes back up as a thicker line.

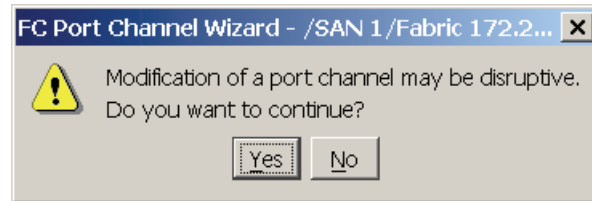
- Step 8** On the map, right-click the remaining PortChannel.
- Step 9** Click **Edit...**
- Step 10** Move the remaining ISL (fc1/1 - fc1/9) into the Selected column. You see the screen in [Figure 5-2](#) again.



**Step 11** Click **Finish**.

**Step 12** A warning is displayed concerning converting ISLs into PortChannels (see [Figure 5-5](#)). Since we are adding a link, the PortChannel is not affected. The second ISL is cycled and FSPF routes traffic over the previously created PortChannel. Click **Yes** to continue.

Figure 5-5 Port Modification Warning



As before, the ISL is marked with a red X as it goes down. However, it is soon removed from the map, leaving only the PortChannel represented by a thick line.

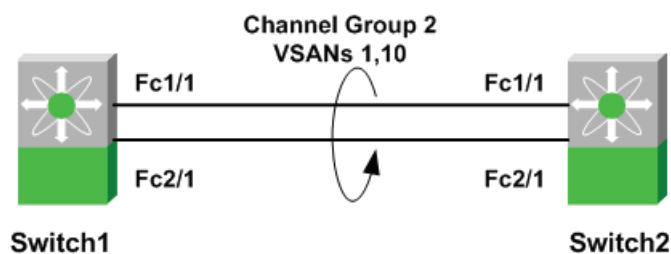
## Creating a PortChannel from the CLI

There are two ways to create a PortChannel: with the Fabric Manager Wizard or with the CLI. This procedure uses the CLI.

The following resources are used in this example:

- Switch1: Channel Group 2 and Interfaces fc1/1 and fc2/1
- Switch2: Channel Group 2 and Interfaces: fc1/1 and fc2/1
- Allowed VSANs: 1,10
- The topology shown in [Figure 5-7](#)

Figure 5-6 PortChannel Topology



To create a PortChannel using the CLI, follow these steps:

---

**Step 1** Create a PortChannel on switch1.

- a. Create the PortChannel on switch1 with the **channel-group** command. Create a description for the port with the **switchport description** command.

```
switch1# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch1(config)# interface fc1/1, fc2/1
switch1(config-if)# channel-group 2
fc1/1 fc2/1 added to port channel 2 and disabled
please do the same operation on the switch at the other end of the port channel,
then do "no shutdown" at both ends to bring them up
switch1(config-if)# switchport description "To switch2 PortChannel2"
```

- b. Enable trunking and set the VSAN allowed list on switch1.

```
switch1# config terminal
switch1(config)# int port channel 2
switch1(config-if)# switchport trunk mode on
switch1(config-if)# switchport trunk allowed vsan 1
switch1(config-if)# switchport trunk allowed vsan add 10
```

**Step 2** Create a PortChannel on switch2.

- a. Create the PortChannel on switch2 with the **channel-group** command. Create a description for the port with the **switchport description** command.

```
switch2# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch2(config)# interface fc1/1, fc2/1
switch2(config-if)# channel-group 2
fc1/1 fc2/1 added to port channel 2 and disabled
please do the same operation on the switch at the other end of the port channel,
then do "no shutdown" at both ends to bring them up
switch2(config-if)# switchport description "To switch1 PortChannel2"
```

- b. Enable trunking (TE) and set the VSAN allowed list on switch1.

```
switch2# config terminal
switch2(config)# int port channel 2
switch2(config-if)# switchport trunk mode on
switch2(config-if)# switchport trunk allowed vsan 1
switch2(config-if)# switchport trunk allowed vsan add 10
```

**Step 3** Enable the interfaces to bring up the PortChannel.

- a. Enable switch1 interfaces with the **interface** command.

```
switch1# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch1(config)# interface fc1/1, fc2/1
switch1(config-if)# no shut
```

- b. Enable switch2 interfaces with the **interface** command.

```
switch2# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch2(config)# interface fc1/1, fc2/1
switch2(config-if)# no shut
```

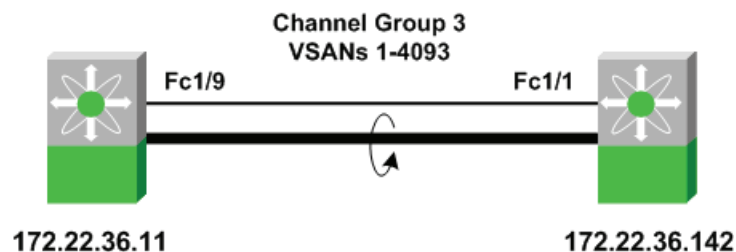
**Step 4** Verify that the PortChannel has come up using the **show interface port channel** command.

```
switch1# show interface port channel 2
port channel 2 is trunking
 Port description is To switch2 PortChannel2
 Hardware is Fibre Channel
 Port WWN is 24:02:00:0c:85:e9:d2:c0
 Admin port mode is E, trunk mode is on
 Port mode is TE
 Port vsan is 1
 Speed is 4 Gbps
 Trunk vsans (admin allowed and active) (1,10)
 Trunk vsans (up) (1,10)
 Trunk vsans (isolated) ()
 Trunk vsans (initializing) ()
 5 minutes input rate 64 bits/sec, 8 bytes/sec, 0 frames/sec
 5 minutes output rate 56 bits/sec, 7 bytes/sec, 0 frames/sec
 78296342 frames input, 72311141128 bytes
 0 discards, 0 errors
 0 CRC, 0 unknown class
 0 too long, 0 too short
 56299070 frames output, 26061293700 bytes
 0 discards, 0 errors
 0 input OLS, 2 LRR, 0 NOS, 0 loop inits
 4 output OLS, 2 LRR, 0 NOS, 0 loop inits
 Member[1] : fc1/2
 Member[2] : fc2/1
 iSCSI authentication: None
```

## Adding a New Member to a PortChannel Using Fabric Manager

This recipe adds a new member to a PortChannel using Fabric Manager. The topology shown in [Figure 5-7](#) is used in this example.

Figure 5-7 PortChannel Expansion with FM Topology

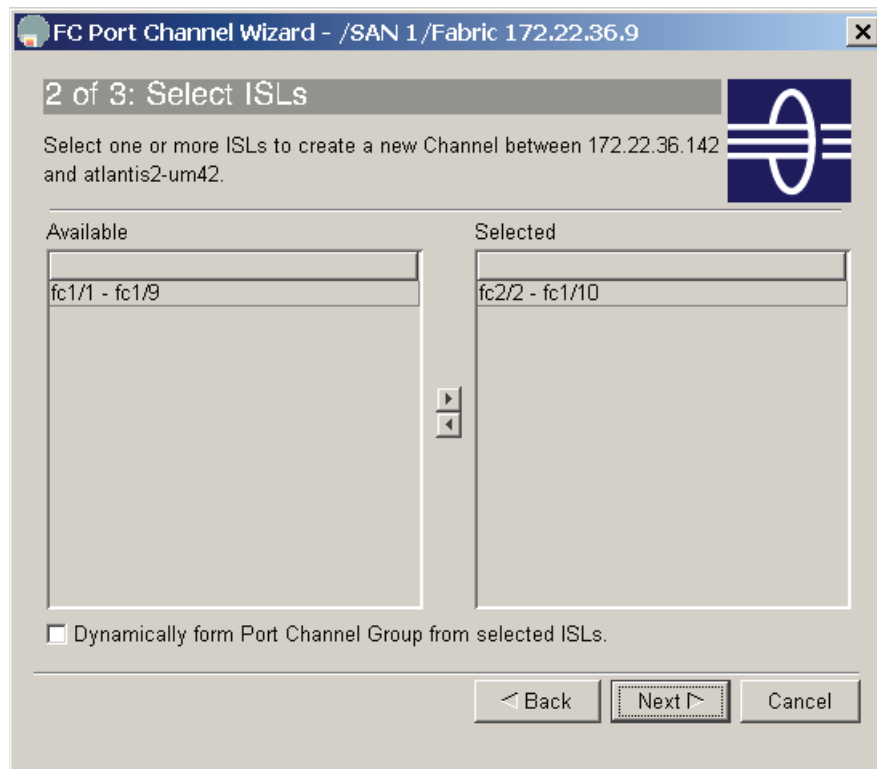


To add a new member to a PortChannel using Fabric Manager, open the topology map and follow these steps:

**Step 1** On the map, right-click the first link to be added to a PortChannel.

**Step 2** Click **Edit...** You see the screen in [Figure 5-8](#).

Figure 5-8 Create PortChannel Wizard



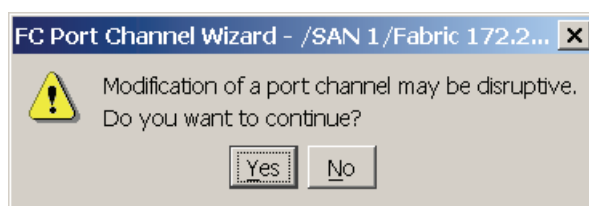
Existing PortChannel members appear in the Selected column while available candidate links appear in the Available column.

**Step 3** Move the available ISL (fc1/1 - fc1/9) into the Selected column.

**Step 4** Click **Finish...**

A warning is displayed concerning modifying a PortChannel. Because we are adding a link, the PortChannel is not affected. However, the ISL is cycled while FSPF continues to route traffic over the previously created PortChannel.

Figure 5-9 PortChannel Modification Warning



**Step 5** Click **Yes**.

On the Fabric Manager map, the link selected to be added to the PortChannel momentarily goes down (represented by a red X) then disappears from the map, leaving only the PortChannel represented by a thick line.

## Adding New Members to a PortChannel from the CLI

This recipe adds a new member to a PortChannel using the CLI. This example uses the following resources:

- Switches 1 and 2
- Existing interfaces fc1/1 and fc2/1
- New interface fc3/1

To add a new member to a PortChannel from the CLI, follow these steps:

- Step 1** Use the **force** keyword with the **channel-group** command to add the new member to switch1. This makes a new link inherit the parameters of the existing links in channel group 2.

```
switch1# conf t
switch1(config)# int fc3/1
switch1(config-if)# channel-group 2 force
fc3/1 added to port channel 2 and disabled
please do the same operation on the switch at the other end of the port channel,
then do "no shutdown" at both ends to bring them up
switch1(config-if)# no shut
```

- Step 2** Use the **force** keyword with the **channel-group** command to add the new member to switch2. This makes a new link inherit the parameters of the existing links in channel group 2.

```
switch2# conf t
switch2(config)# int fc3/1
switch2(config-if)# channel-group 2 force
fc3/1 added to port channel 2 and disabled
please do the same operation on the switch at the other end of the port channel,
then do "no shutdown" at both ends to bring them up
switch3(config-if)# no shut
```

- Step 3** Verify that the PortChannel now has three members. Use the **show interface port channel** command.

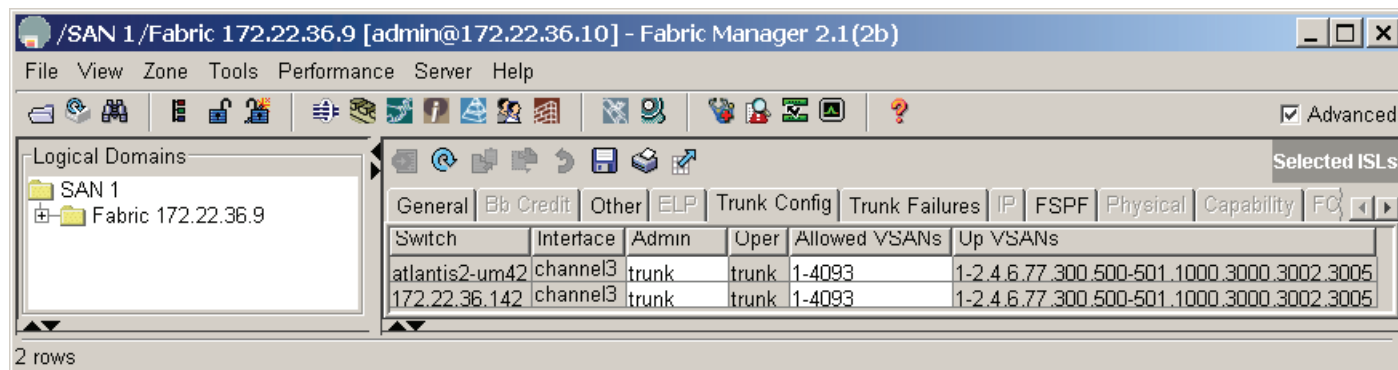
```
switch1# show interface port channel 2
port channel 2 is trunking
 Port description is To switch2 PortChannel2
 Hardware is Fibre Channel
 Port WWN is 24:02:00:0c:85:e9:d2:c0
 Admin port mode is E, trunk mode is on
 Port mode is TE
 Port vsan is 1
 Speed is 6 Gbps
 Trunk vsans (admin allowed and active) (1,10)
 Trunk vsans (up) (1,10)
 Trunk vsans (isolated) ()
 Trunk vsans (initializing) ()
 5 minutes input rate 64 bits/sec, 8 bytes/sec, 0 frames/sec
 5 minutes output rate 56 bits/sec, 7 bytes/sec, 0 frames/sec
 78296342 frames input, 72311141128 bytes
 0 discards, 0 errors
 0 CRC, 0 unknown class
 0 too long, 0 too short
 56299070 frames output, 26061293700 bytes
 0 discards, 0 errors
 0 input OLS, 2 LRR, 0 NOS, 0 loop inits
 4 output OLS, 2 LRR, 0 NOS, 0 loop inits
 Member[1] : fc1/2
 Member[2] : fc2/1
 Member[3] : fc3/1
 iSCSI authentication: None
```

## Modifying the VSAN Allowed List on a PortChannel Using Fabric Manager

To modify the VSAN allowed list for a PortChannel in Fabric Manager, follow these steps (this is the same procedure as the one used to modify a standard trunking E port (TE port)):

- Step 1** In Fabric Manager, right-click the PortChannel displayed in the map pane.
- Step 2** Click **Interface Attributes**.
- Step 3** Choose the **Trunk Config** tab. You see the table in the top pane shown in [Figure 5-10](#).

Figure 5-10 Modify VSAN Allowed List for a PortChannel



- Step 4** Modify the **Allowed VSANs** column for both rows, as each row represents the configuration of the PortChannel on each switch.
- Step 5** Click **Apply Changes**.

## Modifying the VSAN Allowed List on a PortChannel From the CLI

The following example modifies the VSAN allowed list for a PortChannel and adds VSAN 17 to PortChannel 2 with the **switchport trunk allowed** command. (This is the same process used for a standard, single link TE port.)

```
switch2# config terminal
switch2(config)# int port channel 2
switch2(config-if)# switchport trunk allowed vsan add 17
```

Remove VSAN 17 from PortChannel 2 using the **no switchport trunk allowed** command.

```
switch2# config terminal
switch2(config)# int port channel 2
switch2(config-if)# no switchport trunk allowed vsan add 17
```



## CHAPTER 6

# VSANs

---

A virtual SAN (VSAN) is a logical grouping of ports in a single switch or across multiple switches that function like a single fabric. A VSAN is isolated from other VSANs in terms of traffic, security, and fabric services. Because of this, changes made to one VSAN do not affect the remaining VSANs, even though they may be present in the same physical SAN infrastructure hardware. Using VSANs, multiple logical SANs can be hosted on a physical SAN hardware infrastructure. A VSAN lends itself to SAN island consolidation on a higher port density physical switch, along with traffic isolation and increased security. Once a VSAN is created, it has all the properties and functions of a SAN.

Multiple VSANs can be defined on a physical switch. Each VSAN will require its own domain\_ID. A single VSAN can span 239 physical switches (a Fibre Channel standards limit). At the current time, a maximum of 256 VSANs are supported in a physical switch.

Using VSANs provides some important advantages:

- VSAN traffic stays within the VSAN boundaries. Devices can be part of just one VSAN.
- VSANs allow you to create multiple logical SAN instances on top of a physical SAN infrastructure. This allows for the consolidation of multiple SAN islands onto a physical infrastructure, which minimizes the hardware that needs to be managed.
- Each VSAN has its own set of fabric services, which allows the SAN infrastructure to be scalable and highly available.
- Additional SAN infrastructure resources such as VSAN ports can be added and changed as needed without impacting VSAN ports that are already a part of the SAN infrastructure. Moving ports between VSANs is as simple as assigning the port to a different VSAN.

VSANs are numbered from 1 through 4094. VSAN 1 and VSAN 4094 are predefined and have very specific roles. The user-specified VSAN range is from 2 through 4093. VSAN 1 is the default VSAN that contains all ports by default. VSAN 1 is used as a management VSAN. VSAN 4094 is the isolated VSAN into which all orphaned ports are assigned. Devices that are part of VSAN 4094 cannot communicate with each other.



**Note**

---

We recommend using VSAN 1 as a management VSAN and not as a production VSAN.

---

# Creating a VSAN and Adding Interfaces Using Fabric Manager

This recipe creates a VSAN (3005) and adds an interface to it.



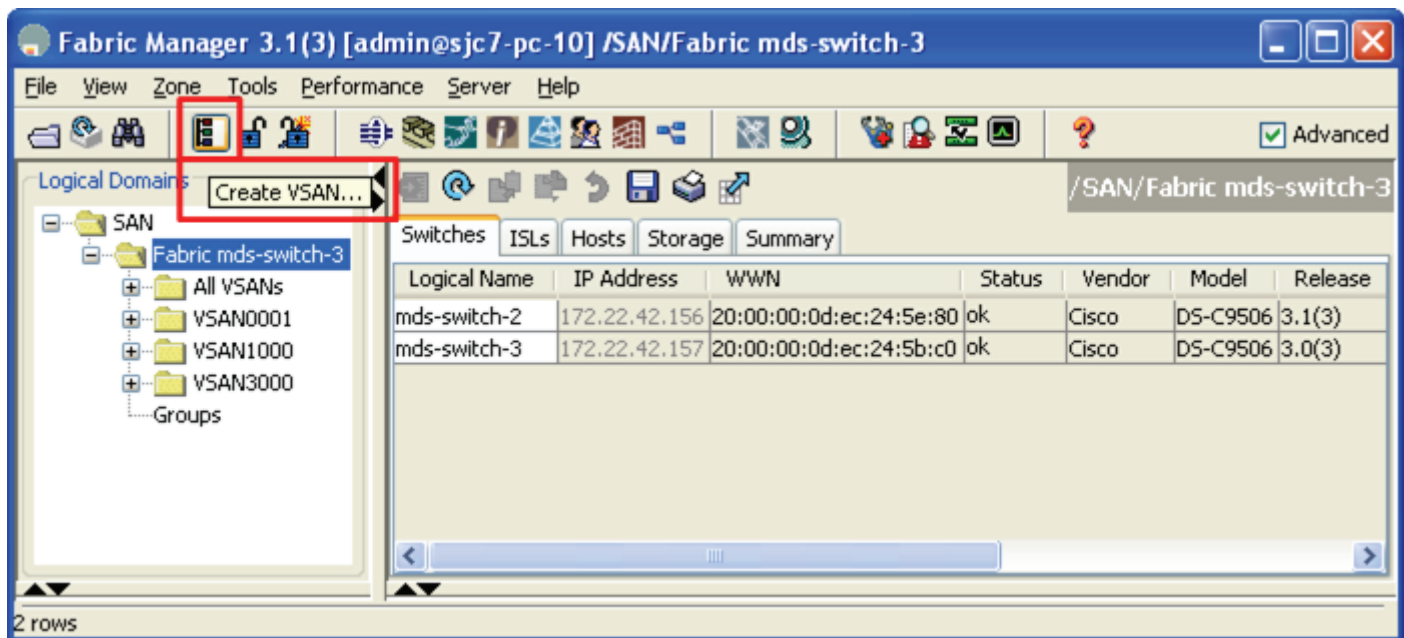
## Note

Moving a port from one VSAN to another does not change its configuration (F, FL, TL), its speed, or its administrative state (shut/noshut). However, any device attached to the port needs to do a fabric login (FLOGI) back into the switch.

To create a VSAN from Fabric Manager, follow these steps:

- Step 1** On the toolbar, click the **Create VSAN** icon (see [Figure 6-1](#)).

Figure 6-1 Create VSAN on Fabric Manager



- Step 2** Select the switch(es) for the VSAN.
- Step 3** Enter the VSAN id in **VSAN ID** dialog box (see [Figure 6-2](#)). In this recipe VSAN 3500 is being created.



Figure 6-2 VSAN Wizard

**Create VSAN - /SAN/Fabric mds-switch-3**

Switches:

- ☒ mds-switch-2
- ☒ mds-switch-3

VSAN Id:  1..4093

Name:  (blank=default)

LoadBalancing: ☐ srcId/DestId ☒ srcId/DestId/OxId

InterOperValue: ☒ default ☐ Interop-1 ☐ Interop-2 ☐ Interop-3 ☐ Interop-4

AdminState: ☒ active ☐ suspended

☐ Static Domain Ids:

☐ FICON

☐ Enable Fabric Binding for Selected Switches

☐ All Port Prohibited

**Step 4** Enter a name for the VSAN. If nothing is entered, then default name of VSAN3500 is assigned to the VSAN.

**Step 5** If the VSAN is to be attached to a third-party switch, select the appropriate Interop Mode.



**Note**

You can specify a static domain\_ID for a VSAN at the time of creation. Otherwise, you can follow the recipe in [Converting an Existing VSAN to Static Domain ID and Enabling a Persistent FCID Using the CLI](#), page 6-13 to specify the new domain\_ID.

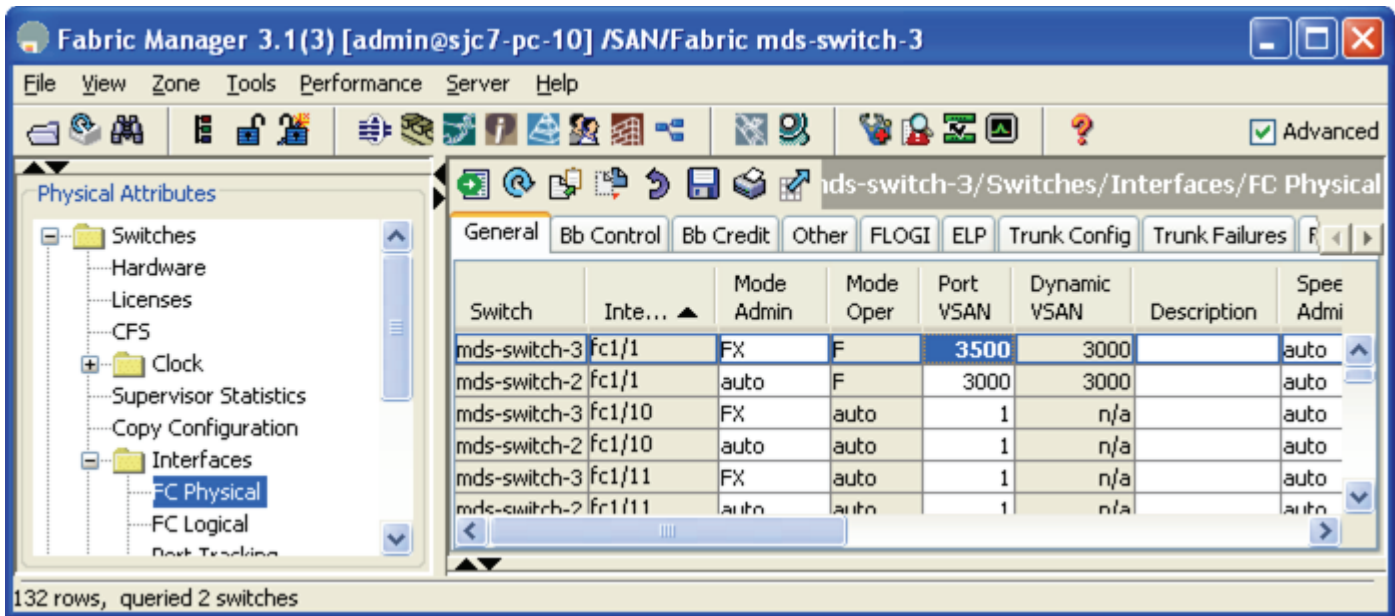
**Step 6** Click **Create** to create VSAN 3500 on the above switches.

To add interfaces to the new VSAN in Fabric Manager, follow these steps:

**Step 1** In the **Physical Attributes** pane, expand **Switches > Interfaces > FC Physical**. (See [Figure 6-3](#).)

**Step 2** Modify the **Port VSAN** field for the switch interface to be moved to the specified VSAN. (See [Figure 6-3](#).)

Figure 6-3 Change Port VSAN Membership



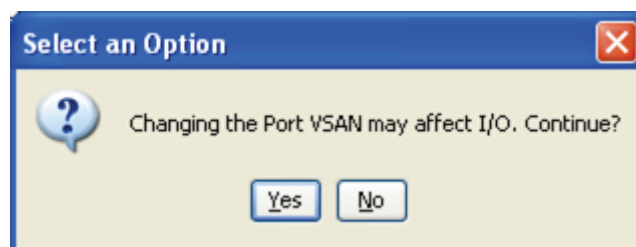
**Step 3** (Optional) If the port has the Admin Status **down**, enable it by changing the value to **up** in the **status Admin** Field.

**Step 4** Click **Apply Changes...**

A warning is displayed that moving a port between two VSANs can be disruptive to that port as it will have to log in again to the fabric and will no longer have access to resources in the previous VSAN.

**Step 5** Click **Yes** to move the port to the new VSAN. (see [Figure 6-4](#))

Figure 6-4 Move Port to New VSAN Confirmation



# Modifying VSAN Attributes with Fabric Manager

These recipes modify the attributes of a VSAN using Fabric Manager. The attributes of a VSAN include:

- VSAN name
- Load balancing (src/dst or src/dst/ox-id)
- Administrative state (suspended or active)
- Interoperability mode to work with third-party switches
- Order delivery

To modify the attributes of a VSAN, follow these steps:

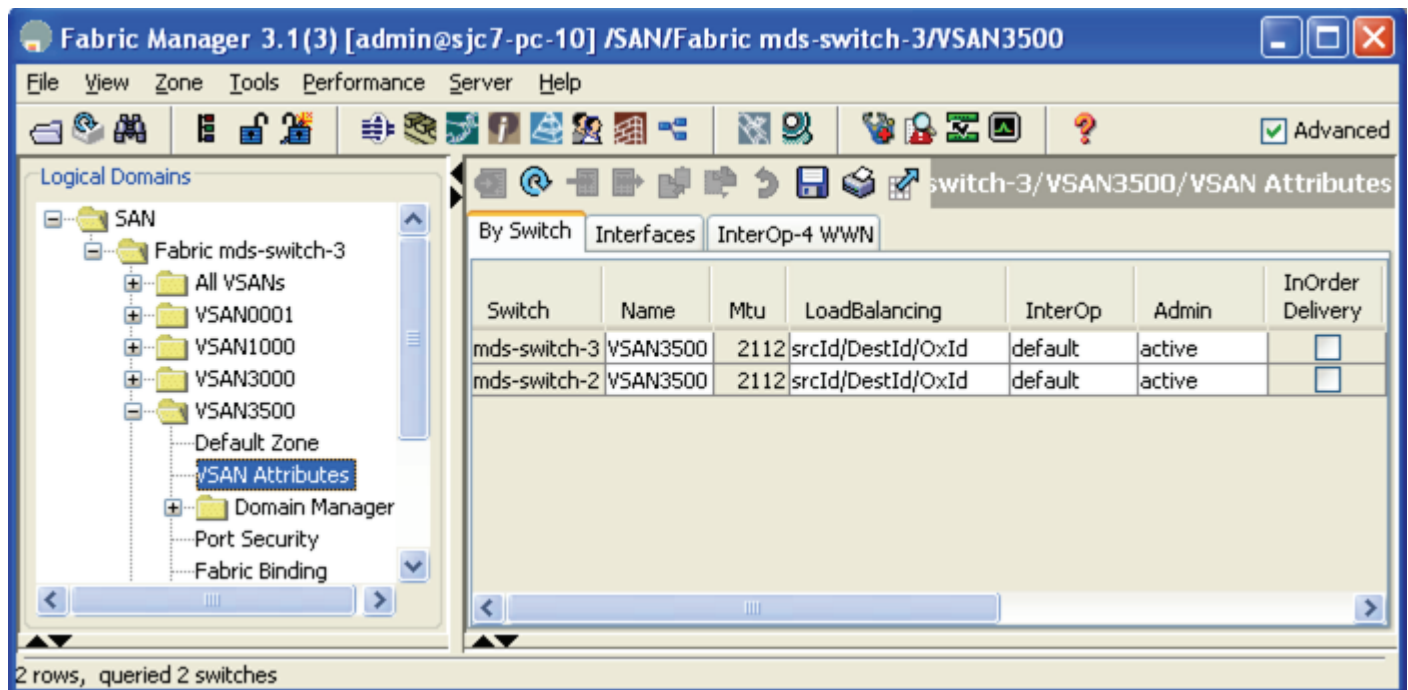
- Step 1** Select a fabric in the Logical Domains pane.
- Step 2** Expand the VSAN to be modified (VSAN 3500 in [Figure 6-5](#)).
- Step 3** Choose **VSAN Attributes**.
- Step 4** Make changes to the desired fields.



**Note** Standard editing keyboard shortcuts (Ctrl+X to cut, Ctrl+C to copy, and Ctrl+V to paste) can be used to edit the text fields.

- Step 5** Click **Apply Changes....**

Figure 6-5 Modify VSAN Attributes



## Changing the Domain ID and Its Configuration of VSAN Using Fabric Manager

Within a VSAN, the domain manager process on the principal switch in a fabric is responsible for assigning a domain ID to a switch joining the fabric. When a switch boots up or joins a new fabric, it can request a specific domain ID or take any available domain ID.

A domain ID can be configured in two ways:

- **Preferred** — The new switch requests a specific domain ID. However, if it receives a different domain ID, it accepts it.
- **Static** — The new switch requests a specific domain ID. If it receives a different domain ID, it isolates itself from the fabric. Use static domain IDs when the same domain ID must be maintained under all circumstances.

After obtaining the domain ID from the principal switch in the VSAN, the local switch assigns Fibre Channel Identifiers (FC IDs) to each end device as they log into the fabric. This process is known as FLOGI.



Tip

HPUX and AIX are two operating systems that use a FC ID in the device path to storage. For the switch to always assign the same FC ID to a device across switch reboots, configure a persistent FC ID and static domain ID for the VSAN. If an FC ID changes for a device accessed by either an AIX or a HPUX host, the host may lose access to the device.

By default, the switch assigns the same FC ID to a device. However, if the switch is rebooted this database of port world-wide name (WWN)/FC ID mapping is not maintained. Enabling persistent FC IDs will make this database persistent across reboots.

A persistent FC ID can be configured two ways:

- **Dynamic** — The FC ID is determined and assigned by the switch and if the persistent FC ID database is manually purged by the user this entry will be deleted. These entries are persistent across reboots of the switch and are VSAN specific.
- **Static** — The FC ID is determined by the user before attaching the device to the switch. If the persistent FC ID database is manually purged by the user these entries will not be removed. These entries are persistent across reboots of the switch and are VSAN-specific.

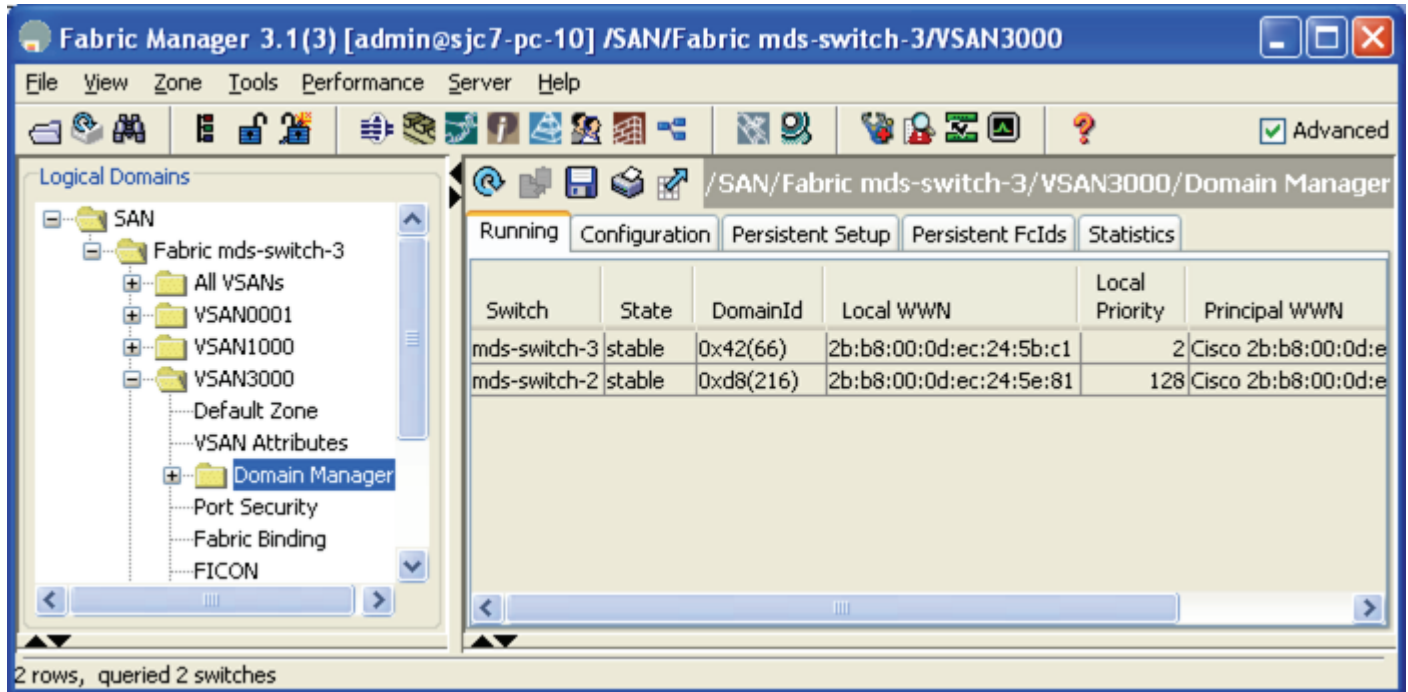
When a persistent FC ID is enabled, the switch makes persistent all of the devices in that VSAN, and the administrator is not required to manually type in devices entering that VSAN.

The recipe below shows changing the domain\_ID for VSAN 3000 on switches mds-switch-2 and mds-switch-3 and setting the FC ID to be persistent for the same VSAN.

To change the domain\_ID for a VSAN and set the FC ID to be persistent for the same VSAN, follow these steps:

- Step 1** Select a fabric in the Logical Domains pane.
- Step 2** Expand the **VSAN** whose attribute needs to be modified and select **Domain Manager**. (This brings up the current domain\_ID of the VSAN 3000. See [Figure 6-6](#).)

Figure 6-6 Current Domain ID of the VSAN



**Step 3** Select the **Configuration** tab on the right-hand side. Then edit the domain ID of the required switch in the **config domainid** field. (See Figure 6-7).

**Step 4** Change the domain ID to the required value, and then in the pull-down menu from the **Config Type** field, change it from **preferred** to **static**. (See Figure 6-7.)

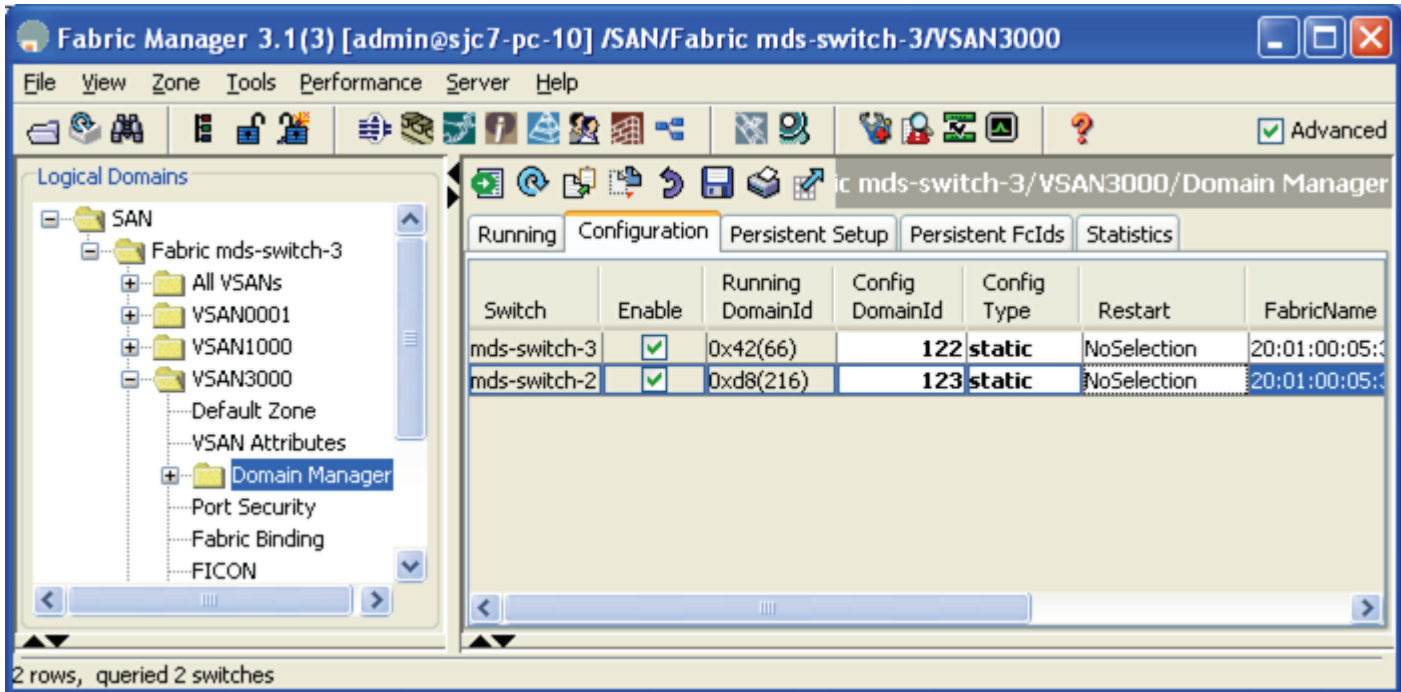
**Note**

The domain IDs can range between 1 through 239 in a Fibre Channel environment. When a VSAN spans multiple switches, the domain ID for the VSAN in each switch that it spans also has to be unique.

**Caution**

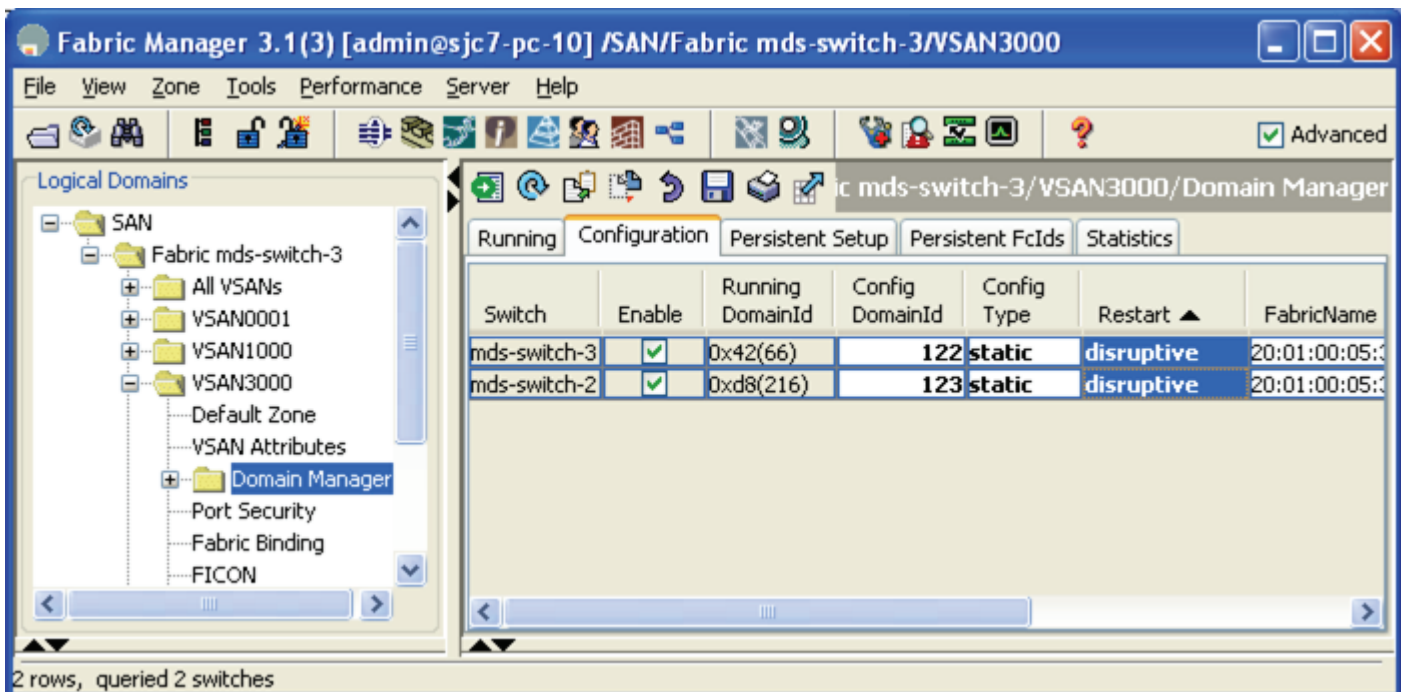
Changing domain IDs and therefore FC IDs for a device is disruptive, as an end device has to log in again to the fabric (FLOGI) to obtain a new FC ID. However, making a domain ID static without changing its value is not disruptive.

Figure 6-7 Modify the Domain ID for VSAN 3000



**Step 5** In the **Restart** field select **disruptive**. (See Figure 6-8.)

Figure 6-8 Restart the VSAN to Apply Changes





**Note**

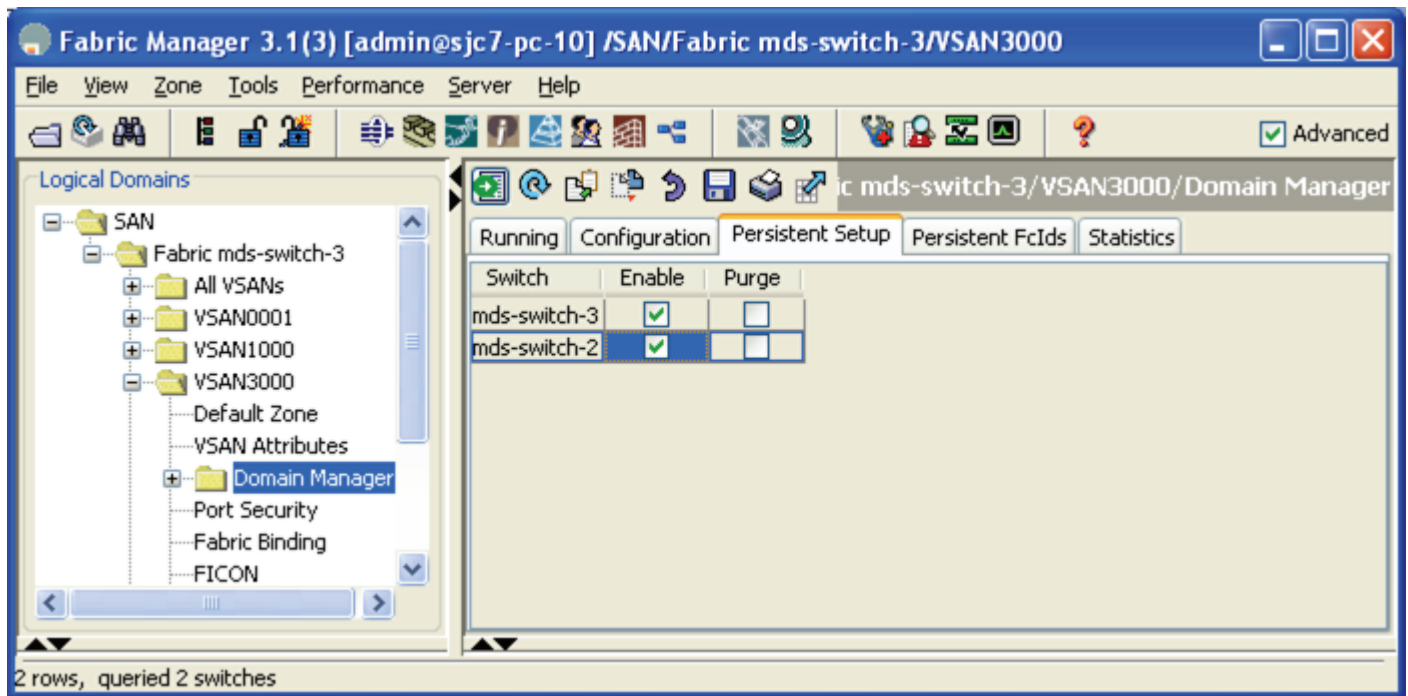
When the domain ID of a VSAN changes, a disruptive restart for that VSAN is required to change its domain ID. This forces all the device to log out and log back in. At that time, new FC IDs are allocated to all the devices in that VSAN.

**Step 6** Click **Apply Changes** to apply the configuration.

## Changing the FCID Configuration of VSAN Using Fabric Manager

- Step 1** Select a fabric in the Logical Domains pane.
- Step 2** Expand the **VSAN** whose attribute needs to be modified and select **Domain Manager**. (This brings up the current domain\_ID of the VSAN 3000.) See [Figure 6-6](#).
- Step 3** Select the **Persistent Setup** tab on the right-hand side. Check the **enable** check box for all the switches visible here. This enables a persistent FC ID for the VSAN on all the switches that the VSAN currently resides on. (See [Figure 6-9](#).)

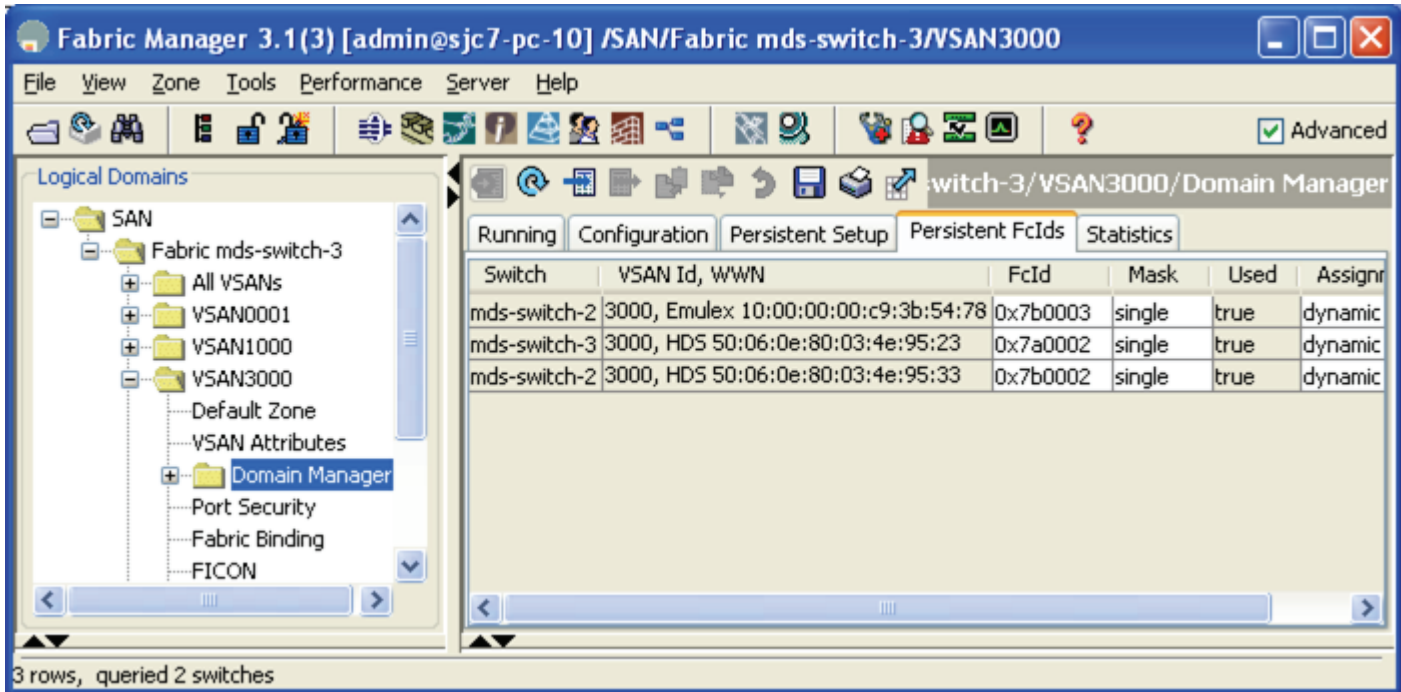
Figure 6-9 Enable Persistent FC ID



**Step 4** Click **Apply changes** to apply the changes.

At this point, the domain ID is statically set and FC IDs will remain persistent across reboots for VSAN 3500 on the switches mds-switch-2 and mds-switch-3. The persistent FC ID database can be viewed in the **Persistent FCIDs** tab (see [Figure 6-10](#)).

Figure 6-10 Persistent FC ID Database



## Modifying VSAN Attributes with the CLI

These recipes modify the attributes of a VSAN using the CLI. This includes interop modes, load balancing, and setting static domain\_IDs and persistent FC IDs.

### Creating a VSAN on a Single Switch and Adding an Interface

Create and name a VSAN on a single switch with the **vsan name** command. This example creates VSAN 200 with the name TapeVSAN and adds fibre channel interface fc 1/1 is with the **vsan interface** command.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# vsan database
mds-switch-1(config-vsan-db)# vsan 3000 name VSAN3000
mds-switch-1(config-vsan-db)# vsan 3000 interface fc 1/1
mds-switch-1(config-vsan-db)# ^Z
mds-switch-1#
```

### Setting VSAN Interop Mode

Set Interop mode for VSANs that need to interact with other third-party switches. Use different Interop modes under different circumstance as shown in [Figure 6-11](#).



Figure 6-11 Interop Modes

| Interop Mode | When to use it                                                                                                                        |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------|
| Mode 1       | Required when all vendor switches are set in their respective interop modes. In interop mode 1, only Domain IDs 97 to 127 are allowed |
| Mode 2       | Required when VSAN has to work with a Brocade 2800/3800 switch in native corePID 0 mode                                               |
| Mode 3       | Required when the VSAN has to work with a Brocade switch running in corePID 1 mode                                                    |
| Mode 4       | Required when the VSAN has to work with a Mcdatal switch. Only domain IDs 1 through 31 are allowed.                                   |

For more information, refer to the [Cisco MDS 9000 Family Switch-to-Switch Interoperability Configuration Guide](#) on [www.cisco.com](#). Consult this manual before doing interoperability tasks; it explains the different Interop modes.

The following examples set Interop modes to 1, 2, and 3 for a VSAN.

## Interop Mode 1

Interop mode 1 is required when all vendor switches are set in their respective interop modes. Ensure that the domain\_ID of the VSAN is between 97 and 127 for mode 1 to work. Change the interop mode with the **vsan interop** command.

```
mds-switch-1# conf t
mds-switch-1(config)# vsan database
mds-switch-1(config-vsan-db)# vsan 200 interop 1
mds-switch-1(config-vsan-db)# ^Z
mds-switch-1#
```

## Interop Mode 2

Interop mode 2 is required when a VSAN has to work with a Brocade 2800/3800 switch in native corePID 0 mode. Change the interop mode with the **vsan interop** command.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# vsan database
mds-switch-1(config-vsan-db)# vsan 200 interop 2
mds-switch-1(config-vsan-db)# ^Z
mds-switch-1#
```

## Interop Mode 3

Interop mode 3 is required when a VSAN has to work with a Brocade switch running in corePID 1 mode. Change the interop mode with the **vsan interop** command.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# vsan database
mds-switch-1(config-vsan-db)# vsan 200 interop 3
mds-switch-1(config-vsan-db)# ^Z
mds-switch-1#
```

## Interop Mode 4

Interop mode 4 is required when a VSAN has to work with a McData switch. Change the interop mode with the **vsan interop** command.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# vsan database
mds-switch-1(config-vsan-db)# vsan 200 interop 4
mds-switch-1(config-vsan-db)# ^Z
mds-switch-1#
```

**Note**

Besides setting the VSAN in interop mode 4, additional configurations are required to successfully bring up the VSAN to operate in interop mode 4. Refer to the *Cisco MDS 9000 Family Switch-to-Switch Interoperability Configuration Guide* on [www.cisco.com](http://www.cisco.com) for more comprehensive instructions.

## Changing the Load-Balancing Scheme

Configure the load-balancing scheme with VSAN S\_ID (source id), D\_ID (destination id)-based load-balancing, and the exchange level (S\_ID, D\_ID, OX\_ID) on the switch.

These recipes configure load-balancing for VSAN 200.

### Sequence Level Load-Balancing (Source\_ID, Destination\_ID)

Change the load-balancing scheme for VSAN 200 to S\_ID, D\_ID mode with the **vsan loadbalancing** command.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# vsan database
mds-switch-1(config-vsan-db)# vsan 200 loadbalancing src-dst-id
mds-switch-1(config-vsan-db)# ^Z
mds-switch-1#
```

### Exchange Level Load-Balancing (S\_ID, D\_ID, OX\_ID)

Change the load-balancing scheme for VSAN 200 to S\_ID, D\_ID, and OX\_ID modes with the **vsan loadbalancing** command. This is the default load-balancing scheme.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# vsan database
mds-switch-1(config-vsan-db)# vsan 200 loadbalancing src-dst-ox-id
mds-switch-1(config-vsan-db)# end
mds-switch-1#
```

## Converting an Existing VSAN to Static Domain ID and Enabling a Persistent FCID Using the CLI

This recipe configures a static domain ID for a VSAN and enables a persistent FC ID for VSAN 3000 on switch mds-switch-1.

In this recipe an existing VSAN (3000) on switch mds-switch-1 with domain ID 239 is statically configured and the persistent FC ID is enabled. This recipe does not alter the running domain ID.

To configure a static domain ID for a VSAN and enable a persistent FC ID for a VSAN, follow these steps:

- 
- Step 1** Display the current domain ID for VSAN 3000 using the command **show domain-list**.

```
mds-switch-1# show fcdomain domain-list vsan 3000
Number of domains: 2
Domain ID WWN

0xef(239) 2b:b8:00:05:30:00:68:5f [Local] [Principal]
```

- Step 2** Configure the static domain\_ID with the **domain static** command.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# fcdomain domain 239 static vsan 3000
```

**Step 3** Enable persistent FC ID with **fcid persistent**.

```
mds-switch-1(config)# fcdomain fcid persistent vsan 3000
mds-switch-1(config)# end
```

**Step 4** Save the configuration.

```
mds-switch-1# copy running-config startup-config
[#####] 100%
```

**Note**

If the domain ID of VSAN 200 is different from what is currently running (22 in this case), then the VSAN has to be restarted before configuration changes to the domain ID and FC ID persistence can take effect.

**Caution**

Changing domain IDs (and therefore FC IDs) for a device is disruptive, because an end device has to log in again to the fabric to obtain a new FC ID. However, making a domain ID static without changing its value is not disruptive.

## Changing the Domain ID in a VSAN and Making It Static

Sometimes the VSAN on a switch needs a specific domain ID for various operational requirements. If the domain ID of the VSAN needs to be changed from its current running domain ID, the VSAN has to be restarted for the new domain ID to take effect.

To configure the domain ID of VSAN 200 from its running domain ID 145 to 229, follow these steps:

**Step 1** List the current domain\_ID of VSAN 200.

```
mds-switch-1# show fcdomain domain-list vsan 200

Number of domains: 1
Domain ID WWN

0x91(145) 20:c8:00:0d:ec:24:5e:c1 [Local] [Principal]
mds-switch-1#
```

**Step 2** Change the domain\_ID of VSAN 200 to 229 and make it static.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# fcdomain domain 229 static vsan 200
```

**Step 3** Restart VSAN 200 for the new domain\_ID to take effect using **vsan suspend** command.

```
mds-switch-1(config)# vsan database
mds-switch-1(config-vsan-db)# vsan 200 suspend
mds-switch-1(config-vsan-db)# no vsan 200 suspend
mds-switch-1(config-vsan-db)# end
```

**Step 4** List the current running domain\_ID of the VSAN 200.

```
mds-switch-1# sh fcdomain domain-list vsan 200
```

```

Number of domains: 1
Domain ID WWN

0xe5(229) 20:c8:00:0d:ec:24:5e:c1 [Local] [Principal]
mds-switch-1#

```

## Assigning a Predetermined FC ID to a pWWN

When performing a migration or host bus adapter (HBA) replacement, the FC ID may need to be reassigned to the new pWWN. This recipe assigns a predetermined FC ID to a specific pWWN.



### Note

A new FC ID cannot be assigned to a pWWN that is logged into the fabric. Before assigning a new FC ID, log the device out of the fabric. You can log out the device by shutting down the Fibre Channel interface.

FC ID 0x160000 will be assigned to pWWN 50:06:0b:82:bf:d1:db:cd permanently. When the pWWN logs into the switch (FLOGI), it receives this assigned FC ID.



### Note

The FC ID to be assigned (0x160000) should contain the same domain\_ID (0x16) as the currently running domain in the VSAN.

To assign a predetermined FC ID to a specific pWWN, follow this example:

```

mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# fcdomain fcid database
mds-switch-1(config-fcid-db)# vsan 22 wwn 50:06:0b:82:bf:d1:db:cd fcid 0x160000 dynamic
mds-switch-1(config-fcid-db)# ^Z
mds-switch-1#

```



### Note

If the device is currently logged in, you cannot change the FC ID of the device. To change the FCID, the switch port to which it is connected has to be shut down and the current FC ID, if persistent, needs to be purged from the FC ID database and the new FC ID for that interface has to be configured. This procedure for a currently logged in device is disruptive for the device only.

## Assigning a New Predetermined FCID to a Currently Logged In pWWN

In the following recipe describes the steps required to change the FC ID of the device that is logged into the fabric. The device with pWWN 50:06:0e:80:03:4e:95:33 has a FCID 0x7b0002 and needs to be changed to 0x7b0010.

To change the FC ID of a device currently logged in to the fabric, follow these steps:

- Step 1** Display the port on the switch that the device is connected to and its current FCID on the switch to which the device is connected.
- Step 2** Shut down the interface on the switch to which the device is connected.
- Step 3** Purge the FC ID database for this device.
- Step 4** Configure the new FC ID for the device.

**Step 5** Enable the port.

**Step 6** Use the **show FLOGI database** command to see the new FC ID of the logged in port.

```
mds-switch-2# show flogi database vsan 3000
```

| INTERFACE | VSAN | FCID     | PORT NAME                                    | NODE NAME               |
|-----------|------|----------|----------------------------------------------|-------------------------|
| fc1/1     | 3000 | 0x7b0002 | 50:06:0e:80:03:4e:95:33<br>[storage-CL4D]    | 50:06:0e:80:03:4e:95:33 |
| fc1/9     | 3000 | 0x7b0003 | 10:00:00:00:c9:3b:54:78<br>[sjc7-pc-9-emplx] | 20:00:00:00:c9:3b:54:78 |

Total number of flogi = 2.

```
mds-switch-2#
```

```
mds-switch-2# conf t
```

Enter configuration commands, one per line. End with CNTL/Z.

```
mds-switch-2(config)# interface fc 1/1
```

```
mds-switch-2(config-if)# shutdown
```

```
mds-switch-2(config-if)# exit
```

```
mds-switch-2(config)# fcdomain fcid database
```

```
mds-switch-2(config-fcid-db)# no vsan 3000 wwn 50:06:0e:80:03:4e:95:33 <-- clears
persistent FCID
```

```
mds-switch-2(config-fcid-db)# end
```

```
mds-switch-2# show fcdomain fcid persistent vsan 3000 <-- list he <-- lists the current
persistent FCIDs in VSAN 3000
```

Total entries 1.

Persistent FCIDs table contents:

| VSAN | WWN                     | FCID     | Mask        | Used | Assignment |
|------|-------------------------|----------|-------------|------|------------|
| 3000 | 10:00:00:00:c9:3b:54:78 | 0x7b0003 | SINGLE FCID | YES  | DYNAMIC    |

```
mds-switch-2# config t
```

```
mds-switch-2(config)# fcdomain fcid database
```

```
mds-switch-2(config-fcid-db)# vsan 3000 wwn 50:06:0e:80:03:4e:95:33 fcid 0x7b0010 dynamic
```

```
mds-switch-2(config-fcid-db)# end
```

```
mds-switch-2# show fcdomain fcid persistent vsan 3000 <-- lists the persistent FCID fro
VSAN 3000
```

Total entries 2.

Persistent FCIDs table contents:

| VSAN | WWN                     | FCID     | Mask        | Used | Assignment |
|------|-------------------------|----------|-------------|------|------------|
| 3000 | 50:06:0e:80:03:4e:95:33 | 0x7b0010 | SINGLE FCID | NO   | DYNAMIC    |
| 3000 | 10:00:00:00:c9:3b:54:78 | 0x7b0003 | SINGLE FCID | YES  | DYNAMIC    |

```
mds-switch-2#
```

```
mds-switch-2# conf t
```

Enter configuration commands, one per line. End with CNTL/Z.

```
mds-switch-2(config)# interface fc 1/1
```

```
mds-switch-2(config-if)# no shutdown
```

```
mds-switch-2(config-if)# exit
```

```
mds-switch-2# show flogi database
```

| INTERFACE | VSAN | FCID     | PORT NAME                                    | NODE NAME               |
|-----------|------|----------|----------------------------------------------|-------------------------|
| fc1/1     | 3000 | 0x7b0010 | 50:06:0e:80:03:4e:95:33<br>[storage-CL4D]    | 50:06:0e:80:03:4e:95:33 |
| fc1/9     | 3000 | 0x7b0003 | 10:00:00:00:c9:3b:54:78<br>[sjc7-pc-9-emplx] | 20:00:00:00:c9:3b:54:78 |

Total number of flogi = 2.

```
mds-switch-2#
```



# CHAPTER 7

## Zoning

Zones are the basic form of data path security in a Fibre Channel environment. Zones are used to define which end devices (two or more) in a fabric can communicate with each other. Zones are grouped together into zone sets. For the zones to be active, the zone set to which the zones belong needs to be activated. Individual zone members can be part of multiple zones. Zones can be part of multiple zone sets. Multiple zone sets can be defined in a fabric. At any given time, only one zone set can be active.

If zoning is not activated in a fabric, all the end devices are part of the default zone. If zoning is activated, any end devices that are not part of an active zone are part of the default zone. The default zone policy is set either to deny (none of the end devices that are part of the default zone can communicate with each other) or permit (all the devices that are part of the default zone can communicate with each other).

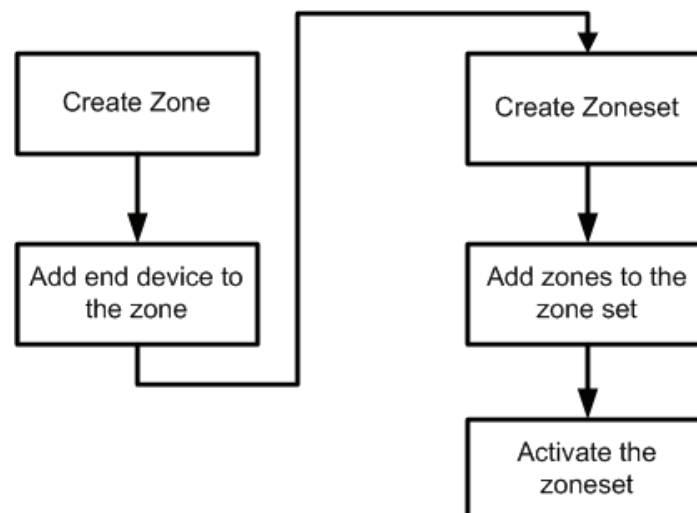


### Note

We recommend that you set the default zone policy to deny. This is the default setting on MDS 9000 series switches and directors.

Figure 7-1 shows the basic zoning flow.

Figure 7-1 Basic Zoning Flow



The MDS series products support two modes of zoning: basic zoning (FC-GS-3) and enhanced zoning (FC-GS-4). Both basic zoning and enhanced zoning (introduced in Cisco SAN-OS Release 2.0) use the concepts of zones and zone sets.

# Enhanced Zoning

Enhanced zoning, introduced in Cisco SAN-OS Release 2.0, was defined in the FC-GS-4 and FC-SW-3 standards. It provides significant enhancements to basic zoning.

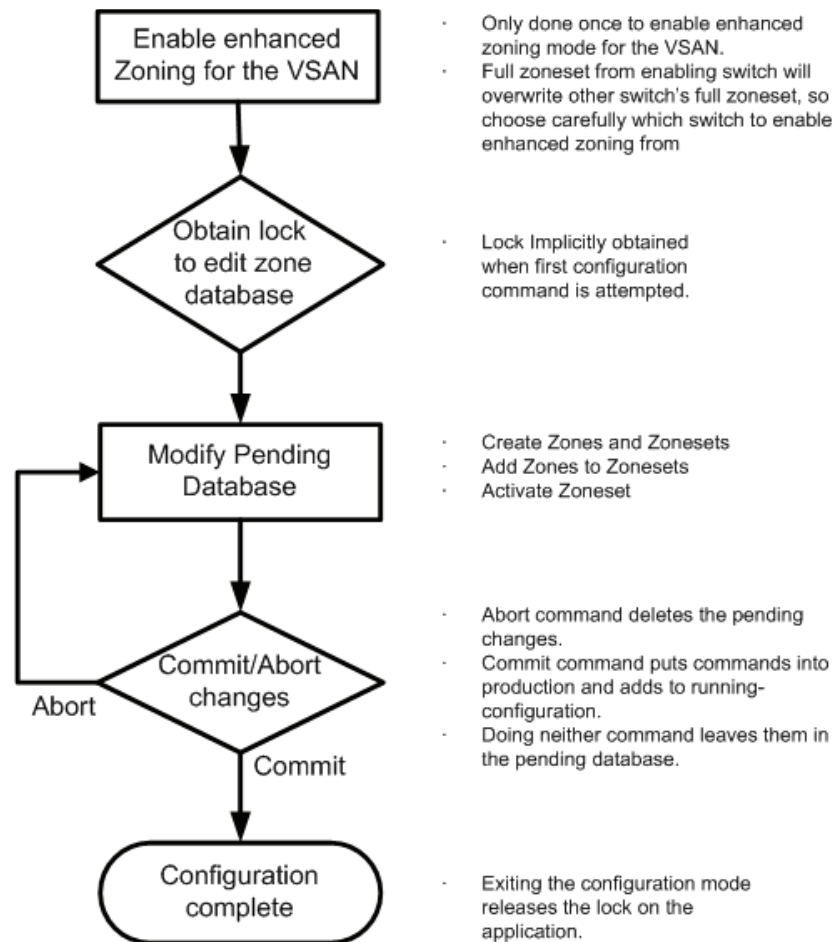
Enhanced zoning has the following features:

- Has a VSAN scope, so that while VSAN X is using enhanced zoning, other VSANs can continue to use basic zoning.
- Is IVR compatible.
- Provides session locking, so that two SAN administrators cannot simultaneously modify a zoning database within a VSAN.
- Provides implicit full zone set distribution, so that the zone set database local to each switch remains in sync when a zone set is modified.
- Allows full zone set changes to be distributed without having to activate a zone set. You can use this to ready features in the daytime and activate the zone set at night.
- Stages modifications until they are explicitly committed or aborted, allowing the SAN administrator to review changes before activation.
- Can control how a zone merge is done. Merging can be accomplished either by performing a union of two zone sets according to the same rules as basic zoning, or by merging only identical active zone sets. The latter method prevents accidental merging.

Enhanced zoning uses the same techniques and tools as basic zoning, with a few added commands that are covered in these recipes. The flow of enhanced zoning, however, differs from that of basic zoning. For one thing, a VSAN-wide lock, if available, is implicitly obtained for enhanced zoning. Second, all zone and zone set modifications for advanced zoning include activation. Last, changes are either committed (put into production) or aborted (pending changes are scrapped) with advanced zoning. The flow is illustrated in [Figure 7-2](#).



Figure 7-2 Enhanced Zoning Flowchart



## Enabling Enhanced Zoning

Enhanced zoning, with its VSAN scope, requires that all switches in a VSAN be capable of enhanced zoning and have enhanced zoning enabled later on. Due to its distributed architecture and abilities, enhanced zoning is enabled only on one switch in the VSAN. Commands are then propagated to other switches in the VSAN. The rules for enabling enhanced zoning with Fabric Manager are as follows:

- Enhanced zoning is enabled on just one switch in the VSAN. Attempting to enable it on multiple switches in the same VSAN can result in a failure to activate.
- Enabling enhanced zoning does not trigger a zone set activation.
- The switch chosen to perform migration distributes its full zone database to other switches in the VSAN, which overwrites the destination switches' full zone set database.



### Caution

It is critical that you select the correct switch for enhanced zoning; otherwise, you can accidentally delete the wrong full zone set database.

## Enabling Enhanced Zoning with the CLI

To enable enhanced zoning with the CLI, follow these steps:

- 
- Step 1** Enter configuration mode and enable enhanced zoning with the **zone mode enhanced** command.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# zone mode enhanced vsan 3000
WARNING: This command would distribute the zoning database
 of this switch throughout the fabric.
Please enter yes to proceed.(y/n) [n]? y
Set zoning mode command initiated. Check zone status
mds-switch-2(config)# end
mds-switch-2#
```

- Step 2** Display the zoning status with the **show zone status** command.

```
mds-switch-2# sh zone status vsan 3000
VSAN: 3000 default-zone: deny distribute: active only Interop: default
 mode: enhanced merge-control: allow session: none
 hard-zoning: enabled broadcast: enabled
Default zone:
 qos: none broadcast: disabled ronly: disabled
Full Zoning Database :
 Zonesets:1 Zones:1 Aliases: 0 Attribute-groups: 1
Active Zoning Database :
 Name: Zoneset1 Zonesets:1 Zones:1
Status: Set zoning mode complete at 03:57:48 UTC Jun 1 2007
mds-switch-2#
```

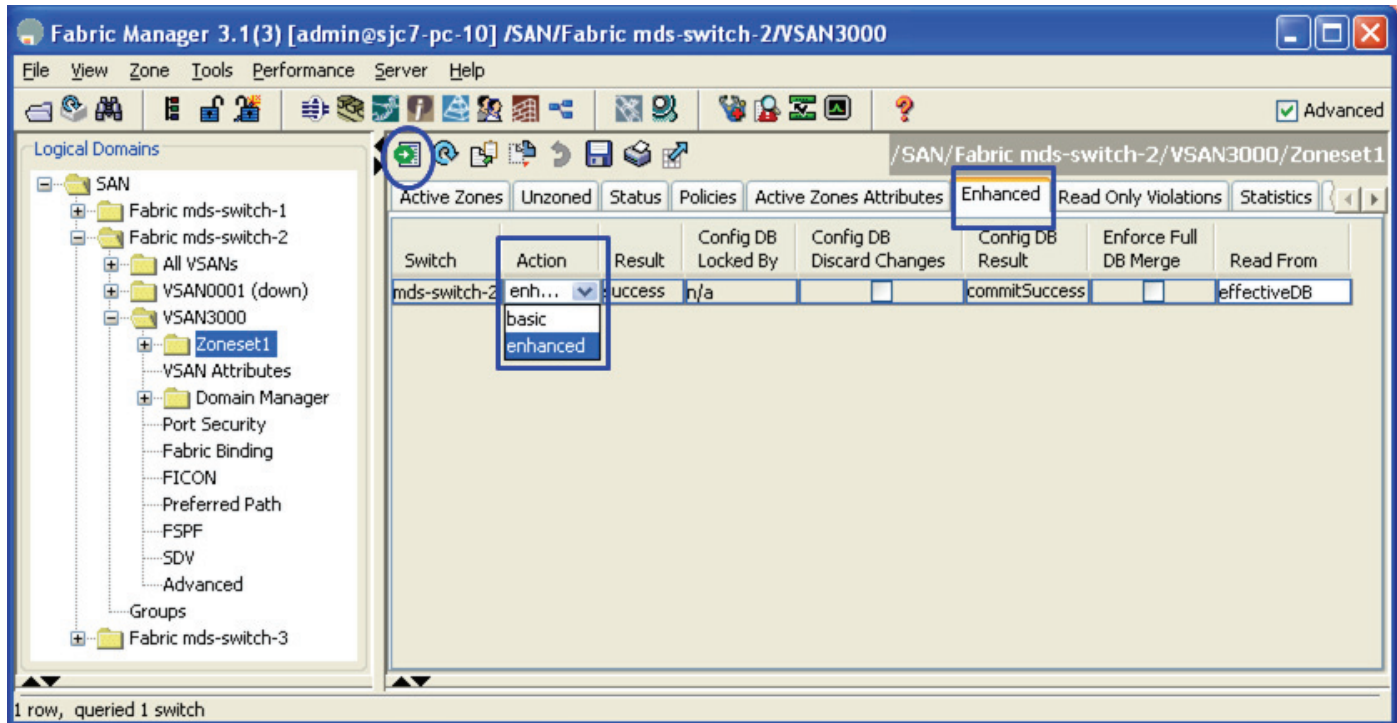
---

## Enabling Enhanced Zoning with Fabric Manager

To enable enhanced zoning with Fabric Manager, follow these steps:

- 
- Step 1** In the Logical Domains pane, choose a VSAN and then choose the folder corresponding to the name of the active zone set (see [Figure 7-3](#)). If no active zone set exists, choose default zone.
- Step 2** Choose the **Enhanced** Tab (see [Figure 7-3](#)).
- Step 3** In the Action column for the enabling switch, change the cell to **Enhanced** and save the configuration using the green Apply button (see [Figure 7-3](#)). From now on, this switch distributes its full zone database for this VSAN, overwriting all other switches in the enhanced zone database.

Figure 7-3 Enabling Enhanced Zoning with Fabric Manager



**Step 4** Click **Apply Changes**.

## Displaying a User with the Current Lock in CLI and Fabric Manager

With enhanced zoning, only one user at a time can make changes to the zone database within a VSAN. The database is implicitly locked.

To determine who has the database locked using the CLI, use the **show zone status** command.

```
mds-switch-2# show zone status vsan 3000
VSAN: 3000 default-zone: deny distribute: active only Interop: default
mode: enhanced merge-control: allow session: cli [admin]
hard-zoning: enabled broadcast: enabled
Default zone:
 qos: none broadcast: disabled ronly: disabled
Full Zoning Database :
 Zonesets:1 Zones:1 Aliases: 0 Attribute-groups: 1
Active Zoning Database :
 Name: Zoneset1 Zonesets:1 Zones:1
Status: Operation failed: [Error: Zoneset already present]:
 at 04:37:15 UTC Jun 1 2007
```

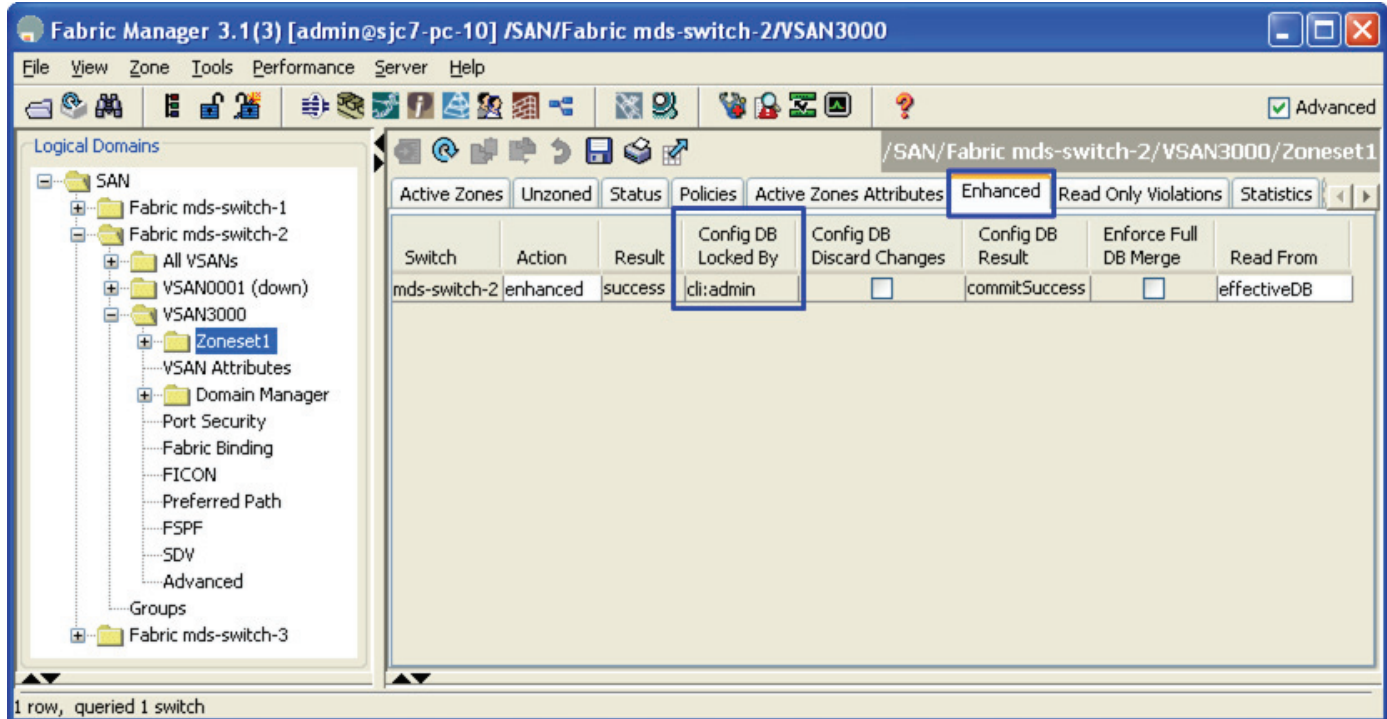
To determine who has this lock in Fabric Manager, follow these steps:

- Step 1** Choose the Logical Domains Pane (see [Figure 7-4](#)).
- Step 2** Choose the VSAN to be investigated (see [Figure 7-4](#)).
- Step 3** Choose the name of the zone set, or Default Zone if there is no active zone set. (See [Figure 7-4](#).)

**Step 4** Click the **Enhanced** tab (see Figure 7-4).

The user is displayed in the **Config DB Locked By** column as shown in Figure 7-4.

Figure 7-4 Displaying the Enhanced Zoning Lock



## Zone Sets

Zone sets are containers for zones. There are two types of zone sets on MDS switches: active zone sets and local zone sets.

- **Active zone set**—It provides the rules by which the MDS platform enforces its zoning security policy. It cannot be modified and it is distributed to all switches in the VSAN. There are specific rules for merging the active zone set when two switches are connected by an ISL as set by the Fibre Channel standards. There is only one active zone set per fabric and per VSAN.
- **Local zone set(s)**—They are contained in the full zone set database on the switch. The zone sets are edited directly and then activated to become the active zone set. They can optionally be distributed to other switches, either manually or when a zone set is activated. There can be multiple local zone sets in a fabric and in a VSAN.

## Distributing Zone Sets

The zone set in the full zone set database can be distributed to other switches either during activation or manually when basic zoning is enabled. When enhanced zoning is enabled, the full zone set is always distributed when changes are committed to the full zone set database.

**Tip**

This feature should be enabled on all switches in the fabric, and can be specified in the initial setup script. This is also the default function when enhanced zoning is enabled.

## Distributing Zone Sets Automatically

Enabling automatic full zone set distribution distributes the local zone set to all other switches in the VSAN when a zone set is activated.

**Note**

When two VSANs with full zone set distribution enabled are merged, they try to merge the full zone set database according to standard zone set merge rules. Failure to merge the full zone set database does not isolate the ISL; only failure to merge the active zone set results in an isolated ISL. This failure to merge the full zone set database produces the following syslog error message:

```
2007 May 31 14:35:59 mds-switch-2 %ZONE-2-ZS_MERGE_FULL_DATABASE_MISMATCH: %$VSAN 1000%$
Zone merge full database mismatch on interface fcl/1
```

## Distributing Zone sets Automatically with the CLI

To automatically distribute zone sets from the CLI, use the **zoneset distribute** command:

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# zoneset distribute full vsan 3000
```

## Distributing Zone Sets Automatically with Fabric Manager

To automatically distribute zone sets using Fabric Manager, follow these steps:

- Step 1** In the Logical Domains window, choose the fabric.
- Step 2** Choose the VSAN to be modified.
- Step 3** Select the name of the Active Zone set (or default zone if none is active).
- Step 4** Click the **Policies** tab.
- Step 5** Change the Propagation field to **FullZoneSet** for all switches in that VSAN.
- Step 6** Click the green **Apply Changes** icon.

## Distributing Zone Sets Manually

You can distribute the full zone set database to other switches without activating a zone set. Do this when a new switch is brought into the fabric and the zone set with its zones and Fibre Channel aliases need to be distributed. This **zoneset distribute** command overwrites the existing zone set database in the target switch.

```
mds-switch-2# zoneset distribute vsan 3000
Zoneset distribution initiated. check zone status
```

# Zones

In order for two devices to communicate, they must be in the same zone. Valid members of a zone are:

- Port WWN
- FC alias
- FC ID
- FWWN (WWN of a FC interface)
- Switch interface (fc X/Y)
- Symbolic node name
- Device alias

The four most common zone member types are the port world-wide name (pWWN), device alias, FC alias, and the switch interface.

**Tip**

We recommend that you use device aliases for zoning because they provide hardware-enforced zoning and associate a zone member to a specific HBA rather than to the switch port. Also, device aliases have the added benefit of being VSAN-independent and are based on an easy-to-understand name rather than a cryptic pWWN.

Equally important is the name of the zone. Environments use many different zone names. However, all name formats should provide relevant information as to their contents. Names like “Zone1” or “TapeZone” do not provide sufficient information about their contents.

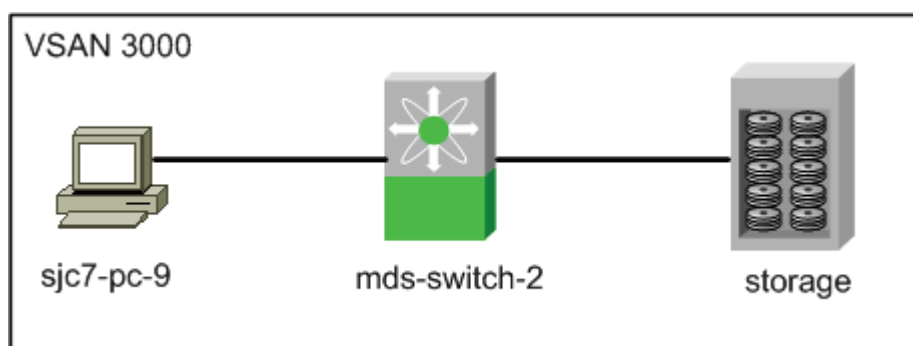
**Tip**

A zone should contain two members and the zone name should contain identifiers related to the two devices (for example, Z\_testhost\_fcaw0\_symm13FA3aa). That name may be longer than Z\_testhost\_hba0, but it provides detailed information about the contents, and you will not have to consult further sources of documentation.

## Creating a Zone and Adding It to a Zone Set with Fabric Manager

This recipe creates a zone set, creates zones, adds them to the zone set, and then activates the zone set. The method used is the same for both basic zoning and enhanced zoning. The following topology is used:

Figure 7-5 Fabric Manager Zoning Topology



In addition, these resources are used in this example:

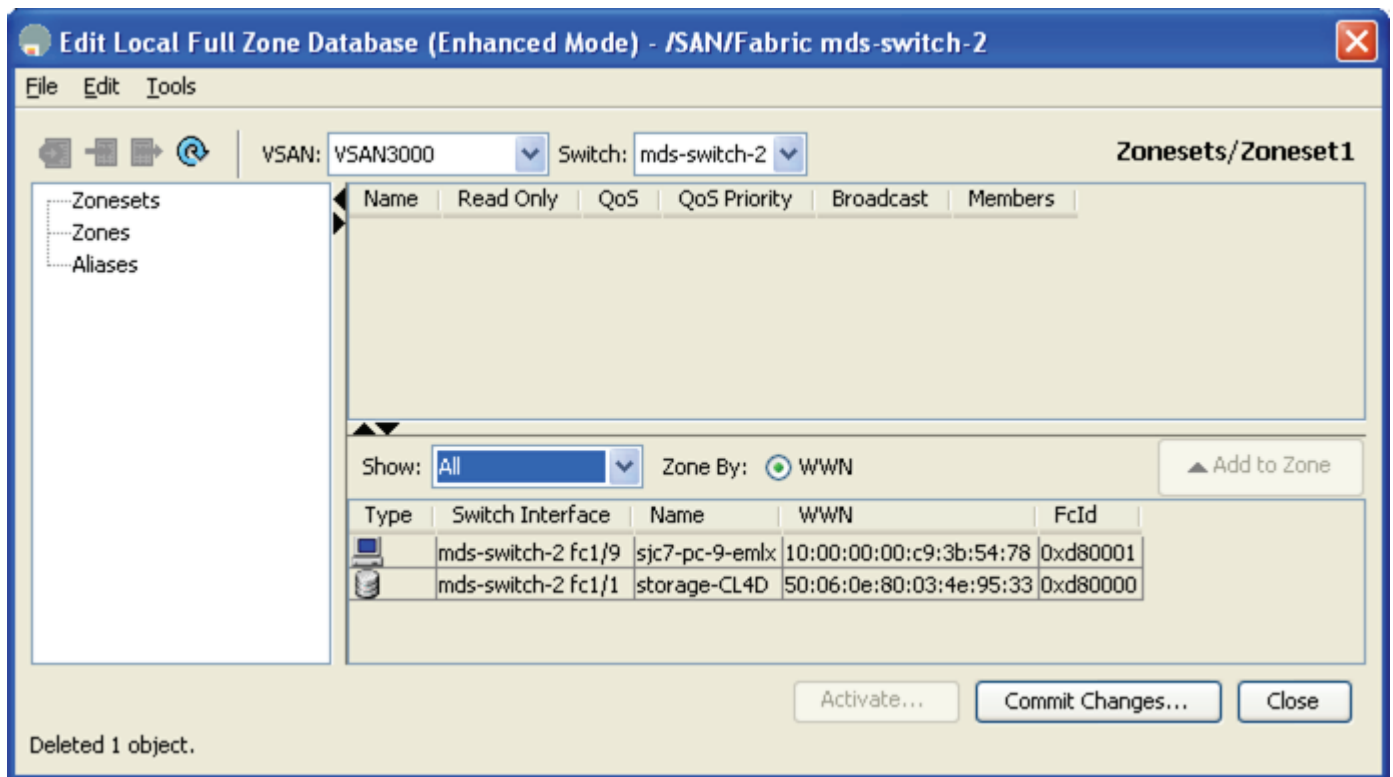
- Zone set: ZS\_cookbook
- W2k3 host: sjc7-pc-9 lpfc
- Storage array: storage

## Creating a Zone Set

To create a zone set, follow these steps:

- Step 1** In the Logical Domains pane, right-click the VSAN, and select **Edit Local Full Zone Database**. You see the screen shown in [Figure 7-6](#).

Figure 7-6 Edit Local Full Zone Database



### Note

- The VSAN field displays the VSAN whose database is to be modified.
- The Switch field displays the switch being edited.
- The Name column lists either FC aliases or Global Device Aliases ([Device Aliases](#), page 1-41) if they are used.

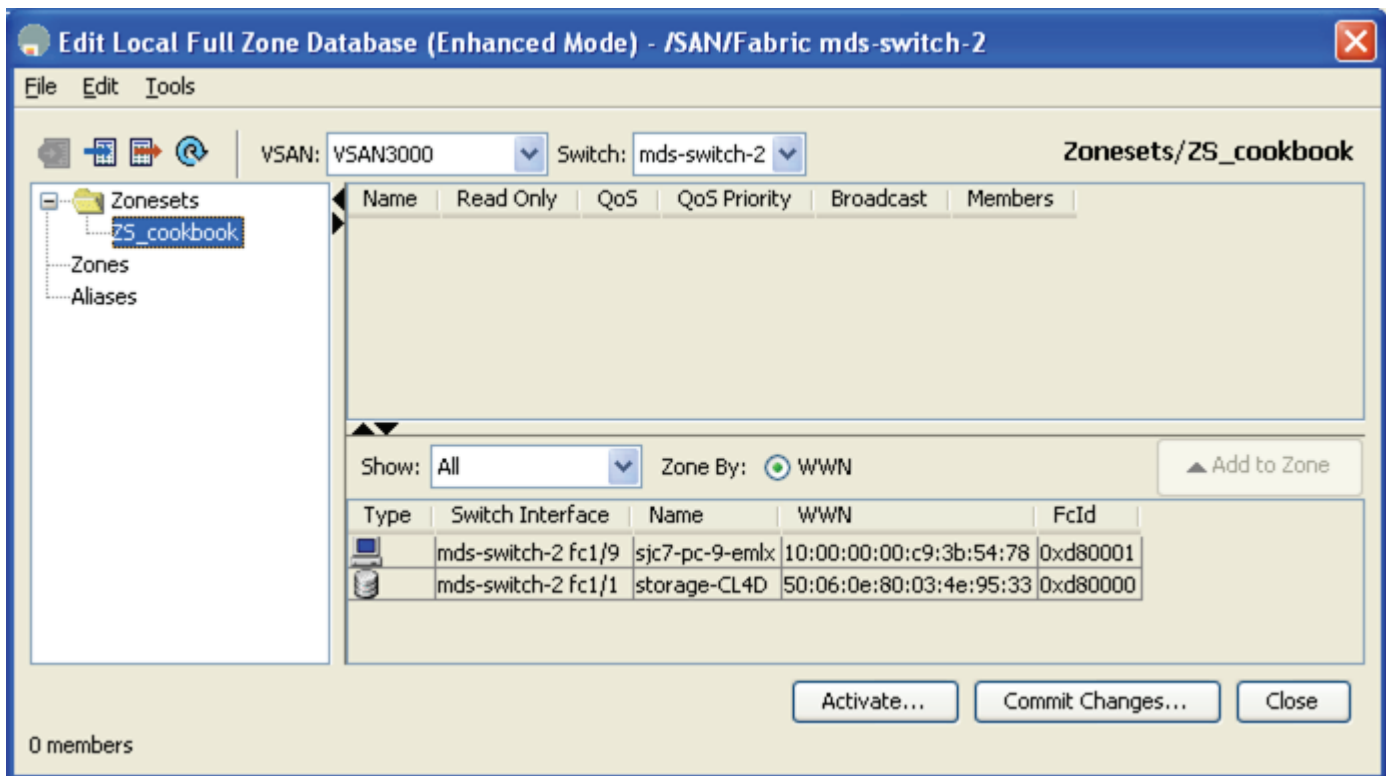
- If Full Zone Set Distribution is enabled, the left column lists existing zone sets and zones. If Active Zone Set Distribution is enabled, choose the switch that contains the Full Zone Database.

## Creating a Zone Set

To create a zone set from Fabric Manager, follow these steps:

- Step 1** In the left pane, right-click **Zonesets**.
- Step 2** In the resulting pop-up menu, select **Insert...**
- Step 3** Enter a zone set name, such as ZS\_cookbook, as shown in [Figure 7-7](#), and then click **OK**.

Figure 7-7 Create Zone Set



At this point a zone set has been created. The next phase is to create a zone and add members to it.

## Creating a Zone and Adding Members

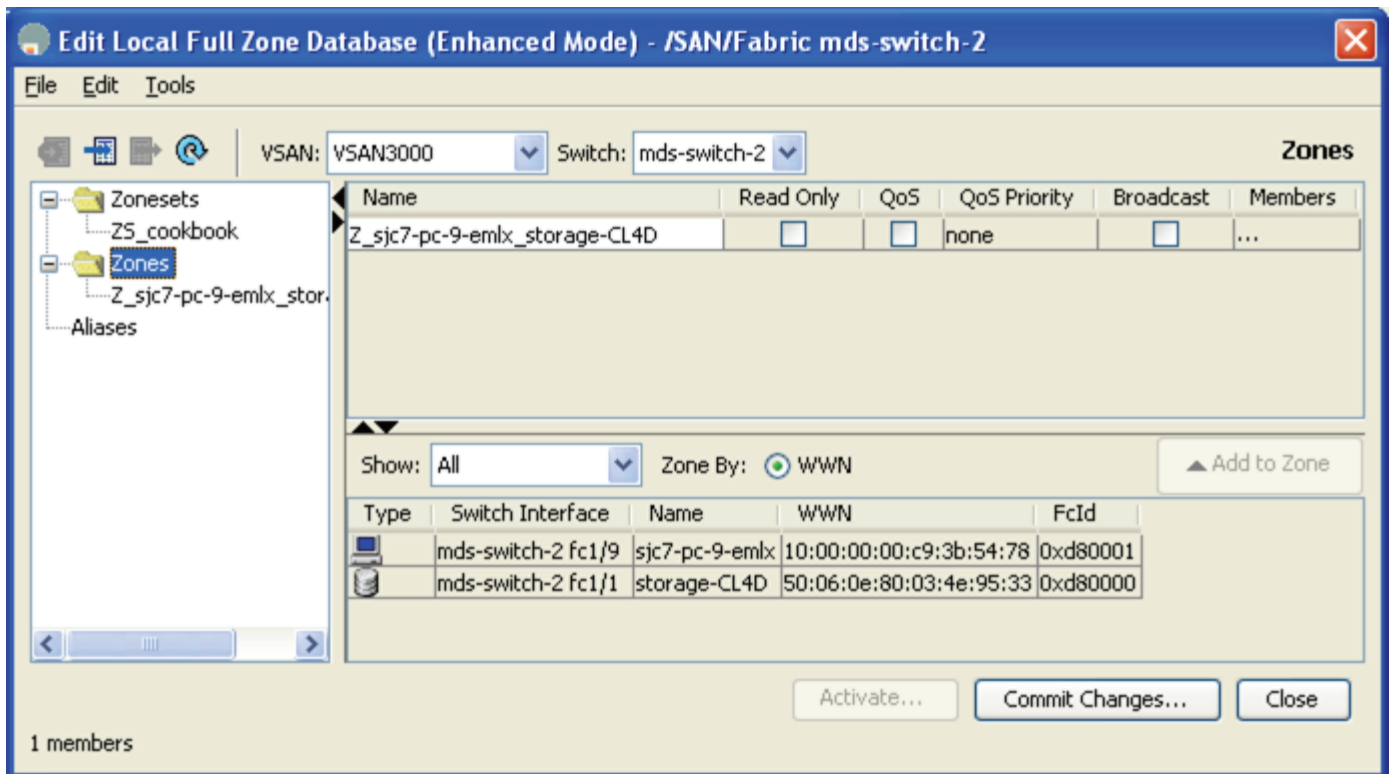
To create a zone and add members to it from Fabric Manager, follow these steps:

- Step 1** Right click **Zones**.



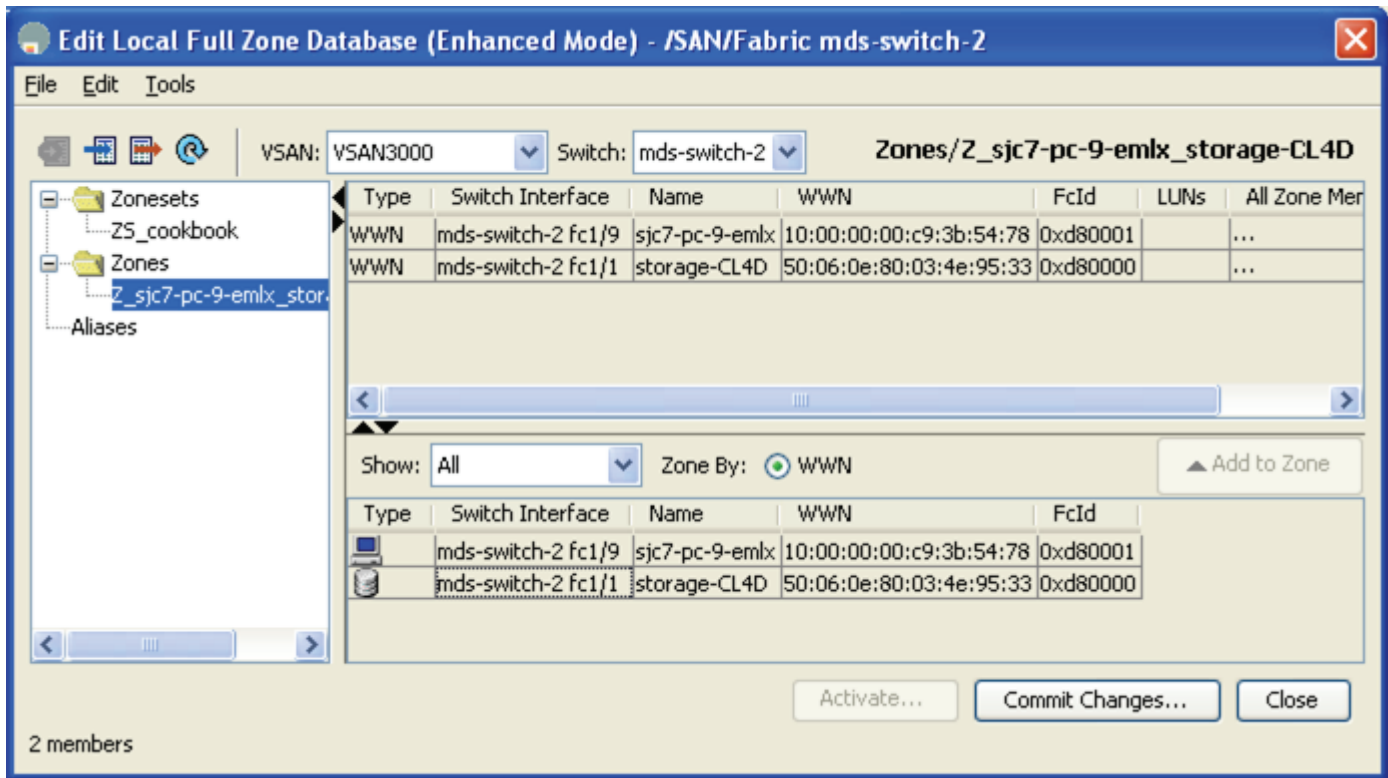
- Step 2** In the resulting pop-up menu, select **Insert....**
- Step 3** Enter a meaningful zone name such as **Z\_sjc7-pc-9-emlx\_storage\_CL4D** to represent both the initiator and target in the name.
- Step 4** Click **OK**.
- You see the dialog box in [Figure 7-8](#).

Figure 7-8 Zone Database after Creating Zone Set and Zones



- Step 5** Drag the two end devices from the bottom pane into the new zone. This creates a pWWN-based zone. If non-pWWN zone members (such as interface, FCID, or Global Device Alias) are needed, refer to [Creating Non-pWWN-Based Zones](#), page 7-14 to specify these member types before continuing.

Figure 7-9 Zone with Newly Added Members



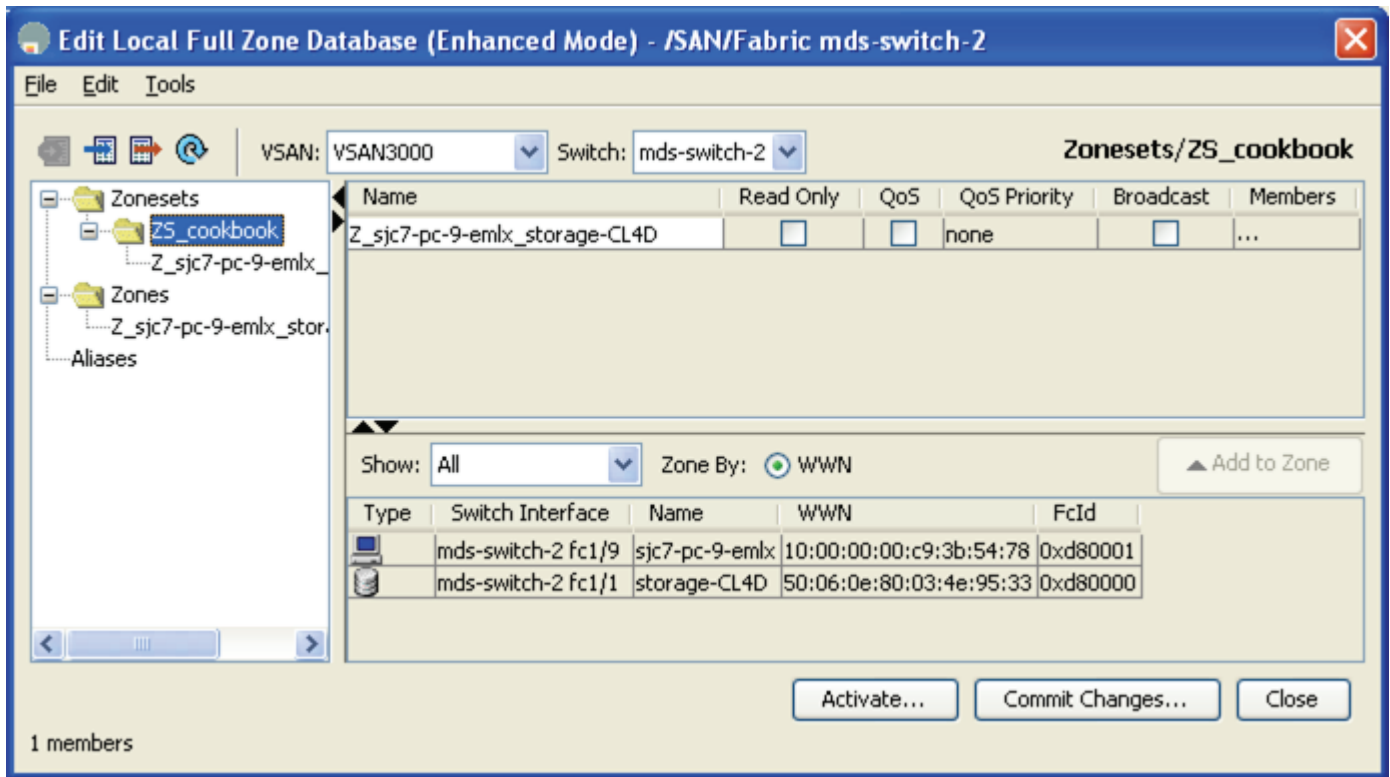
This zone is created, as shown in [Figure 7-9](#). Next, add the zone to the zone set.

### Adding the Zone to the Zone Set and Activating It

To add the zone to the zone set, follow these steps:

- Step 1** In the left pane, drag the zone (Z\_sjc7-pc-9-emplx\_storage-CL4D) into the zone set (ZS\_cookbook). The zone set's icon changes by appending a folder icon, and it expands with the newly added zone underneath it. ([Figure 7-10](#))

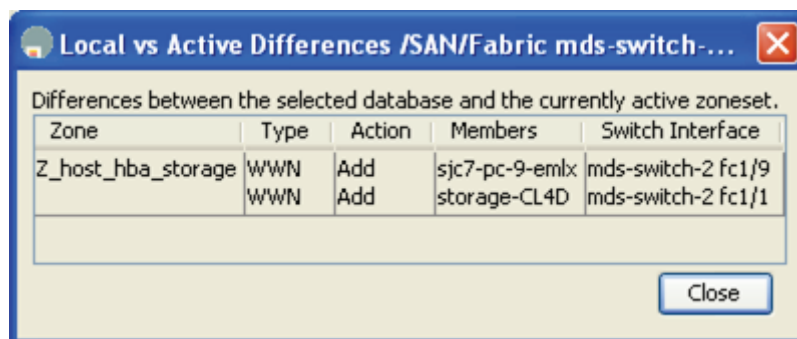
Figure 7-10 Zone Added to Zone Set



Now you need to activate the new zone set. This instructs the switch program to update its Access Control Lists, and modify the running configuration of the zone server to allow the two devices to communicate. When enhanced zoning is enabled, clicking **Activate** commits the changes as well.

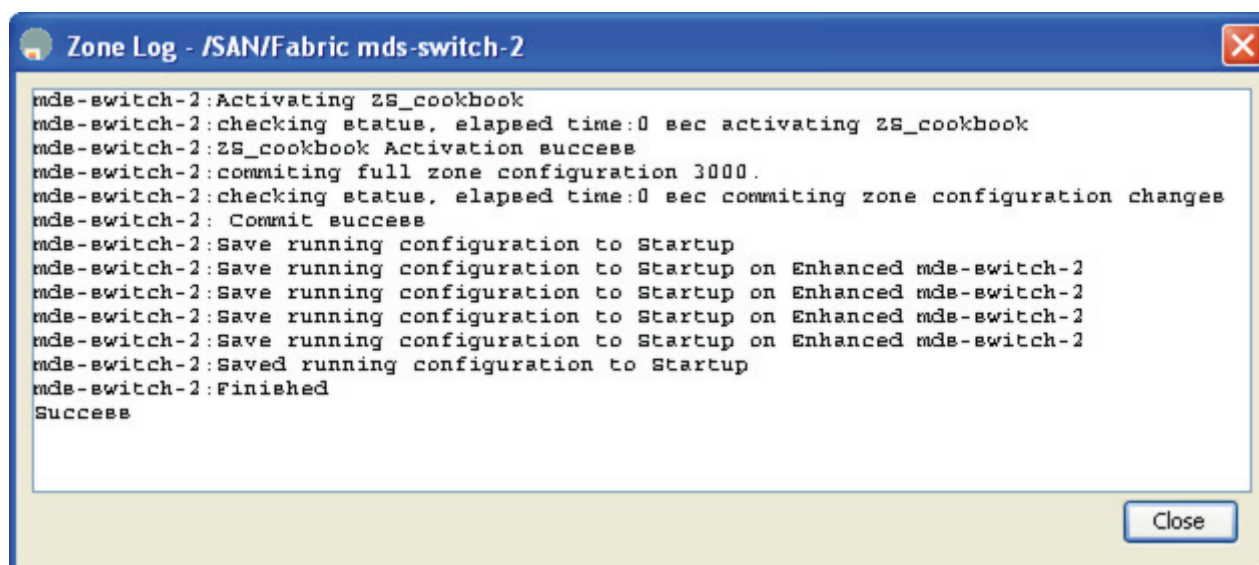
- Step 2** In the left pane, right-click the zone set (ZS\_cookbook) and choose **Activate...**
- If the current Active zone set is empty, click **Continue Activation...**
  - If the current Active zone set is not empty, Fabric Manager prompts the user with the option to view the differences between the active zone set and zone set that is being activated. If you Select **YES**, the Fabric Manager displays what is being added or removed from the active zone set. You see the dialog box in [Figure 7-11](#).
- Step 3** Click **close** and then select **Continue Activation...**

Figure 7-11 Zone Set Proposed Changes



The Zone Set activation results dialog is as shown in Figure 7-12.

Figure 7-12 Zoneset Activation Results



The zone set is now active and the two end devices can communicate.

## Creating Non-pWWN-Based Zones

This recipe creates a zone that is not based on pWWN. The procedure is the same for either basic or enhanced zoning.

To create the zones, follow these steps:

- Step 1** In the Logical Domains pane, right-click the VSAN, and select **Edit Local Full Zone Database**.
- Step 2** In the resulting dialog box, right-click **Zones** and select **Insert...**
- Step 3** Specify a zone name and click **OK**.
- Step 4** Right-click the newly created zone and select **Insert...** You see the options shown in Figure 7-13.

Figure 7-13 Possible Zone Member Types

- Step 5** Select the type of zone member require. This selection changes the rest of the screen. For example, if **Switch & Port** is selected, the text boxes change to **Switch Interface** (for example, fc1/1) and **Switch Address** (for example, 192.168.1.2). Also, the meaning of... and the pull-down menus change depending on the zone member type.

**Note**

- Domain and Port zoning should only be done when working in interop mode 2 or 3. See the section [Setting VSAN Interop Mode, page 6-10](#) for more information about interop modes.
- Alias refers to both FC Alias and Global Device Alias, depending on which mode Fabric Manager is in.

The resulting zone still must be added to a zone set and the zone set activated, which is described in [Creating a Zone and Adding It to a Zone Set with Fabric Manager, page 7-8](#).

## Creating a Zone and Adding It to a Zone Set with the CLI Standalone Method

This procedure creates a single zone for a Solaris host with a disk storage port, then adds the zone to the zone set ZS\_cli\_cookbook. This is done with the standalone method, which does not automatically add the zone to the zone set upon creation of the zone.

You can also use this procedure to add an existing zone to a zone set. The procedure is the same for both basic zoning and enhanced zoning with one exception. With enhanced zoning, the pending database must be committed at the end.

## Creating a Device Alias-Based Zone with the CLI

This example uses device alias as zone members. This recipe assumes that device alias is operating in enhanced mode.

Obtain pWWNs either from the device itself or from the **show flogi database vsan 3000** command.



### Note

When using device alias in enhanced mode and doing zoning using device alias, if the pWWN of a device changes (HBA replacement) and if the device alias of the device is not suitably changed in the device alias database, the zone information will not automatically change to reflect the new pWWN. The device alias has to be updated with the new pWWN for the changes to propagate to the zones.

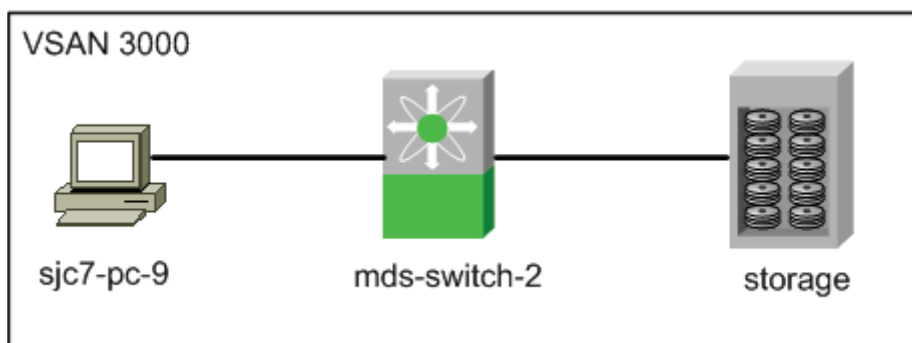
```
ds-switch-2# sh flogi database vsan 3000
```

| INTERFACE | VSAN | FCID     | PORT NAME               | NODE NAME               |
|-----------|------|----------|-------------------------|-------------------------|
| fc1/1     | 3000 | 0xd80000 | 50:06:0e:80:03:4e:95:33 | 50:06:0e:80:03:4e:95:33 |
| fc1/9     | 3000 | 0xd80001 | 10:00:00:00:c9:3b:54:78 | 20:00:00:00:c9:3b:54:78 |

Total number of flogi = 2  
mds-switch-2#

The topology in [Figure 7-14](#) is used in the example.

Figure 7-14 Standalone Zoning Topology



These resources are also used in this example:

- Zone set: ZS\_cli\_cookbook
- W2k3, HBA instance's device alias: sjc7-pc-9-emplx
- Storage port's device alias: storage\_CL4D

To create a single zone for a W2k3 host with a disk storage port, follow these steps:

**Step 1** Create the device alias for the host HBA and the storage port using the device alias database commands.

```
mds-switch-2# config t
mds-switch-2(config)# device-alias database
mds-switch-2(config-device-alias-db)# device-alias name storage-CL4D pwwn
50:06:0e:80:03:4e:95:33
ds-switch-2(config-device-alias-db)# device-alias name sjc7-pc-9-emplx pwwn
10:00:00:00:c9:3b:54:78
mds-switch-2(config-device-alias-db)# exit
mds-switch-2(config)# device-alias commit
```

**Note**

A device alias commit is required to store the device alias into the device alias database.

- Step 2** Create the zone with the **zone name** command. Use a zone name that reflects the names of the members. Then add members to the zone with the **member device-alias** command.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# zone name Z_sjc7-pc-9-emplx_storage-CL4D vsan 3000
mds-switch-2(config-zone)# member device-alias storage-CL4D
mds-switch-2(config-zone)# member device-alias sjc7-p-9-emplx
mds-switch-2(config-zone)# exit
mds-switch-2(config)#
```

- Step 3** Add the zone to the zone set with the **zoneset name** command.

```
mds-switch-2(config)# zoneset name ZS_cli_cookbook vsan 3000
mds-switch-2(config-zoneset)# member Z_sjc7-pc-9-emplx_storage-CL4D
mds-switch-2(config-zoneset)# exit
mds-switch-2(config)# exit
```

- Step 4** Display the zone set with the **show zoneset name** command.

```
mds-switch-2# show zoneset name ZS_cli_cookbook vsan 3000
zoneset name ZS_cli_cookbook vsan 3000
 zone name Z_sjc7-pc-9-emplx_storage-CL4D vsan 3000
 device-alias storage-CL4D
 device-alias sjc7-pc-9-emplx
```

- Step 5** Put the zone set into production with the command **zoneset activate name ZS\_cli\_cookbook vsan 3000**. This activates all the zones in the zone set, not just the new one.

```
mds-switch-2(config)# zoneset activate name ZS_cli_cookbook vsan 3000
```

- Step 6** If you are using enhanced zoning, the zone has to be committed to the database using the command **zone commit vsan 3000**.

```
mds-switch-2(config)# zone commit vsan 3000
Commit operation initiated. Check zone status
```

**Caution**

Only when the commit is complete does the zone set become active.

- Step 7** Display the zone set with the **show zoneset** command.

```
mds-switch-2# sh zoneset active v 3000
zoneset name ZS_cli_cookbook vsan 3000
 zone name Z_sjc7-pc-9-emplx_storage-CL4D vsan 3000
 * fcid 0xd80000 [device-alias storage-CL4D]
 * fcid 0xd80001 [device-alias sjc7-pc-9-emplx]
mds-switch-2#
```

## Creating a pWWN-based Zone with the CLI

This example uses pWWNs as zone members. You can obtain pWWNs either from the device itself or from the **show flogi database vsan 3000** command.

```
mds-switch-2# sh flogi database vsan 3000

```

| INTERFACE | VSAN | FCID     | PORT NAME                                    | NODE NAME               |
|-----------|------|----------|----------------------------------------------|-------------------------|
| fc1/1     | 3000 | 0xd80000 | 50:06:0e:80:03:4e:95:33<br>[storage-CL4D]    | 50:06:0e:80:03:4e:95:33 |
| fc1/9     | 3000 | 0xd80001 | 10:00:00:00:c9:3b:54:78<br>[sjc7-pc-9-emplx] | 20:00:00:00:c9:3b:54:78 |

Total number of flogi = 2.  
mds-switch-2#

The topology in [Figure 7-14](#) is used in the example.

These resources are also used in this example:

- Zone set: ZS\_cli\_cookbook
- W2k3, hba instance lpfc: 10:00:00:00:c9:3b:54:78
- Storage port: 50:06:0e:80:03:4e:95:33

To create a single zone for a W2k3 host with a disk storage port, follow these steps:

- Step 1** Create the zone with the **zone name** command. Use a zone name that reflects the names of the members. Then add members to the zone with the **member pwwn** command.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# zone name Z_sjc7-pc-9-emplx_storage-CL4D vsan 3000
mds-switch-2(config-zone)# member pwwn 50:06:0e:80:03:4e:95:33
mds-switch-2(config-zone)# member pwwn 10:00:00:00:c9:3b:54:78
mds-switch-2(config-zone)# exit
mds-switch-2(config)#
```

- Step 2** Add the zone to the zone set with the **zoneset name** command.

```
mds-switch-2(config)# zoneset name ZS_cli_cookbook vsan 3000
mds-switch-2(config-zoneset)# member Z_sjc7-pc-9-emplx_storage-CL4D
mds-switch-2(config-zoneset)# exit
mds-switch-2(config)#
```

- Step 3** Display the zone set with the **show zoneset name** command.

```
mds-switch-2# show zoneset name ZS_cli_cookbook vsan 3000
zoneset name ZS_cli_cookbook vsan 3000
 zone name Z_sjc7-pc-9-emplx_storage-CL4D vsan 3000
 pwwn 50:06:0e:80:03:4e:95:33 [storage-CL4D]
 pwwn 10:00:00:00:c9:3b:54:78 [sjc7-pc-9-emplx]
```

- Step 4** Put the zone set into production with the **zoneset activate name ZS\_cli\_cookbook vsan 3000** command. This activates all the zones in the zone set, not just the new one. .

```
mds-switch-2(config)# zoneset activate name ZS_cli_cookbook vsan 3000
```

- Step 5** If you are using enhanced zoning, the zone has to be committed to the database using the **zone commit vsan 3000** command.

```
mds-switch-2(config)# zone commit vsan 3000
Commit operation initiated. Check zone status
```



#### Caution

Only when the commit is complete does the zone set become active.

- Step 6** Display the zone set with the **show zoneset** command.

```
mds-switch-2# sh zoneset active v 3000
```



```

zoneset name ZS_cli_cookbook vsan 3000
 zone name Z_sjc7-pc-9-emplx_storage-CL4D vsan 3000
 * fcid 0xd80000 [pwwn 50:06:0e:80:03:4e:95:33] [storage-CL4D]
 * fcid 0xd80001 [pwwn 10:00:00:00:c9:3b:54:78] [sjc7-pc-9-emplx]
mds-switch-2#

```

## Creating a Zone and Adding it to a Zone Set with the CLI Inline Method

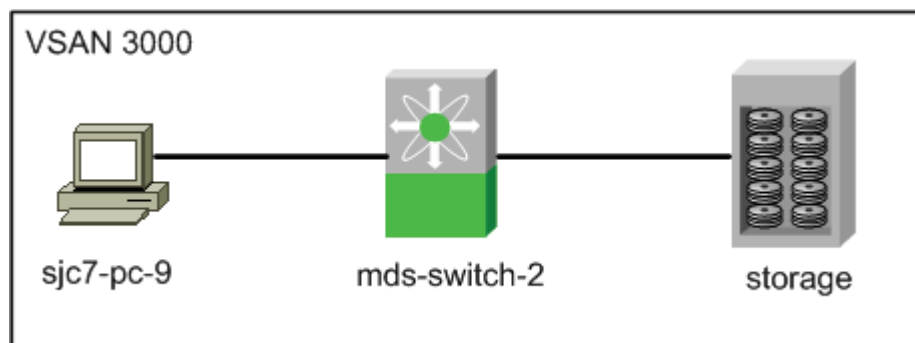
This procedure creates a single zone for a Solaris host with a disk storage port in it, then adds the zone to the zone set **ZS\_Engr\_primary**. Using the inline method automatically adds the zone to the zone set upon creation of the zone.

The procedure is the same for both basic zoning and enhanced zoning, with one exception. With enhanced zoning, the pending database must be committed at the end.

This example uses pWWNs as zone members. You can obtain pWWNs either from the device itself or from the **show flogi database vsan 3000** command.

The topology in [Figure 7-15](#) is used in the example.

Figure 7-15 Inline Zoning Topology



```
mds-switch-2# sh flogi database vsan 3000
```

| INTERFACE | VSAN | FCID     | PORT NAME                                    | NODE NAME               |
|-----------|------|----------|----------------------------------------------|-------------------------|
| fc1/1     | 3000 | 0xd80000 | 50:06:0e:80:03:4e:95:33<br>[storage-CL4D]    | 50:06:0e:80:03:4e:95:33 |
| fc1/9     | 3000 | 0xd80001 | 10:00:00:00:c9:3b:54:78<br>[sjc7-pc-9-emplx] | 20:00:00:00:c9:3b:54:78 |

Total number of flogi = 2.

```
mds-switch-2#
```

The following resources are also used in this example:

- Zone set: **ZS\_cli\_cookbook**
- W2k3, hba instance lpfc: 10:00:00:00:c9:3b:54:78
- Storage port: 50:06:0e:80:03:4e:95:33

To create a single zone for a W2k3 host with a disk storage port, follow these steps:

**Step 1** Enter the submode of the zone set with the **zoneset name** command.

```
mds-switch-2# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# zoneset name ZS_cli_cookbook vsan 3000
```

**Step 2** Create the zone with the **zone name** command.

```
mds-switch-2(config-zoneset)# Z_sjc7-pc-9-emplx_storage-CL4D vsan 3000
```

**Step 3** Add members to the zone with the **member** command.

```
mds-switch-2(config-zoneset-zone)# member pwwn 50:06:0e:80:03:4e:95:33
mds-switch-2(config-zoneset-zone)# member pwwn 10:00:00:00:c9:3b:54:78
```

**Step 4** For basic zoning, put the zone set into production with the **zoneset activate** command. This activates all zones in the zone set, not just the new one. For enhanced zoning, in addition to activating the zone set, you must commit it in Step 5.

```
mds-switch-2(config)# zoneset activate name ZS_cli_cookbook vsan 3000
```

**Step 5** If enhanced zoning is used, explicitly commit the zone set with **zoneset commit** command.

```
mds-switch-2(config)# zone commit vsan 3000
```

**Step 6** Display the zone set with the **show zoneset** command.

```
mds-switch-2# show zoneset name ZS_cli_cookbook vsan 3000
zoneset name ZS_cli_cookbook vsan 3000
 zone name Z_sjc7-pc-9-emplx_storage-CL4D vsan 3000
 pwwn 50:06:0e:80:03:4e:95:33 [storage-CL4D]
 pwwn 10:00:00:00:c9:3b:54:78 [sjc7-pc-9-emplx]
```

**Step 7** Display the active zone set with the **show active zoneset** command.

```
mds-switch-2# show zoneset active vsan 3000
zoneset name ZS_cli_cookbook vsan 3000
 zone name Z_sjc7-pc-9-emplx_storage-CL4D vsan 3000
 * fcid 0xd80000 [pwwn 50:06:0e:80:03:4e:95:33] [storage-CL4D]
 * fcid 0xd80001 [pwwn 10:00:00:00:c9:3b:54:78] [sjc7-pc-9-emplx]
```

## Creating a FC Alias-Based Zone with the CLI

Fibre Channel aliases let the administrator assign a plain text, human readable name to a pWWN, FC ID interface, IP address, nWWN, or symbolic nodename. FC aliases are restricted to the VSAN where they were created. The most common and recommended method of naming is using the pWWN, which is demonstrated in this procedure.



### Note

We recommend using device alias instead of FC alias as Cisco best practice.

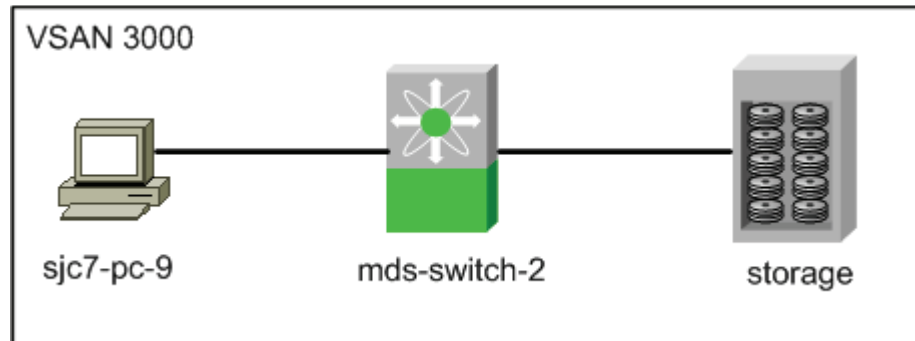


### Tip

- Aliases are distributed with the full zone set database, so if multiple switches are changed, enable full zone set distribution to distribute the aliases.
- An alias can be mapped to more than one device, however, we recommend one-to-one mapping.

The topology in [Figure 7-16](#) is used in the example.

Figure 7-16 Alias Zoning Topology



The following resources are also used in this example:

- Zone set: ZS\_cli\_cookbook
- W2k3, HBA instance lpfc: 10:00:00:00:c9:3b:54:78
- Storage port: 50:06:0e:80:03:4e:95:33

To create an FC alias-based zone, follow these steps:

**Step 1** Create an FC alias-to-pWWN mapping for each FC alias, using the **member pwwn** command.

```
mds-switch-2# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# fcalias name sjc7-pc-9-emplx vsan 3000
mds-switch-2(config-fcalias)# member pwwn 10:00:00:00:c9:3b:54:78
mds-switch-2(config-fcalias)# exit
mds-switch-2(config)# fcalias name storage_CL4D vsan 3000
mds-switch-2(config-fcalias)# member pwwn 50:06:0e:80:03:4e:95:33
mds-switch-2(config-fcalias)# end
```

**Step 2** Display the newly created FC aliases with the **show fcalias** command.

```
ca-9506# show fcalias vsan 3000
fcalias name sjc7-pc-9-emplx vsan 3000
pwwn 10:00:00:00:c9:3b:54:78

fcalias name storage_CL4D vsan 3000
pwwn 50:06:0e:80:03:4e:95:33
```

**Step 3** Create an alias-based zone in the zone set with the **zone name** command. Add members to the zone using the **member fcalias** command and the names of the FC aliases.

```
mds-switch-2# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# zoneset name Zs_cli_cookbook vsan 3000
mds-switch-2(config-zoneset)# zone name Z_sjc7-pc-9-emplx_storage-CL4D vsan 3000
mds-switch-2(config-zone)# member fcalias sjc7-pc-9-emplx
mds-switch-2(config-zone)# member fcalias storage_CL4D
mds-switch-2(config-zone)# exit
```

**Step 4** Optionally, display the zone set with the **show zoneset** command.

```
mds-switch-2# show zoneset vsan 3000
zoneset name ZS_cli_cookbook vsan 3000
zone name Z_sjc7-pc-9-emplx_storage-CL4D vsan 3000
fcalias name sjc7-pc-9-emplx vsan 3000
pwwn 10:00:00:00:c9:3b:54:78
```

```
fcalias name storage_CL4D vsan 3000
pwwn 50:06:0e:80:03:4e:95:33
```

**Step 5** Activate the zone set with the **zoneset activate** command.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# zoneset activate name ZS_cli_cookbook vsan 3000
Zoneset activation initiated. check zone status
```

**Step 6** If enhanced zoning is enabled, commit the configuration with the **zone commit** command.

```
mds-switch-2(config)# zone commit vsan 3000
```

## Creating an Interface-Based Zone with the CLI

This procedure creates a zone based upon the physical interface (fc X/Y) of the switch. The procedure is the same for both basic zoning and enhanced zoning with one exception. With enhanced zoning, you must commit the pending database at the end. Run the show flogi database command to see which switch ports the host and storage ports have logged into. The storage port is logged into port fc 1/1 and the host port is logged into port fc 1/9.

```
mds-switch-2# sh flogi database vsan 3000
```

| INTERFACE | VSAN | FCID     | PORT NAME                                   | NODE NAME               |
|-----------|------|----------|---------------------------------------------|-------------------------|
| fc1/1     | 3000 | 0xd80000 | 50:06:0e:80:03:4e:95:33<br>[storage-CL4D]   | 50:06:0e:80:03:4e:95:33 |
| fc1/9     | 3000 | 0xd80001 | 10:00:00:00:c9:3b:54:78<br>[sjc7-pc-9-emlx] | 20:00:00:00:c9:3b:54:78 |

Total number of flogi = 2.

```
mds-switch-2#
```

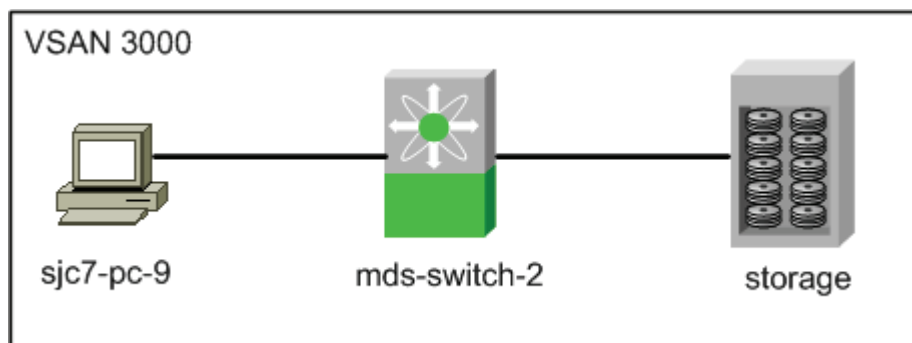


### Tip

Use interface-based zoning when a zone must be created before the HBA is connected to the fabric. After the HBA is connected to the fabric, convert the zone member to a pWWN-based member.

The topology in [Figure 7-17](#) is used in the example.

Figure 7-17 Interface Zoning Topology



To create an interface-based zone with the CLI, follow these steps:

**Step 1** Create the zone using the **zone name** command. Add members with the **member interface** command.

```
ca-9506# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
ca-9506(config)# zoneset name ZS_cli_cookbook vsan 3000
ca-9506(config-zoneset)# zone name Z_sjc7-pc-9-emplx_storage-CL4D vsan 3000
ca-9506(config-zoneset-zone)# member interface fc1/1
ca-9506(config-zoneset-zone)# member interface fc1/9
```

**Step 2** Optionally, display the zone set with the **show zoneset** command.

```
ca-9506# show zoneset vsan 3000
zoneset name ZS_cli_cookbook vsan 3000
 zone name Z_sjc7-pc-9-emplx_storage-CL4D vsan 3000
 interface fc1/1 swwn 20:00:00:0c:85:e9:d2:c0
 interface fc1/9 swwn 20:00:00:0c:85:e9:d2:c0
```



**Note**

The sWWN is the switch WWN as displayed by the **show wwn switch** command:

```
mds-switch-2# show wwn switch
Switch WWN is 20:00:00:0c:85:e9:d2:c0
```

Activate the zone set with the command **zoneset**.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# zoneset activate name ZS_cli_cookbook vsan 3000
Zoneset activation initiated. check zone status
```

**Step 3** If enhanced zoning is enabled, then commit the configuration.

```
mds-switch-2(config)# zone commit vsan 3000
```





## CHAPTER 8

# Inter-VSAN Routing

---

Inter-VSAN Routing (IVR), which was first introduced into the MDS platform in Cisco SAN-OS Release 1.3(1), provides the ability for devices in different VSANs to communicate. VSANs can then be created consisting of devices shared with other VSANs. The classic example is a shared tape library with many hosts in different VSANs. Another common implementation is allowing disk subsystems to communicate over WAN distances without having to merge large zone set databases.

The initial release of IVR (IVR-1) had two disadvantages, although these issues were easily resolved with planning. One, the first release required unique domain\_IDs in the source and destination VSANs. Second, the two VSANs could not have the same number. If you needed to merge two large fabrics and then IVR between them, duplicate VSAN IDs or domain\_IDs in the two fabrics caused problems.

IVR-2 continues to use the same basic principles as in IVR-1, such as IVR topologies, IVR zones, and zone sets as well as transit VSANs. However, the disadvantages are resolved. Starting with Cisco SAN-OS Release 2.1, IVR has the ability to perform Network Address Translation (NAT). IVR with Fibre Channel NAT (IVR-2) solved both issues with IVR-1, eliminating the need for unique domain\_IDs and VSAN IDs.

In addition to the FCNAT capabilities, IVR also gained the ability to leverage Cisco Fabric Services (CFS) (introduced in Cisco SAN-OS Release 2.0(1)) and auto topology (introduced in Cisco SAN-OS Release 2.1(1)). Although these two technologies make implementing IVR easier and faster, careful planning should be done before configuration.



### Tip

The preferred method of configuring IVR either with or without NAT is with CFS. This eases topology configuration and reduces the number of configuration steps and potential configuration errors. See [IVR with CFS, page 8-8](#)

---

# IVR Core Components

This section provides background information about IVR topology, IVR zones and zone sets, and how IVR interacts with CFS.

## IVR Topology

An IVR topology is a set of VSANs that can inter-route one or more IVR-enabled switches. The VSANs specified in the topology can either contain end devices or connect two IVR-enabled switches where the common VSAN does not contain any end devices. This second type of VSAN is referred to as a transit VSAN ([Transit VSANs, page 8-3](#)). Each IVR-enabled switch does not have to include all VSANs in the fabric. However, the topology database must be the same on all switches. For example, in [Table 8-1](#), the switch 172.22.36.11 can route between VSANs 1, 3000, 3001 and 3002; while switch 172.22.36.9 can route between VSANs 1, 3001 and 3003

Table 8-1 IVR Topology

| VSAN Route Switch | VSANs to Route |
|-------------------|----------------|
| 172.22.36.11      | 1,3000-3002    |
| 172.22.36.9       | 1,3001, 3003   |

If a new VSAN was created on switch 172.22.36.11, it could not route between the new VSAN and one of the VSANs in the existing topology until the new VSAN is added into the topology and the database distributed to all the IVR enabled switches.

## Auto Topology

Configuring IVR to use automatic topology discovery (auto topology) frees you from having to configure IVR topology or maintain the IVR topology database. You only need to create IVR zones and zone sets. The MDS fabric creates, distributes, and synchronizes the IVR topology database automatically. When you create or remove a VSAN from an IVR-enabled switch, after approximately 45 seconds the new VSAN is added and distributed to the IVR topology database on the local and remote IVR enabled switches.

IVR's auto topology does have its drawbacks, which should not be overlooked. Auto topology adds every VSAN in every IVR-enabled switch into the topology. This can result in VSANs being unintentionally used as transit VSANs. See [Q.If I have multiple parallel transit VSANs, which VSAN is used?, page 8-4](#).

Some recipes in this chapter create topology manually, for example, [Configuring a Three Switch, Two Transit VSAN Topology with CFS, page 8-4](#), and some use auto topology, such as [Configuring a Single Switch with Two VSANs, page 8-25](#).



### Note

If an end device exists in a VSAN that is in the IVR topology database, it cannot access any other devices until it is part of an IVR zone.



## Transit VSANs

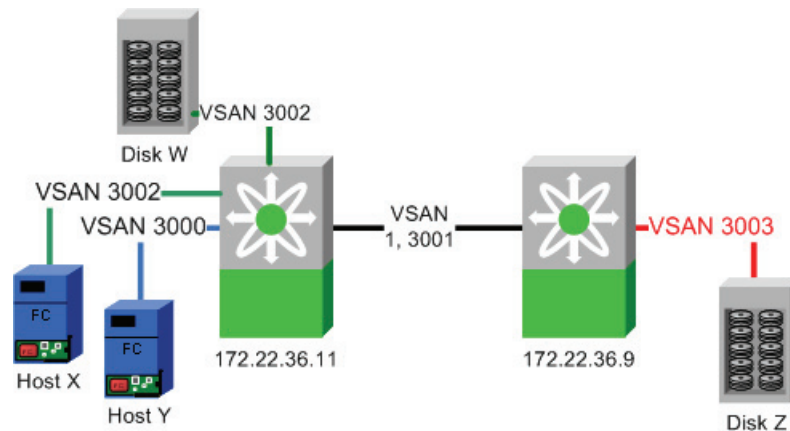
When creating an IVR configuration, you may need to specify VSANs in addition to the source and destination VSANs. VSANs that do not necessarily contain any actual end devices may be required in the IVR topology. These VSANs are known as transit VSANs and their sole purpose is connecting two VSANs together when no one switch contains both the source and destination VSAN. The most common example for using a transit VSAN is to configure only the transit VSAN to span a WAN link and not extend either the source or destination VSANs across the WAN.

In the example below, VSANs 1 and 3001 are potentially being used as a transit VSAN. These two VSANs are the only common VSANs between the two MDS switches, so they can and will be used as transit VSANs. (The IVR topology in [Figure 8-1](#) corresponds to the information in [Table 8-2](#).)

Table 8-2 IVR Topology

| VSAN Route Switch | VSANs to Route |
|-------------------|----------------|
| 172.22.36.11      | 1,3000-3002    |
| 172.22.36.9       | 1,3001, 3003   |

Figure 8-1 Example Topology Using a Transit VSAN



To specify a particular VSAN as a transit VSAN, no special configuration is required. The VSAN only needs to be part of the IVR topology to be used. It does not need to be empty, nor does it need to be in any specific Interop mode. It can have a mix of E and TE ports and can potentially contain nonMDS switches.

**Note**

**Q.** If I have multiple parallel transit VSANs, which VSAN is used?

**A.** The most direct path is used, that is, the path that uses the fewest VSAN hops. For example, a path that requires a frame to go through two different transit VSANs to reach the destination VSAN will not be chosen if there is a path that only requires one VSAN. This applies regardless of the Fabric Shortest Path First (FSPF) cost of the links inside the VSANs.

If there are two VSANs that have the same VSAN hop count (VSANs 1 and 3001 in [Figure 8-1](#)), then the one with the lowest VSAN ID is used. In the [Figure 8-1](#) example, VSAN 1 would be used as the transit VSAN to go from VSAN 3002 to VSAN 3003.

IVR does not load balance across transit VSANs, so IVR would use only VSAN 1 as the transit, unless VSAN 1 failed or became isolated.

**Tip**

- Only allow the VSAN you use as the transit VSAN to be trunked across the ISL.
- Leverage multiple paths within the transit VSAN as PortChannels and use FSPF to route around path failures.

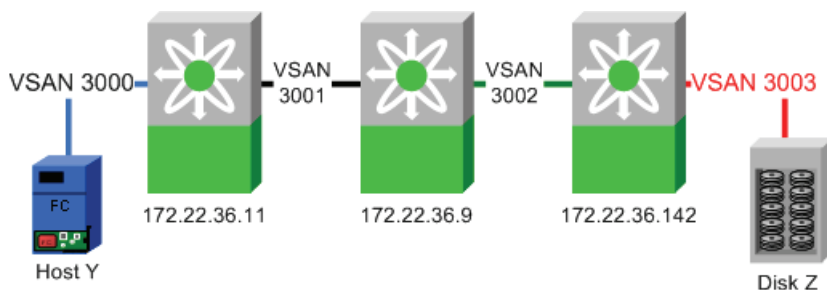
## Configuring a Three Switch, Two Transit VSAN Topology with CFS

This recipe configures the IVR topology for a configuration with three IVR switches using two transit VSANs (see [Figure 8-2](#)). The example uses CFS to distribute the topology. IVR has already been enabled on all three switches. This procedure can be used for IVR-1 or IVR-2 with FC NAT.

**Tip**

Instead of using multiple transit VSANs, use a single transit VSAN extended over multiple switches. It simplifies the topology while providing the isolation of a transit VSAN.

Figure 8-2 Three Switch, Dual Transit VSAN IVR Topology.



Before configuring a topology, decide what is needed. By examining the diagram in [Table 8-2](#), you can determine that the entries in [Table 8-3](#) need to be configured.

Table 8-3 IVR Topology Table

| VSAN Route Switch | VSANs to Route |
|-------------------|----------------|
| 172.22.36.11      | 3000, 3001     |
| 172.22.36.9       | 3001, 3002     |
| 172.22.36.142     | 3002, 3003     |



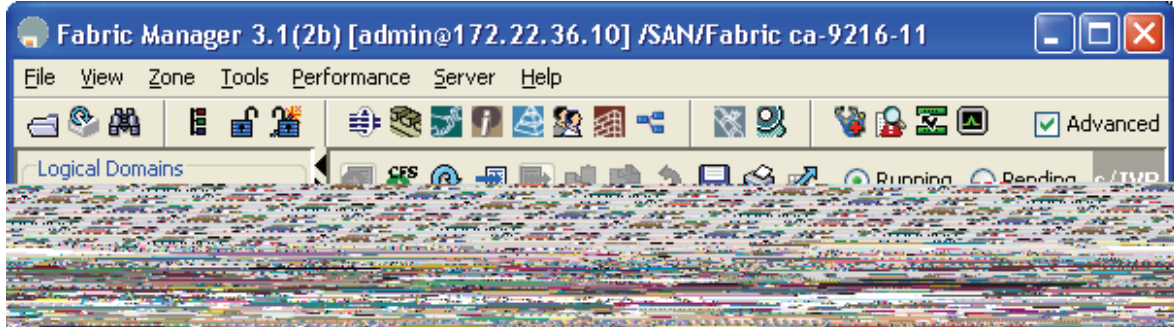
**Tip**

[Table 8-3](#) provides simple documentation that can be easily added to an implementation plan or detailed design document.

To configure the topology, follow these steps:

- Step 1** In the FM **Logical Domains** pane, select a fabric, then **All VSANs**, then **IVR**.
- Step 2** Choose the **CFS** tab to activate the other tabs.
- Step 3** Choose the tab **Local Topology** (see [Figure 8-3](#)).

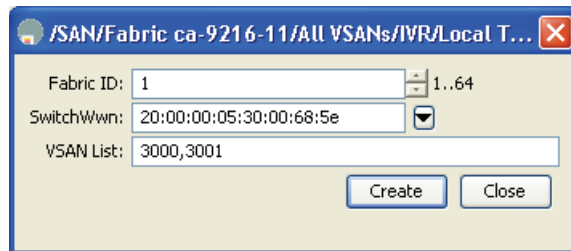
Figure 8-3 Local Topology Tab



- Step 4** Click the blue **Create Row...** icon (see [Figure 8-3](#)).
- Step 5** In the resulting pop up box, from the **Switch** pull-down menu, select the first switch (172.22.36.11) and its associated VSAN list. (This is part of the plan shown in [Table 8-3 on page 8-5](#).)

**Step 6** Complete the switch's **VSAN List** (3000, 3001) as shown in [Figure 8-4](#).

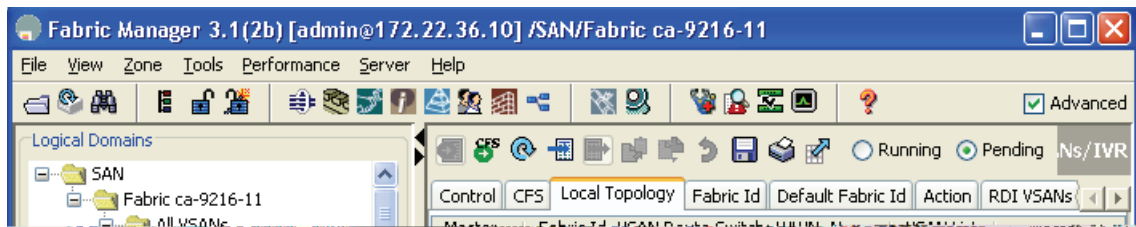
Figure 8-4 Create Topology



**Step 7** Click **Create**.

**Step 8** Repeat this procedure for the second and third switch in [Table 8-3](#) on [page 8-5](#), and then close the dialog box. The local topology should look like the one in [Figure 8-5](#).

Figure 8-5 Local Topology



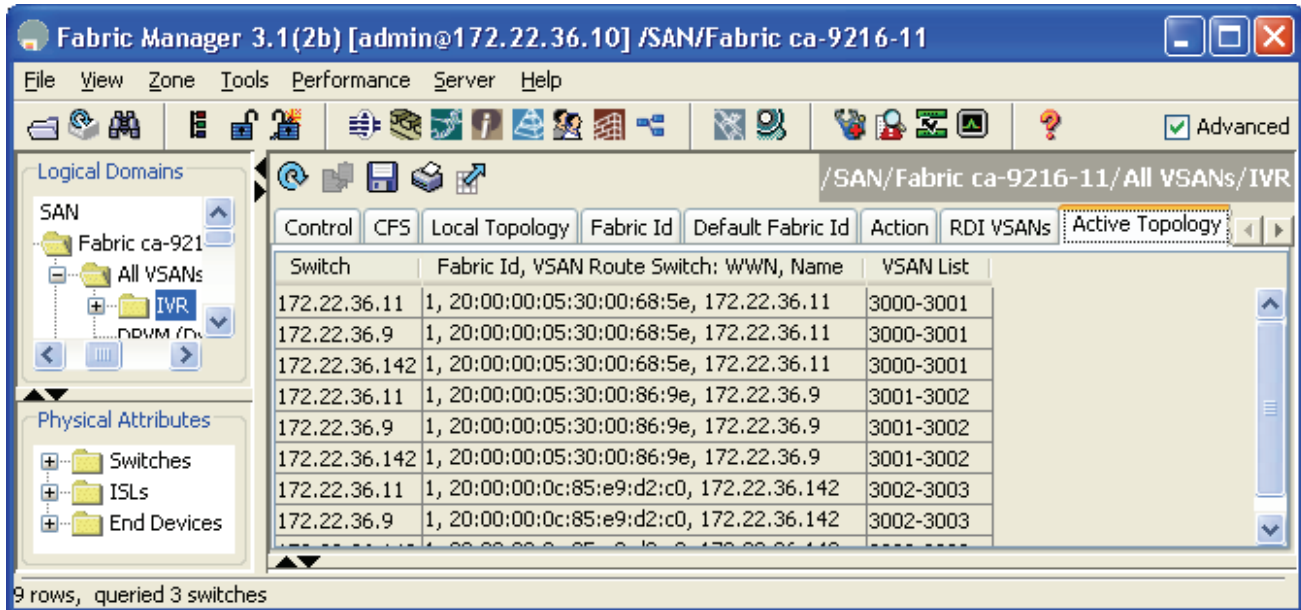
**Step 9** Choose the **Action** tab.

**Step 10** Check the **Activate Local** check box, and then click the green **Apply Changes** icon.

**Step 11** Click **Apply CFS** (green circle with CFS written above it). The message **CFS(ivr):Committed** appears in the bottom left corner of Fabric Manager.

**Step 12** To verify the topology, choose the **Active Topology** tab. You see the topology in Figure 8-6.

Figure 8-6 Active Topology



At this point, the topology is correctly defined and the active topology contains the correct information. When this topology is configured and distributed by CFS, the **Discrepancies** tab will have no entries. Now IVR zone sets can be defined to provide connectivity between the two end devices.

## IVR Zones and Zone Sets

IVR zones and zone sets, the objects that allow an end device in one VSAN to communicate with an end device in another VSAN, have the same features and functionality as a regular zone or zone set with one exception: the zone members are in different VSANs.

Members of IVR zones can be pWWNs or device aliases. Registered State Change Notifications (RSCNs) are restricted to the device within the IVR zone that triggered the RSCN. IVR zone names automatically have the prefix “IVRZ\_” so they are easily identified in an active zone set.

```
switch: show zoneset active vsan 3000
zoneset name ZoneSet1 vsan 3000
zone name Zone1 vsan 3000
 pwwn 50:06:0e:80:03:4e:95:23 [HDS20117-c20-9]
 pwwn 21:00:00:e0:8b:09:78:88 [ca-aix_lpf0]
zone name IVRZ_IvrZone1 vsan 3000
 pwwn 50:06:0e:80:03:4e:95:23 [HDS20117-c20-9]
 pwwn 21:00:00:e0:8b:09:78:47 [ca-sun2_q1c0]
```

IVR zones must be members of IVR zone sets just as regular zones must be members of regular zone sets. An IVR zone set must also be activated in order to be part of the running configuration. There is still only one active zone set at activation, consisting of regular zones and IVR zones, so a switch that is not IVR-enabled (either a non-IVR enabled MDS or a third-party switch) can still receive and apply the new active zone set.

**Tip**

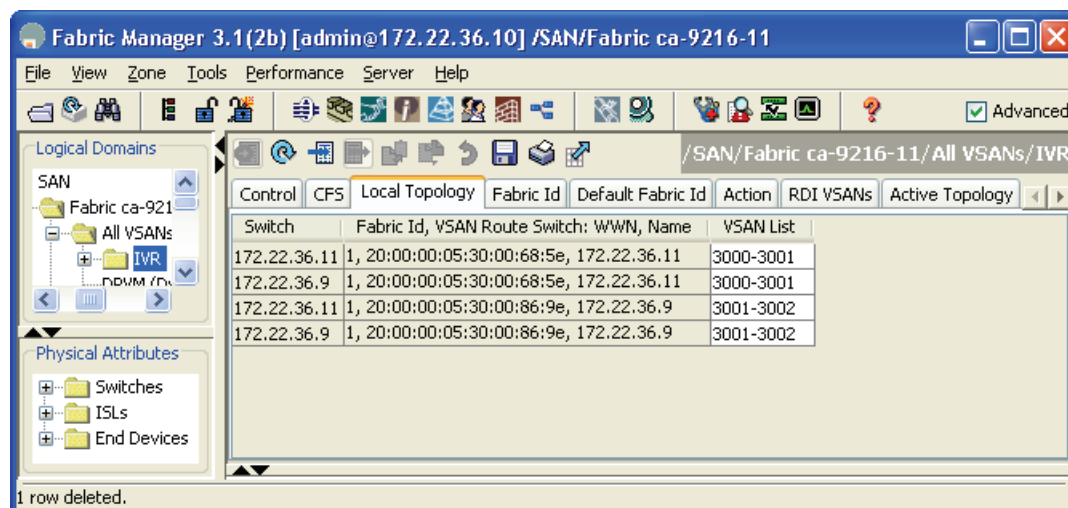
- If there are devices in the VSAN that require IVR access, the IVR-enabled switch should be used for all zoning, including nonIVR zoning, because the resulting active zone set is a union of the regular zone set and the IVR zone set.
- Do not use IVR zones to provide access between devices in the same VSAN. Use regular zones.
- Do not use regular zones to provide access between a real device and the pseudo device created by IVR. Use IVR zones.
- Both VSANs referenced in an IVR zone must be in the IVR topology to communicate.

## IVR with CFS

Before Cisco SAN-OS Release 2.x, IVR topology had to be defined on each switch using either Fabric Manager or the CLI. If a new switch was going to perform IVR, the entire IVR topology had to be manually entered on the new switch, then the other switches each had to be modified to include the new switch. For example, with the old method, a fabric with three switches required nine entries into the topology database (3 switches with IVR \* 3 VSAN route switches).

Figure 8-7 is an example of a two-switch topology configuration without CFS. There are two switches with IVR times two VSAN route switches for a total of four entries.

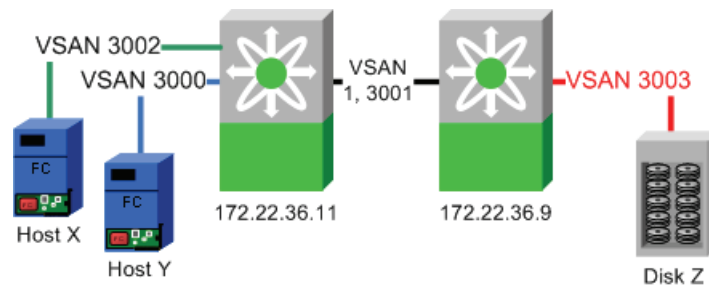
Figure 8-7 IVR Topology without CFS



Modifying a topology is still a manual operation, but you no longer need to modify each switch individually. With the introduction of CFS support for IVR, a single topology is maintained and CFS distributes changes to other IVR switches in the fabric. If a new switch is added to the fabric, CFS automatically synchronizes the new switch with the existing IVR topology.

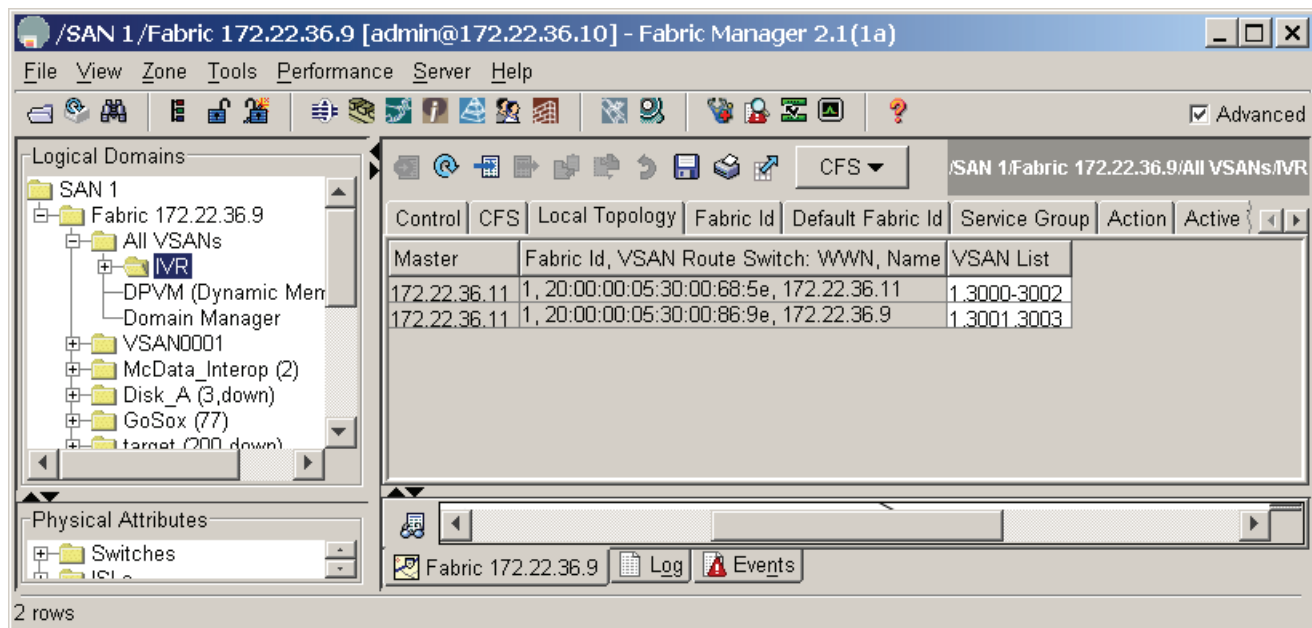
The topology described in Figure 8-7 is illustrated in Figure 8-8 and shown again, using CFS this time, in Figure 8-9.

Figure 8-8 CFS Reference Topology



The two-switch configuration with CFS has one row for each VSAN Route Switch as shown in Figure 8-9.

Figure 8-9 IVR Topology with CFS



The topology using CFS is easier to comprehend, since only one row per IVR-enabled switch is displayed. The first column represents the switch that Fabric Manager uses to perform CFS operations. Columns two and three describe the routes. 172.22.36.11 routes between VSANs 1, 3000-3002, while switch 172.22.36.9 routes between 1, 3001 and 3003. CFS prevents duplicate information from being displayed as the topology is managed on a fabric basis rather than a per switch basis.

**Note**

- If CFS is to be used with IVR, all IVR enabled switches must have CFS distribution for IVR enabled.
- Conversely, If CFS is not going to be used for IVR, then all of the IVR enabled switches should have CFS distribution for IVR disabled.

# IVR-1

The IVR-1 method of IVR has existed since Cisco SAN-OS Release 1.3 and does not do any NAT functions. It requires unique VSAN and domain IDs across the IVR topology.

IVR-1 can use CFS for configuration distribution and application locking. Although IVR-1 can be used without CFS, the recipes in this chapter use CFS.

IVR-1 is first enabled, then configured.



## Note

IVR-1 requires that unique domain\_IDs be used throughout the IVR topology.

## Enabling IVR-1

IVR-1 must be first enabled, then configured. Enabling IVR-1 can be done either from the CLI or with Fabric Manager. Use the same method (CLI or Fabric Manager) to enable all switches acting as border switches or all switches that route frames between VSANs. Enabling CFS must also be consistent; either all IVR switches in a fabric have CFS enabled for IVR, or none of them do.

### Enabling IVR-1 with the CLI

To enable switches for IVR-1 from the CLI, follow these steps:

- 
- Step 1** Enter configure terminal mode and enable IVR with the **ivr enable** command.
- ```
172.22.36.11# conf terminal
Enter configuration commands, one per line. End with CNTL/Z.
172.22.36.11(config)# ivr enable
```
- Step 2** By default, CFS is not enabled with IVR-1, so enable CFS distribution. Notice that the scope is physical as IVR crosses VSANs.
- ```
172.22.36.11(config)# ivr distribute
172.22.36.11(config)# do show cfs application name ivr

Enabled : Yes
Timeout : 300s
Merge Capable : Yes
Scope : Physical-fc
```
- Step 3** Verify that all the switches are recognized by IVR using the **show cfs peers name ivr** command.
- ```
172.22.36.11# show cfs peers name ivr

Scope              : Physical
-----
Switch WWN          IP Address
-----
20:00:00:05:30:00:68:5e  172.22.36.11    [Local]

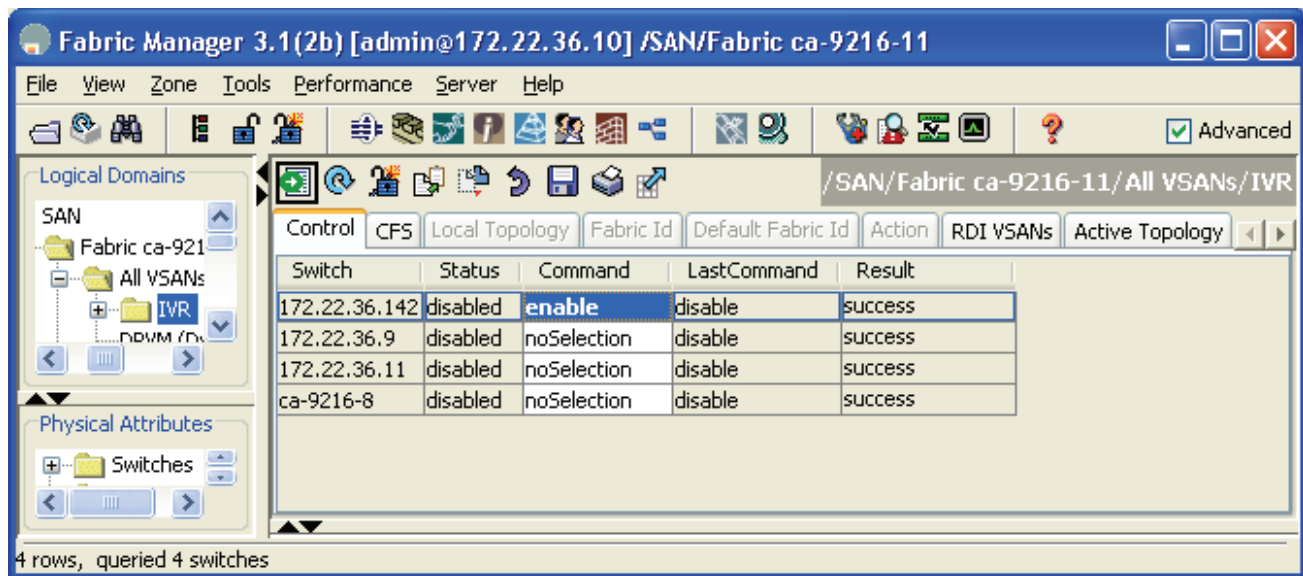
Total number of entries = 1
```
-

Enabling IVR-1 with Fabric Manager

To enable switches for IVR-1 from Fabric Manager, follow these steps:

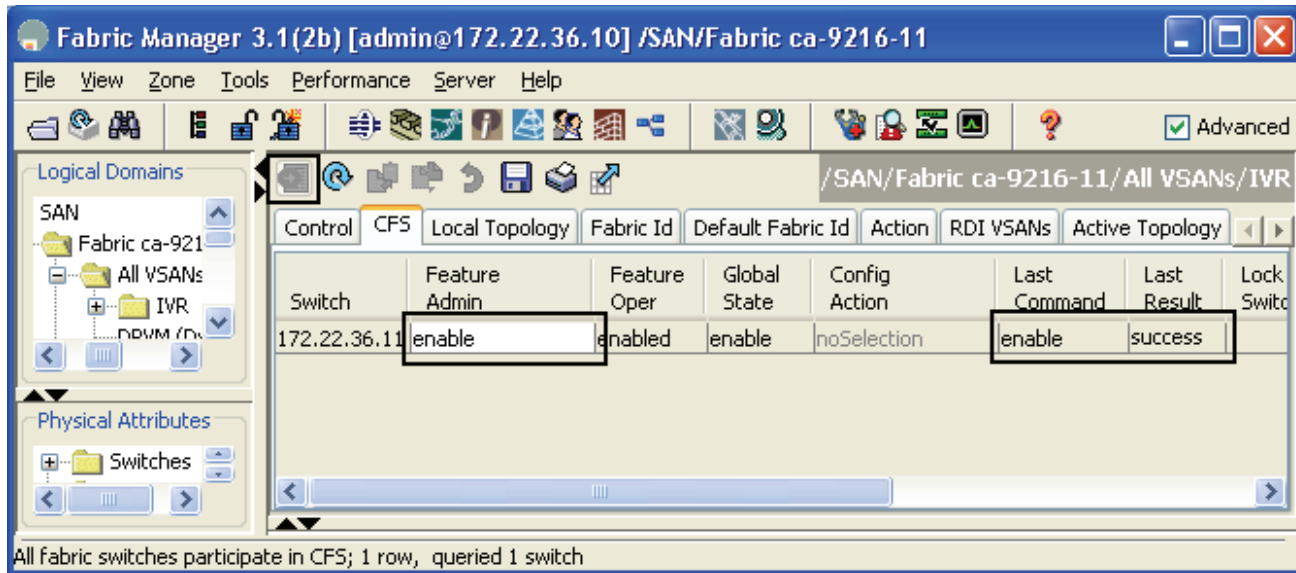
- Step 1** In the Logical Domains pane, expand **All VSANs** then select **IVR**.
- Step 2** Under the Control tab, change the Command option to **enable** for the switches that should have IVR enabled (see [Figure 8-10](#)).
- Step 3** Click the green **Apply Changes** icon shown in [Figure 8-10](#). The status field changes from disabled to enabled.

Figure 8-10 Enable IVR in Fabric Manager



- Step 4** Enable CFS distribution for IVR by following these steps:
 - a. Choose the **CFS** tab (see [Figure 8-11](#)).
 - b. In the **Enable Admin** column, change the option from noSelection to **enable** (see [Figure 8-11](#)).
 - c. Click the green **Apply Changes** icon (see [Figure 8-11](#)). The column Last should display **success**.

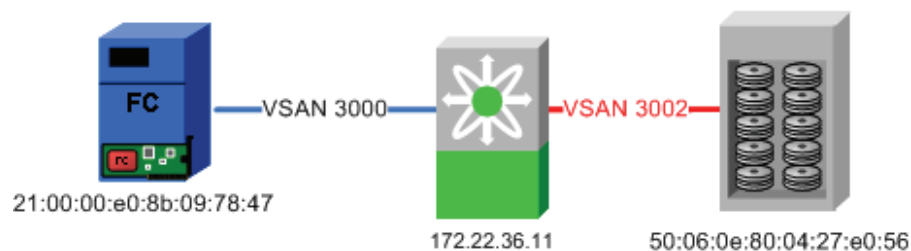
Figure 8-11 Enable CFS Distribution for IVR



Configuring a Single Switch and Two VSANs

In this recipe, the most basic IVR environment is configured with Fabric Manager. This environment has one MDS switch and two VSANs (see Figure 8-12). CFS is used.

Figure 8-12 Single Switch IVR-1, Two VSAN Topology



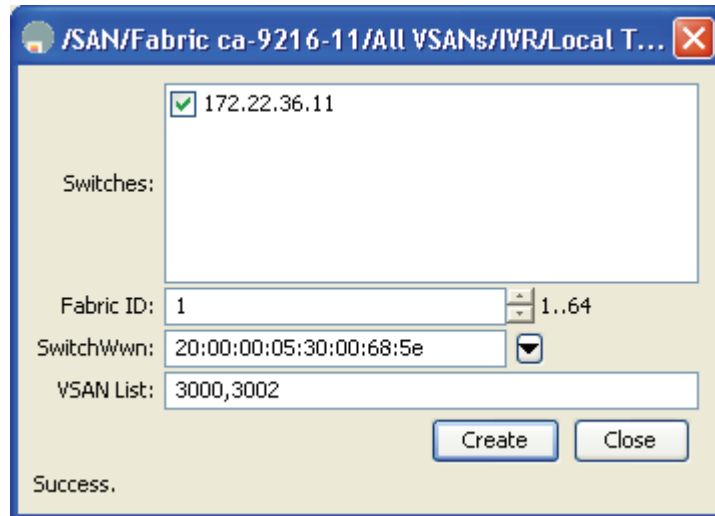
Creating the IVR Topology

To create the IVR topology from Fabric Manager, follow these steps:

- Step 1** Enable IVR as described in [Enabling IVR-1, page 8-10](#).
- Step 2** In the Fabric Manager Logical Domains window, choose the fabric, expand **All VSANs** and select **IVR**.
- Step 3** In the Local Topology tab, click **Create Row...**
- Step 4** In the VSAN list (see [Figure 8-13](#)) enter the VSANs to be routed (3000, 3002).

- Step 5** Expand the Switch List and choose a switch (see [Figure 8-13](#)).
- Step 6** Click **Create**. The word success should be displayed in the bottom of the window.
- Step 7** Close the window in [Figure 8-13](#).

Figure 8-13 Single Switch IVR-1 Create Topology



- Step 8** To close the dialog box, click **Close**.
To activate the topology from Fabric Manager, follow these steps:
- Step 9** Click the **Action** tab.
- Step 10** Check the **Activate Local** check box.
- Step 11** Click the green **Apply Changes** icon. Note that the topology is not active until it has been CFS committed (follow the next steps).
To CFS commit the topology from Fabric Manager, follow these steps:
- Step 12** Click the **Commit CFS Pending Changes** button.
In the bottom left-hand corner of FM, the message CFS(ivr):Committed should be displayed. Also, the Active Topology tab contents should match those of the Local Topology tab.
The next phase is to create the IVR zone set and accompanying zones.

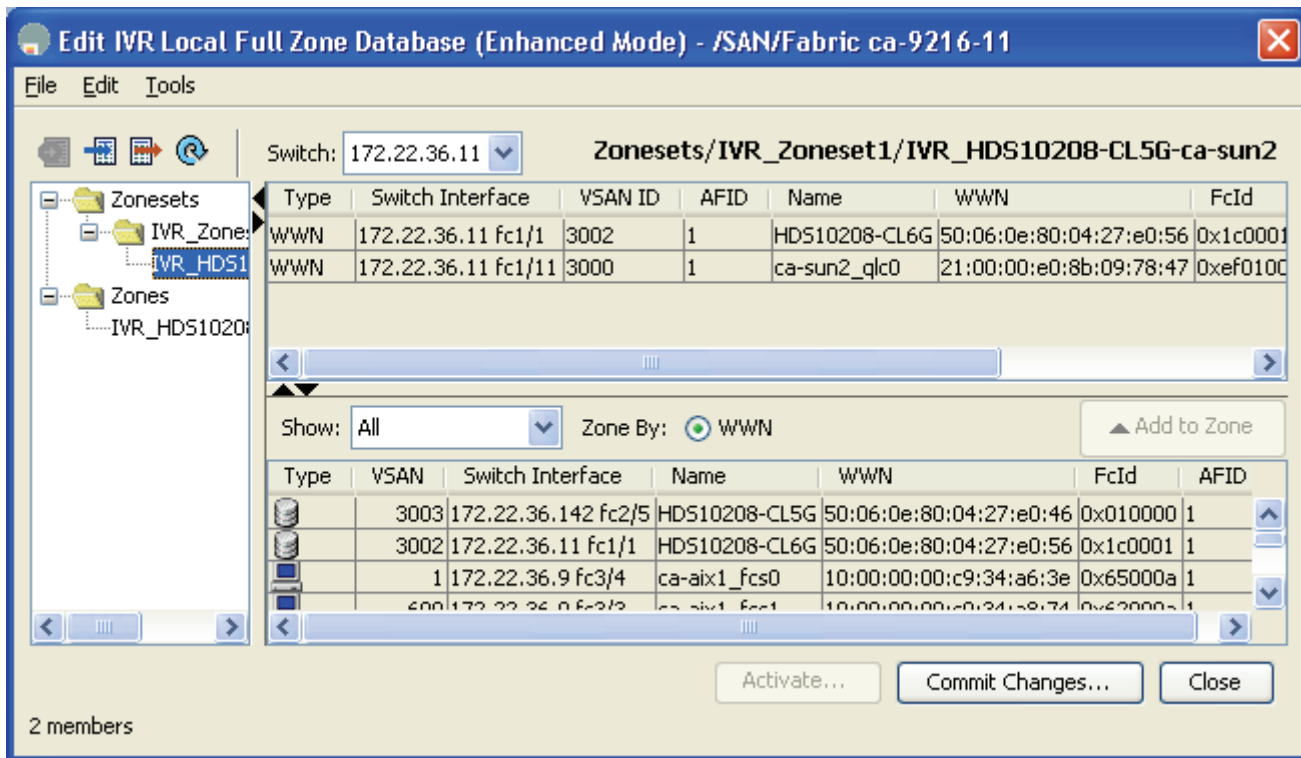
Creating the IVR Zone Set and Zones

To create the IVR zone set and accompanying zones from Fabric Manager, follow these steps:

- Step 1** In the Logical Domains pane, select the fabric, expand **All VSANs**, and then right-click **IVR** (see [Figure 8-14](#)).
- Step 2** Choose **Edit Local Full Zone Database**.
- Step 3** In the resulting dialog box, create an IVR zone set by right-clicking **Zonesets** and choosing **Insert**.
- Step 4** Enter a name for the zone set and click **OK**. The resulting zone set appears in the left pane.

- Step 5** Create the IVR Zone by right-clicking **Zones** in the left pane and choosing **Insert**.
- Step 6** Enter a meaningful zone name and click **OK**.
- Step 7** Drag devices to be zoned from the bottom pane into the newly created zone in the top pane of the screen (see Figure 8-14).
- Step 8** In the left pane, drag the zone into the zone set (see Figure 8-14).

Figure 8-14 Single Switch IVR-1, Create a Zone Set



- Step 9** Activate the zone set and implicitly CFS commit it by right-clicking the zone set and selecting **Activate**.
- Step 10** Click **Continue Activation**. A popup dialog displaying the steps taken during activation will display information similar to the following:

```

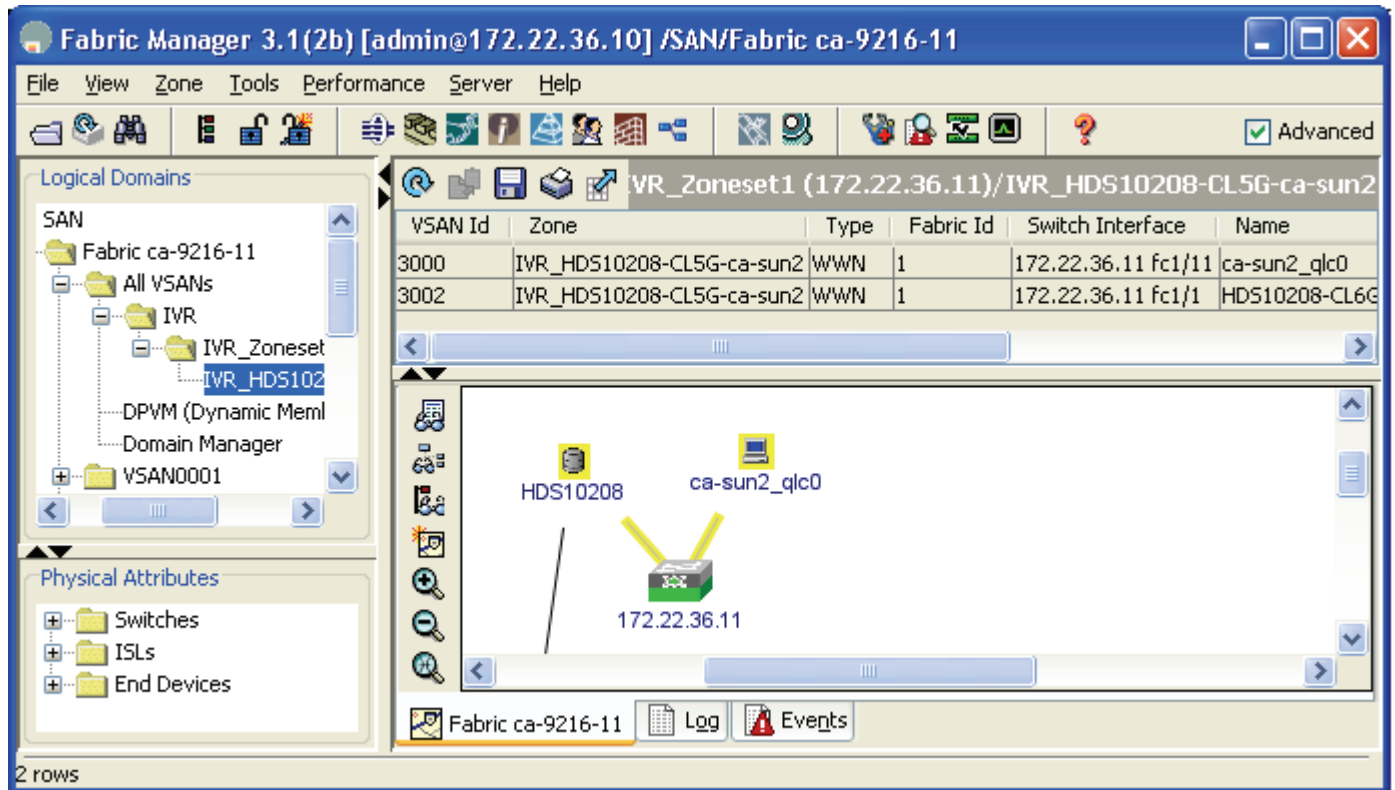
172.22.36.11:Activating IVR_Zoneset1
172.22.36.11:checking status, elapsed time:0 sec activating IVR_Zoneset1
172.22.36.11:checking status, elapsed time:0 sec activating IVR_Zoneset1
Switch:172.22.36.11
  VSAN id:3000 status :active
  VSAN id:3002 status :active
  committing zone configuration changes
172.22.36.11:172.22.36.11:Commit Successful
172.22.36.11:checking status, elapsed time:3 sec activating IVR_Zoneset1
172.22.36.11:IVR_Zoneset1 Activation success
172.22.36.11:Save running configuration to Startup
172.22.36.11:Save running configuration to Startup on Enhanced 172.22.36.11
172.22.36.11:Saved running configuration to Startup
172.22.36.11:Finished
Success

```

- Step 11** Click **Close**. You will see the main Fabric Manager window again.

Step 12 Display the map of IVR members by selecting the IVR Zone in the Logical Domains pane (see Figure 8-15).

Figure 8-15 Single Switch IVR-1, Display IVR Zone Set



IVR-2 with FC NAT

In Cisco SAN-OS Release 2.1(1a), Cisco introduced IVR with FC NAT. NAT provides the ability to:

- Route between fabrics containing duplicate domain IDs.
- Route between VSANs.
- Route between two VSANs of the same VSAN ID.

In addition, instead of representing each native VSAN domain with a unique virtual domain, a single virtual domain represents all of the domains in the native VSAN. Finally, IVR-FC NAT, leveraging the CFS infrastructure, automatically discovers topology, alleviating the process of configuring and maintaining IVR topology.

Enabling IVR-2 (FC NAT)

This recipe uses Fabric Manager to enable IVR-2 with CFS and auto topology discovery.

**Note**

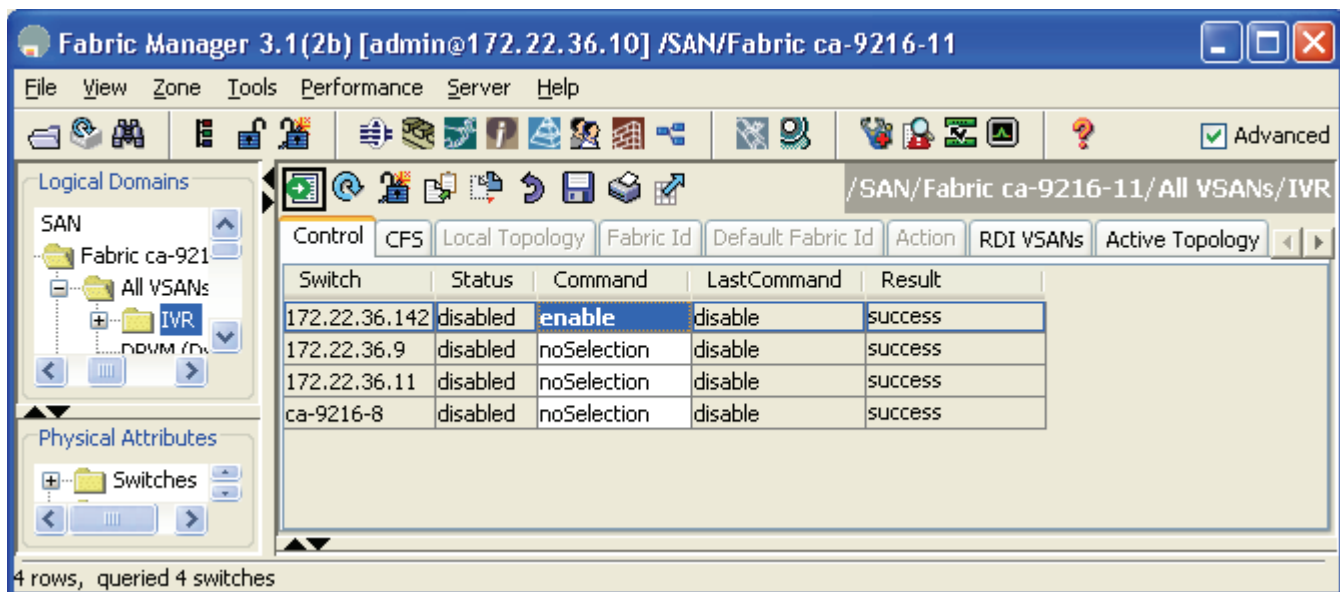
In a multiswitch topology, repeat these steps for each switch in the fabric. Remember, that CFS commands (for example, adding NAT and auto topology) only have to be done from one switch as CFS informs the other switches.

To enable IVR-2 with CFS and auto topology discovery, follow these steps:

- Step 1** In the Logical Domains pane, expand **All VSANs**, then select **IVR** as shown in [Figure 8-16](#).
- Step 2** Under the Control tab, change the Command column entry to **enable** for switches that should have IVR enabled.
- Step 3** Click the green **Apply Changes** icon shown in [Figure 8-16](#).

The Status field entries change from disabled to **enabled** for the switches you selected in Step 2.

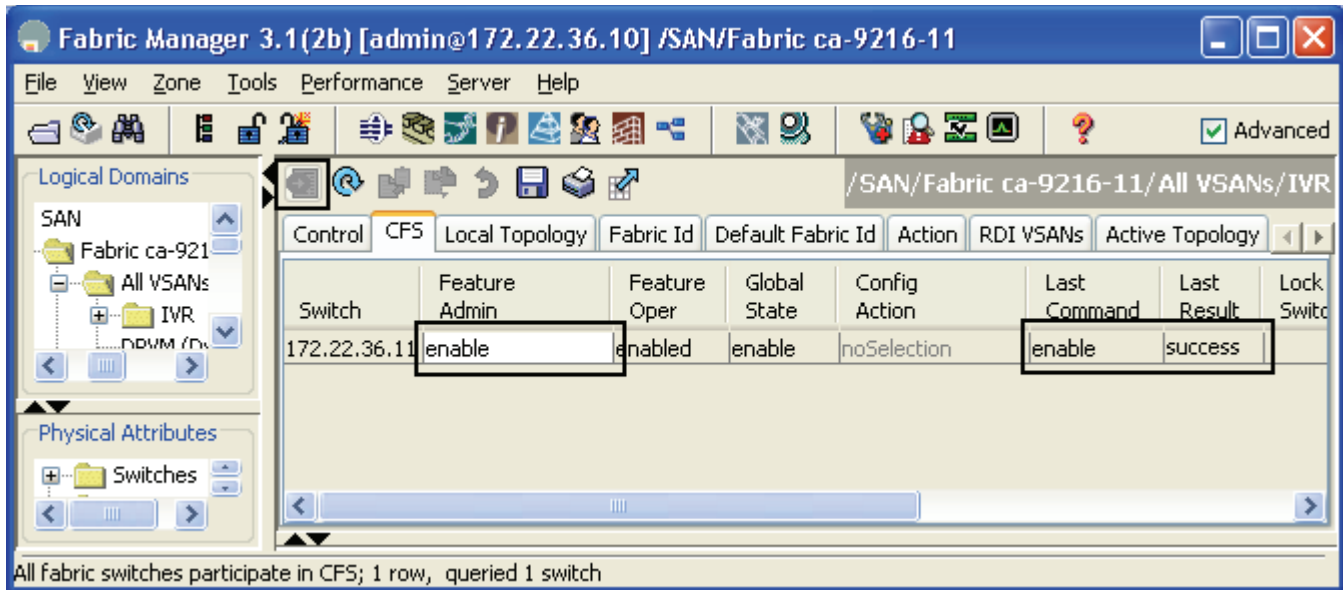
Figure 8-16 Enable IVR in Fabric Manager



IVR is enabled.

- Step 4** Enable CFS distribution for IVR following these steps:
 - a. Choose the **CFS** tab.
 - b. Under Enable Admin, change noSelection (shown in [Figure 8-17](#)) to enable.
 - c. Click the green **Apply Changes** icon. The Last Result changes to success as shown in [Figure 8-17](#).

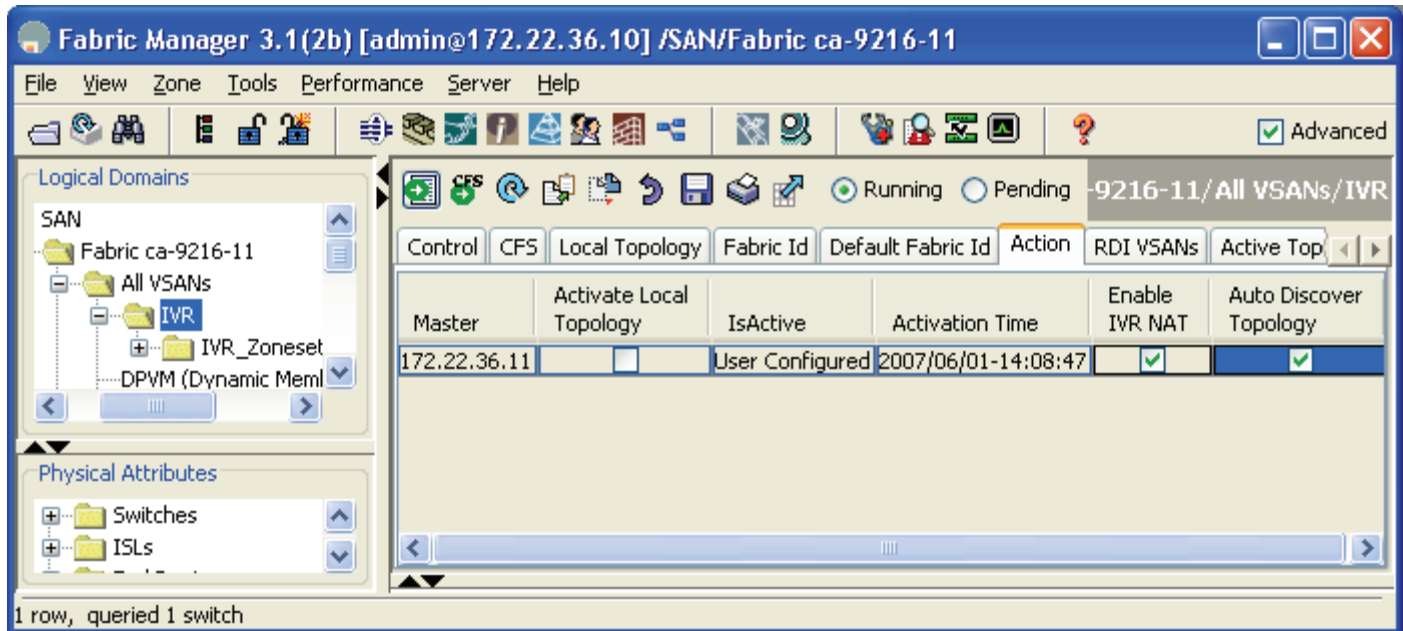
Figure 8-17 Enable CFS Distribution for IVR

**Note**

Auto topology is not required for IVR-2 with FC NAT. If you do not enable it, manually define the topology. In this example, if you do not use auto topology, skip [Step 6](#).

- Step 5** Enable the FC NAT function of IVR by selecting the **Action** tab and checking the **Enable IVR NAT** check box (see [Figure 8-18](#)).
- Step 6** Enable the auto topology discovery function of IVR by checking the **Automatically Discover Topology** check box (see [Figure 8-18](#)).
- Step 7** Click the green **Apply Changes** icon (see [Figure 8-18](#)). Remember that the configuration is not active until it has been CFS committed.

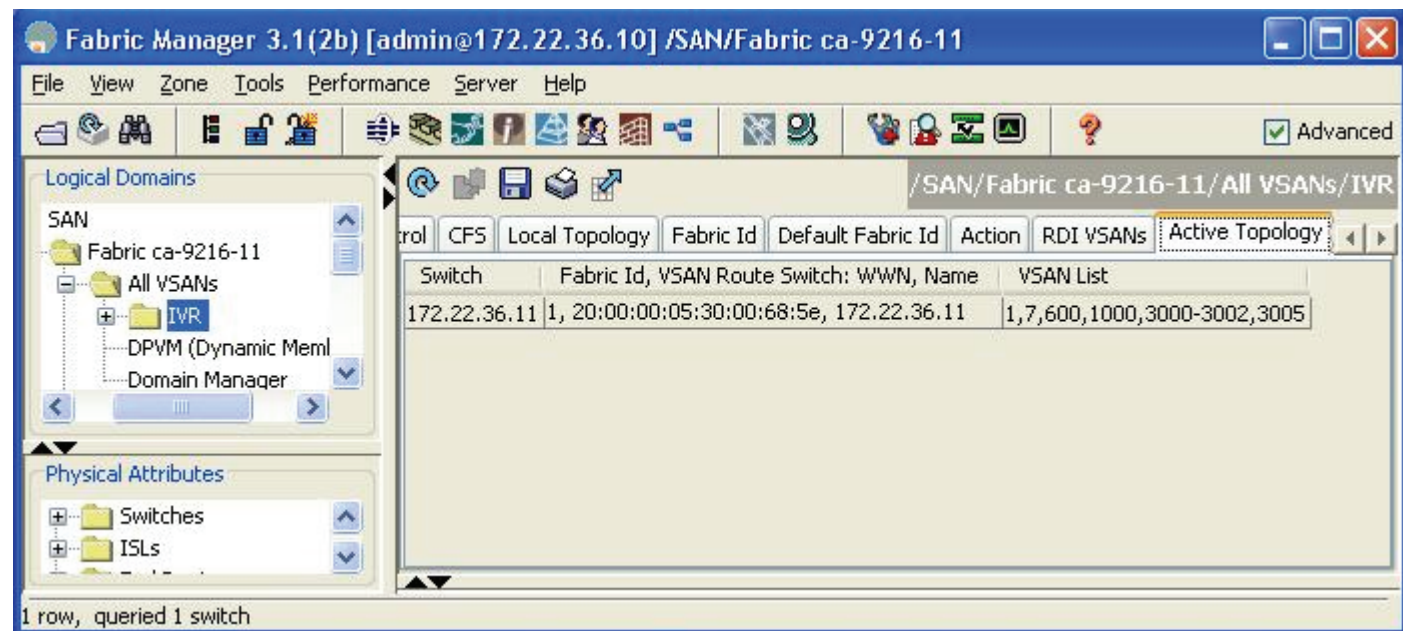
Figure 8-18 Enabling IVR-2's FC NAT and Auto Topology Discovery



Step 8 Click **CFS Commit Pending Changes**. **CFS(ivr): Committed** should be displayed in the bottom left hand corner.

Step 9 Choose the **Active Topology** tab to see the active topology as shown in Figure 8-19.

Figure 8-19 IVR-2 FC NAT Auto Discovered Topology



At this point IVR-2 is enabled for CFS distribution, NAT, and auto topology discovery.

Upgrading from IVR-1 to IVR-2

This recipe upgrades an IVR-1 configuration to IVR-2 using both CFS and auto topology.

**Caution**

Upgrading from IVR-1 to IVR-2 disrupts IVR-based traffic, but does not disrupt nonIVR traffic such as nonIVR zones contained in a VSAN.

Upgrading from IVR-1 to IVR-2 with NAT may change the FC IDs of virtual devices, which requires FC ID dependent hosts, such as HP-UX and AIX, to rescan for the devices.

**Note**

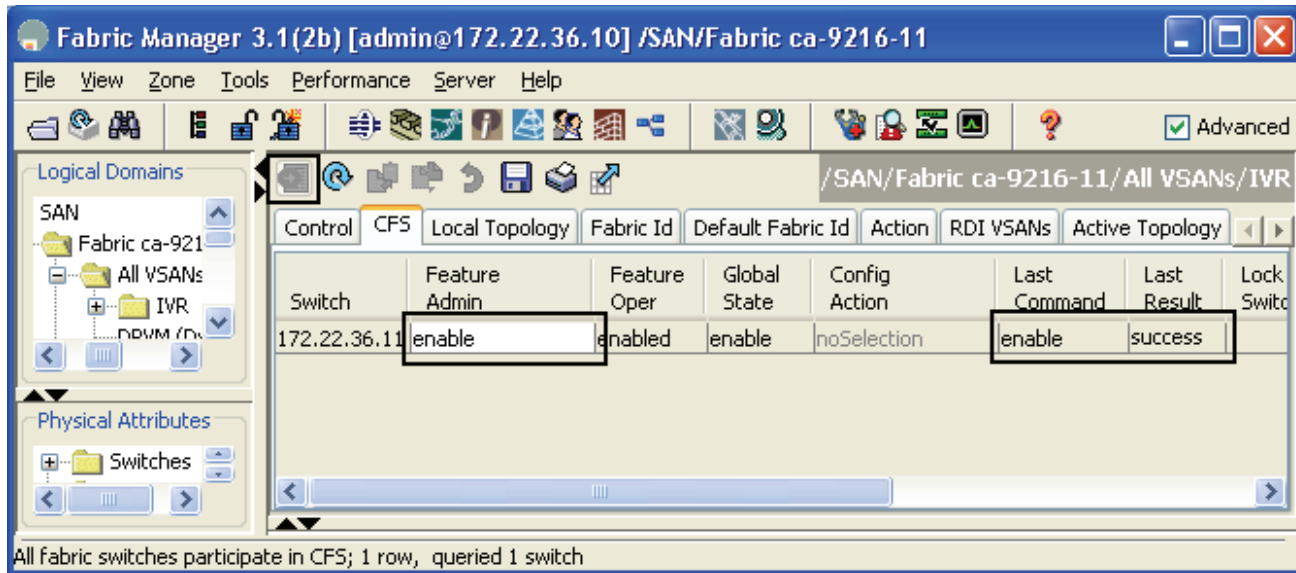
IVR-1 cannot coexist with IVR-2 within a single physical fabric. Border switches running IVR must be either in IVR-1 mode or IVR-2 mode. Mixed configurations are not supported. Use CFS to ensure that all switches are running the same configuration.

To upgrade an IVR-1 configuration to IVR-2 using both CFS and auto topology, follow these steps:

- Step 1** Back up all MDS switch configurations using a procedure similar to [Copying Files to and from a Switch](#), page 1-17.
- Step 2** Upgrade all IVR border switches to Cisco SAN-OS Release 2.1(1a) or later, as IVR-2 with FC NAT was first introduced in Cisco SAN-OS Release 2.1(1a). Switches not acting as IVR border switches are not required to be upgraded, but we recommended that you upgrade them. For upgrading SAN-OS firmware directions, see [Firmware Upgrades and Downgrades](#), page 1-20.
- Step 3** Deactivate the IVR zone set using Fabric Manager. This does not delete the IVR zone set from the local database as it is reactivated once IVR-2 has been enabled.
 - a. In the Logical Domains pane, select a fabric and expand **All VSANs**.
 - b. Right-click **IVR** and choose **Deactivate Zoneset**. A pop-up window opens.
 - c. Click **OK**.

All devices are now isolated from devices in other VSANs.
 - d. Close the pop-up window.
- Step 4** Enable CFS for IVR for all switches that will perform IVR, using Fabric Manager.
 - a. In the Logical Domains pane, select a fabric, expand **All VSANs** and select IVR.
 - b. Choose the **CFS** tab in the top pane (see [Figure 8-20](#)).
 - c. In the Enable Admin column, change the field from noSelection to **enable**.
 - d. Click the green **Apply Changes** icon. The Last command columns should now display enable and success (see [Figure 8-20](#)).

Figure 8-20 Enable CFS



- Step 5** Enable FC NAT and auto topology.
- Click the **CFS** tab to determine the CFS master switch.
 - Click the **Action** tab. Only one switch should be listed here, the switch that FM will use to perform configuration distribution.
 - Check both the **Enable IVR NAT** and the **Automatically Discover Topology** check boxes.
 - Click the green **Apply Changes** icon.

**Note**

A local topology may still be present on the switch. Auto topology modifies the local topology database.

- Step 6** Click **CFS Commit Pending Changes**. **CFS(ivr): Committed** should be displayed in the bottom left-hand corner.
- Step 7** Activate the IVR zone set.
- In the Logical Domains pane, select the fabric and expand **All VSANs**.
 - Right-click **IVR** and choose **Edit Local Full Zone Database**.
- Step 8** Right-click the zone set that was deactivated in Step 3 and choose **Activate**.
- Step 9** Click **Continue Activation** in the resulting confirmation window. It takes a few seconds to commit the changes and save the running-configuration to startup.
- Step 10** Click **Close** to return to the main FM window.

At this point the switches are upgraded to IVR-2 with FC NAT, CFS and auto topology. HPUX and AIX hosts with disks associated with the FC ID may need to re scan for the new FC IDs.

Configuring Persistent FC IDs in IVR from the CLI

Starting with Cisco SAN-OS Release 2.1(2), virtual devices created by an IVR with NAT configuration can have associated persistent FC IDs. This feature, similar to the persistent FC ID feature discussed in [Chapter 6, “VSANs”](#) for actual devices, enables a virtual device to receive the same FC ID across reboots of the switch.



Tip

HPUX and AIX are two operating systems that use FC IDs in device paths to storage. If the FC ID changes for a device accessed by either an AIX or a HPUX host, the host might lose access to the device. Configure persistent FC IDs for IVR to have a switch assign the same FC ID to a virtual device across switch reboots.

This example configures a storage device to use a specific FC ID in the host's VSAN. The actual devices are already logged into the fabric and the IVR topology has been created to include VSANs 3000 and 3002. In addition, this example uses these resources:

- IVR features enabled: CFS and IVR with NAT.
- Host with pWWN: 10:00:00:00:c9:32:8b:a8 and VSAN 3002.
- Storage with pWWN: 50:06:0e:80:03:4e:95:33, VSAN 3000, Real FC ID: 0xef0002, and FC ID to be configured in Host VSAN: 0x630063.
- IVR topology:

| AFID | SWITCH | WWN | | Active | Cfg. | VSANS |
|------|-------------------------|-----|-----|--------|-----------|-------|
| 1 | 20:00:00:05:30:00:68:5e | * | yes | yes | 3000,3002 | |

To configure a storage device to use a specific FC ID in the host VSAN, follow these steps:

- Step 1** Enter IVR FC domain configuration mode for the autonomous fabric ID (AFID) and VSAN for the location of the virtual device. In this case it is the AFID and VSAN where the storage will be virtual.
- ```
switch# conf t
switch(config)# ivr fcdomain database autonomous-fabric-num 1 vsan 3002
switch(config-fcdomain)#
```
- Step 2** Enter the native AFID, VSAN of the storage device, and domain to be used in the host's VSAN. CFS is enabled for IVR so any changes must be committed later. The domain ID in this command is in decimal format.
- ```
switch(config-fcdomain)# native-autonomous-fabric-num 1 native-vsan 3000 domain 99
fabric is locked for configuration. Please commit after configuration is done.
switch(config-fcdomain-fcid)#
```
- Step 3** Specify the pWWN and FC ID to be used. (A device-alias can be used instead of a pWWN.) The virtual domain and FC IDs are not created until the zone set is activated.
- ```
switch(config-fcdomain-fcid)# pwwn 50:06:0e:80:03:4e:95:33 fcid 0x630063
```
- The FCID should correspond to virtual domain 99 specified earlier for this mode
- ```
switch(config-fcdomain-fcid)#
```
- Step 4** Create the IVR zone sets and zones with **ivr zoneset** commands.
- ```
switch(config-fcdomain-fcid)# ivr zoneset name IVR_Zoneset1
switch(config-ivr-zoneset)# zone name IVRZ_host1_lpf0_Array1_port12
switch(config-ivr-zoneset-zone)# member pwwn 10:00:00:00:c9:32:8b:a8 vsan 3002
switch(config-ivr-zoneset-zone)# member pwwn 50:06:0e:80:03:4e:95:33 vsan 3000
```

```
switch(config-ivr-zoneset-zone)# ivr zoneset activate name IVR_Zoneset1
```

- Step 5** CFS commit the changes for IVR to activate both the IVR zone set and the modifications to the IVR persistent FC ID database.

```
switch(config)# ivr commit
commit initiated. check ivr status
```

- Step 6** Verify FC IDs and active zone set with the **show** command.

```
switch(config)# do show ivr fcdomain database

AFID Vsan Native-AFID Native-Vsan Virtual-domain

1 3002 1 3000 0x63(99)
```

Number of Virtual-domain entries: 1

```

AFID Vsan Pwwn Virtual-fcid

1 3002 50:06:0e:80:03:4e:95:33 0x630063
[HDS20117-c20-8]
```

Number of Virtual-fcid entries: 1

```
switch# show fcns database vsan 3002
```

VSAN 3002:

```

FCID TYPE PWWN (VENDOR) FC4-TYPE:FEATURE

0x1c0003 N 10:00:00:00:c9:32:8b:a8 (Emulex) scsi-fcp
[ca-sun1_lpf0]
0x630063 N 50:06:0e:80:03:4e:95:33 [HDS20117-c20-8] scsi-fcp
```

Total number of entries = 2

```
switch# show zoneset active vsan 3002
```

```
zoneset name ZS_test vsan 3002
zone name zone1 vsan 3002
 pwwn 50:06:0e:80:03:4e:99:99
 pwwn 50:06:0e:80:03:4e:98:98

zone name IVRZ_IVRZ_host1_lpf0_Array1_port12 vsan 3002
* fcid 0x630063 [pwwn 50:06:0e:80:03:4e:95:33] [HDS20117-c20-8]
* fcid 0x1c0003 [pwwn 10:00:00:00:c9:32:8b:a8] [ca-sun1_lpf0]
```

## Configuring Persistent FC IDs in IVR Using Fabric Manager

Starting with Cisco SAN-OS Release 2.1(2), virtual devices created by an IVR with NAT configuration can have associated persistent FC IDs. This feature, similar to the persistent FC ID feature discussed in [Chapter 6, “VSANs”](#) for actual devices, enables a virtual device to receive the same FC ID across reboots of the switch.

**Tip**

HPUX and AIX are two operating systems that use FC IDs in device paths to storage. If the FC ID changes for a device accessed by either an AIX or a HPUX host, the host may lose access to the device. Configure persistent FC IDs for IVR to have a switch assign the same FC ID to a virtual device across switch reboots.

This example configures a storage device to use a specific FC ID in the host's VSAN. The actual devices are already logged into the fabric and the IVR topology has been created to include VSANs 3000 and 3002. In addition, this example uses these resources:

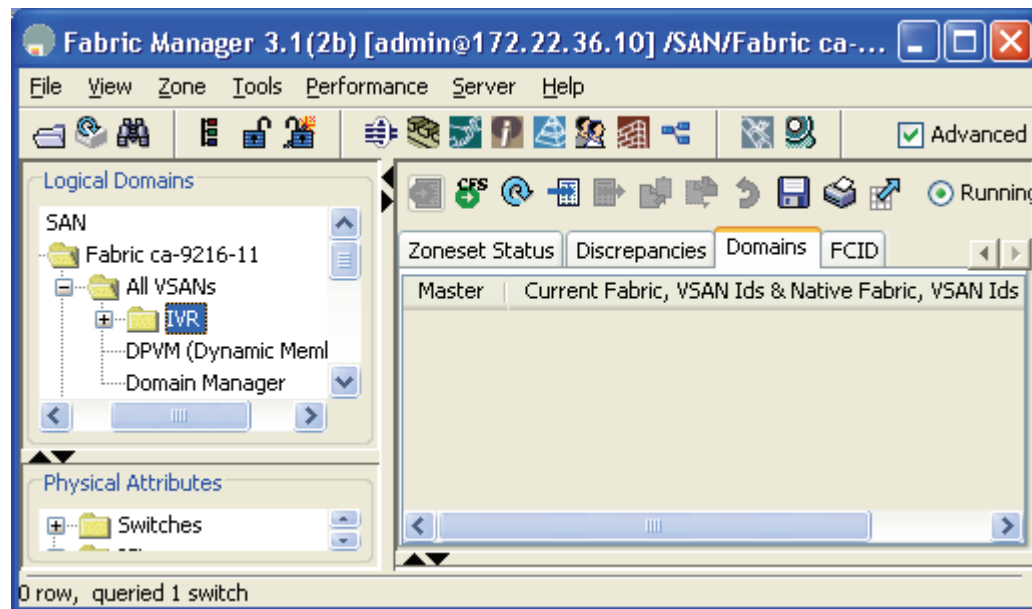
- IVR features enabled: CFS and IVR with NAT.
- Host with pWWN: 10:00:00:00:c9:32:8b:a8 and VSAN 3002.
- Storage with pWWN: 50:06:0e:80:03:4e:95:33, VSAN 3000, Real FC ID: 0xef0002, and FC ID to be configured in Host VSAN: 0x630063.
- IVR topology:

| AFID | SWITCH                  | WWN |     | Active | Cfg. | VSANS     |
|------|-------------------------|-----|-----|--------|------|-----------|
| 1    | 20:00:00:05:30:00:68:5e | *   | yes | yes    | yes  | 3000,3002 |

To configure a storage device to use a specific FC ID in the host VSAN, follow these steps:

- Step 1** In Fabric Manager, in the Logical Domains pane, select your fabric, All VSANs, then IVR. At the far right of the top pane, select the **Domains** tab.

Figure 8-21 IVR Domains Tab



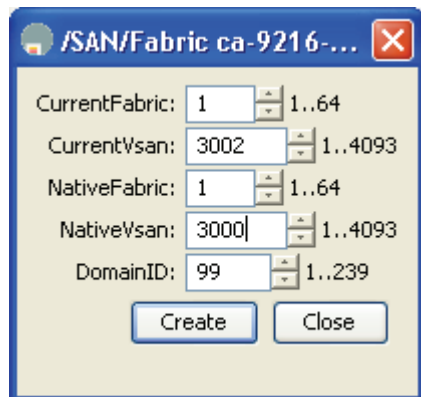
- Step 2** Click **Create Row...**

- Step 3** Enter the following values:

- **Current Fabric number** (AFID where the virtual domain will be created).
- **CurrentVSAN** (where the Virtual Domain will be created).

- **NativeFabric** (AFID where the device actually exists).
- **NativeVSAN** (where the device actually exists).
- **DomainID** (this is the virtual domain\_ID that will be created).

Figure 8-22 Create IVR Domain



**Step 4** Select **Create** and **Close**.

**Step 5** Select the **CFS Commit Pending Changes** button.

At this point, the domain\_ID will be displayed in the **Domains** tab along with the Fabric and VSAN IDs.

To create the IVR Persistent FC ID, follow these steps:

**Step 1** With the IVR folder selected in the Logical Domains pane, select the **FCID** tab.

**Step 2** Click **Create Row...**

**Step 3** Fill in the following information:

- **Current Fabric** (AFID where the virtual device will exist).
- **Current VSAN** (VSAN where the virtual device will exist).
- **PWWN**: (PWWN of virtual device).
- **FCID**: (FC ID for the virtual device). The pull-down arrow automatically inserts the domain\_ID corresponding to the virtual domain configured.

**Step 4** Click **Create** and **Close**.

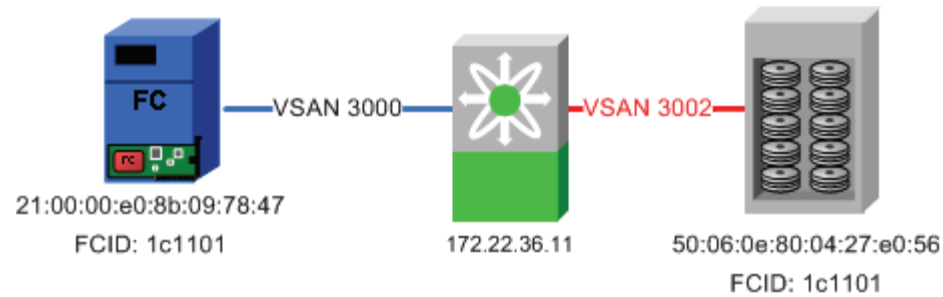
**Step 5** Click **CFS Commit Pending Changes** and **CFS(ivr):Committed** should be displayed in the bottom left corner of Fabric Manager.

At this point the IVR zones and zone sets can be created as per [IVR Zones and Zone Sets, page 8-7](#).

## Configuring a Single Switch with Two VSANs

In this example, a simple two VSAN configuration is done with one switch (see [Figure 8-23](#)). Only IVR-2 with FC NAT works for this example. The topology cannot be done with IVR-1 because the domain IDs are the same between the two VSANs, and the devices themselves have exactly the same FC ID.

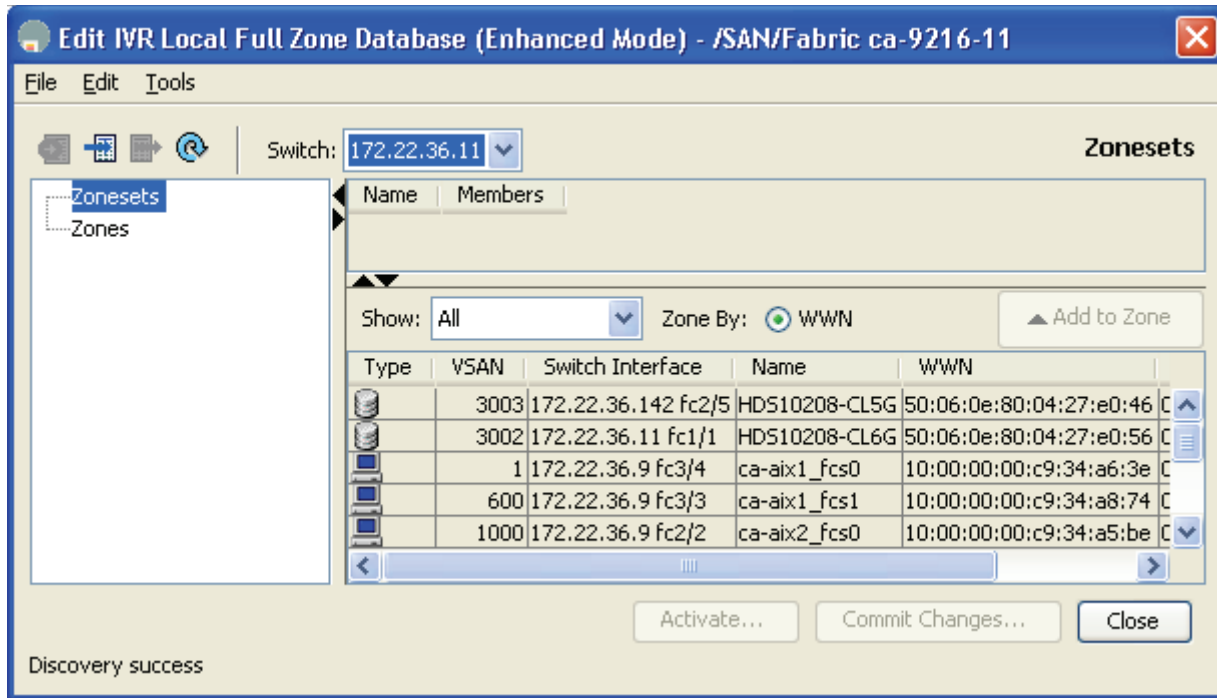
Figure 8-23 IVR-2 Single Switch Example Topology



To configure two VSANs with one switch using Fabric Manager, follow these steps:

- 
- Step 1** Enable IVR-2 with CFS, IVR-NAT and auto topology discovery as described in [Enabling IVR-2 \(FC NAT\)](#), page 8-15.
- Step 2** Create the IVR zones and zone set.
- In the Logical Domains pane, select the fabric and expand **All VSANs**.
  - Right-click **IVR** and choose **Edit Full Local Zone Database**.

Figure 8-24 IVR-2 Create a Zone Set



**Step 3** Create the IVR zone set.

- a. Right-click **Zonesets** in the left pane and choose **Insert**.
- b. Type a name for the zone set and click **OK**.

The resulting zone set appears in the left hand pane.

**Step 4** Create the IVR zone.

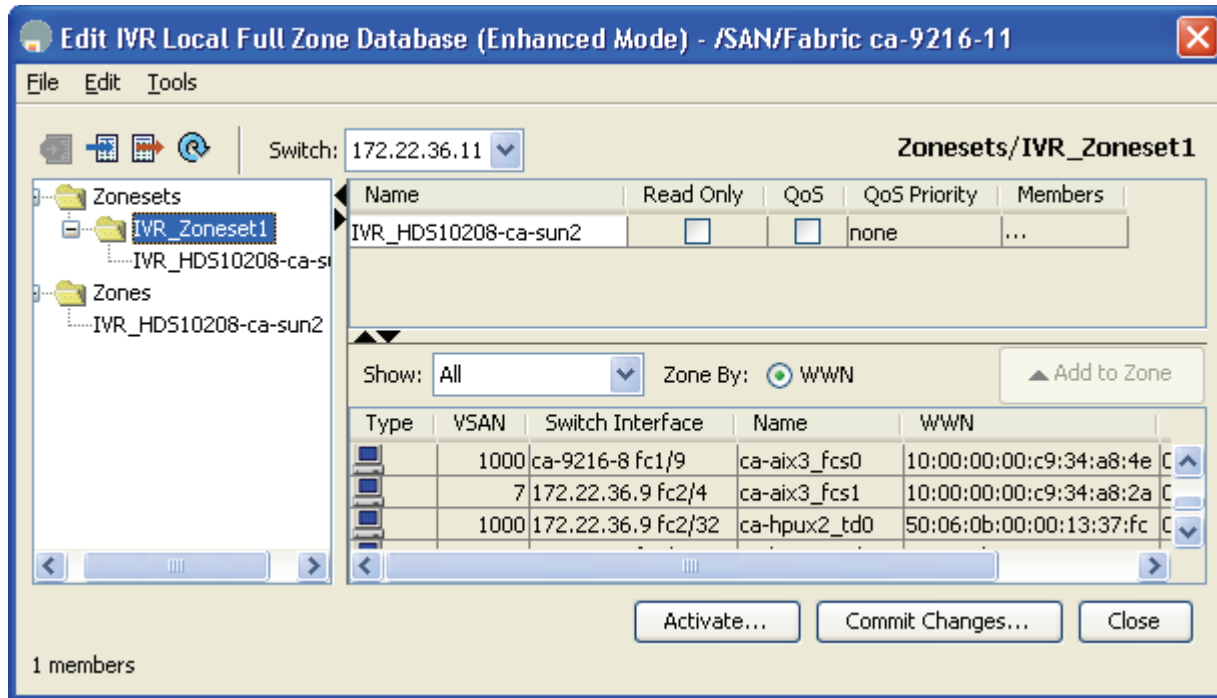
- a. Right-click **Zone** in the left pane and choose **Insert**.
- b. Enter a meaningful zone name and click **OK**.

**Step 5** Add the members to be zoned from the bottom pane into the newly created zone by dragging them into the zone.

**Step 6** In the left pane, drag the zone into the zone set. The pane should now look like [Figure 8-25](#).



Figure 8-25 Single Switch IVR-2, Create Zoneset



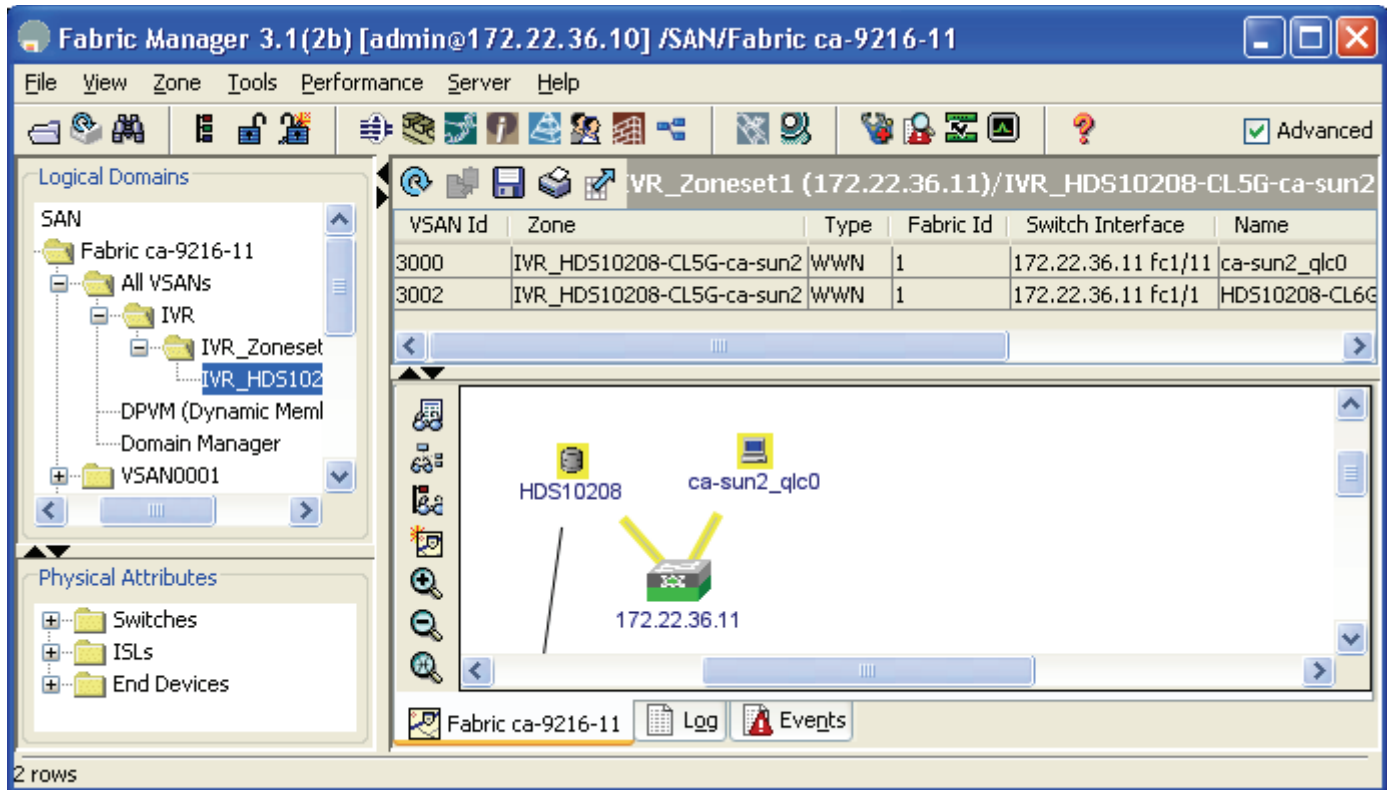
**Step 7** Activate the zone set and implicitly CFS commit it (see [Figure 8-26](#)).

- a. Right-click the zone set in the left pane and choose **Activate**.
- b. Click **Continue Activation**. A pop-up window will display a status log of the activation similar to the following:

```
172.22.36.11:Activating IVR_Zoneset1
172.22.36.11:checking status, elapsed time:0 sec activating IVR_Zoneset1
172.22.36.11:checking status, elapsed time:0 sec activating IVR_Zoneset1
Switch:172.22.36.11
 VSAN id:3000 status :active
 VSAN id:3002 status :active
 committing zone configuration changes
172.22.36.11:172.22.36.11:Commit Successful
172.22.36.11:checking status, elapsed time:3 sec activating IVR_Zoneset1
172.22.36.11:IVR_Zoneset1 Activation success
172.22.36.11:Save running configuration to Startup
172.22.36.11:Save running configuration to Startup on Enhanced 172.22.36.11
172.22.36.11:Saved running configuration to Startup
172.22.36.11:Finished
Success
```

- c. Click **Close** to return to the main Fabric Manager window.

Figure 8-26 Single Switch IVR-2, Display IVR Zoneset

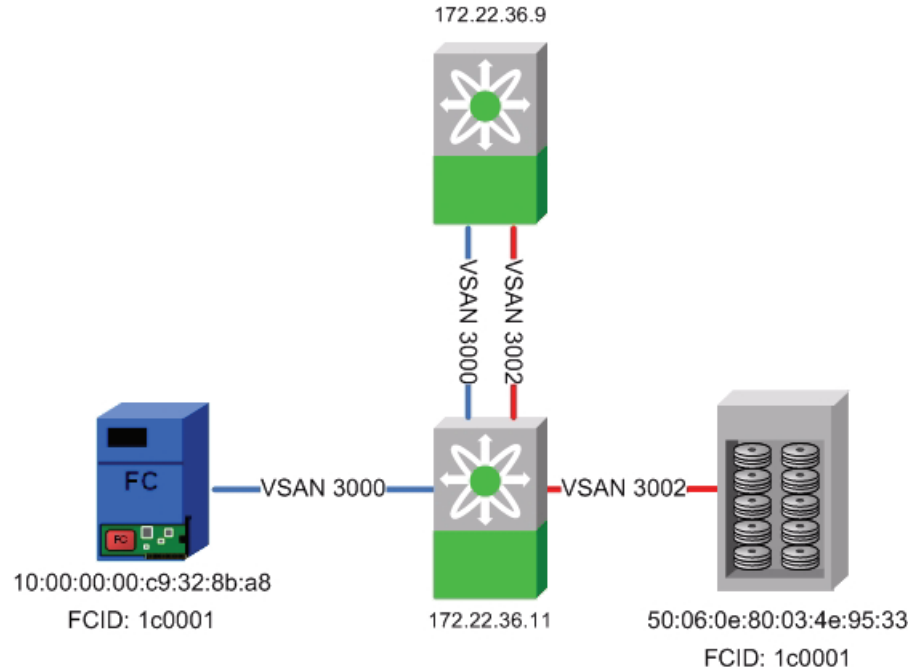


At this point, the host ca-sun1 can access the storage on HDS20017-c20-8.

## Adding a New IVR-Enabled Switch

This recipe adds a new switch to an existing IVR-2 configuration with CFS and auto topology. It builds on the configuration described in [Configuring a Single Switch with Two VSANs](#), page 8-25. The topology is shown in [Figure 8-27](#). The new switch has the IP address 172.22.36.9.

Figure 8-27 Topology for Adding a New IVR-2 Switch



In this configuration, the switch 172.22.36.11 is currently performing IVR-FC NAT between the host in VSAN 3000 and the storage in VSAN 3002. You will add the new switch 172.22.36.9 without impacting the currently running configuration.

**Tip**

When multiple IVR-2 capable switches are configured, one switch can take over the routing functionality of another, provided it can directly see both the source and destination VSANs. With this recipe, if IVR-2 was disabled on 172.22.36.11, switch 172.22.36.9 could automatically take over the routing. This is not possible with IVR-1.

To add a new switch, first enable IVR in Fabric Manager, follow these steps:

- Step 1** In the Logical Domains pane, choose a fabric, expand **All VSANs**, and then select **IVR**.
- Step 2** Click the **Control** tab.
- Step 3** In the Command column, change the 172.22.36.9 switch entry to **enable**.
- Step 4** Click the green **Apply Changes** icon.

The status column entry changes from a yellow progress box, to the word **Success**.

- Step 5** Click the **CFS** tab.
- Step 6** In the Enable Admin column, change the 172.22.36.9 switch entry to **enable**.
- Step 7** Click the green **Apply Changes** icon.

The Last Result column changes from inProgress to **Success**.

- Step 8** Click the Active Topology column to see a topology that includes both switches, even though only the 172.22.36.11 switch is performing any IVR.

**Step 9** Save the configuration of both switches.

---

**Note**

By having CFS already enabled on the first switch (172.22.36.11) and enabling IVR and CFS on the second switch (172.22.36.9), the second switch learned what configuration parameters (FC NAT and auto topology) to enable. There was no need to enable FC NAT and auto topology on the second switch.

---

**Tip**

One of the primary advantages of CFS is the fact that the configuration of the first switch was communicated to the second switch. This is more apparent when the first switch configuration is more complex. In addition, as the topology grows larger and the number of switches, devices and VSANs increase, adding a single switch to an IVR-enabled configuration requires changes to be made to more and more switches, increasing the possibility for human error if done manually.

---



# CHAPTER 9

## FCIP

Fibre Channel over IP (FCIP) is an Internet Engineering Task Force (IETF) standards-based protocol for Fibre Channel SAN extensions over IP networks. The FCIP protocol tunnels Fibre Channel data across an IP network. FCIP protocol extends a Fibre Channel SAN transparently over IP networks, while keeping the Fibre Channel fabric services intact. The FCIP protocol uses the underlying TCP/IP protocol for transport.

FCIP encapsulates the Fibre Channel frame into a TCP packet and transports it across the IP infrastructure. Once a FCIP tunnel is established between the SAN islands over an IP network, the FCIP tunnel works like an Inter-Switch Link (ISL) between the SAN islands.

FCIP on MDS switches supports additional capabilities like compression, read acceleration, write acceleration, tape read acceleration, and tape write acceleration on the FCIP tunnel between the SAN islands. These additional capabilities significantly increase throughput and utilization on an IP network spanning long distances when FCIP is configured.

FCIP can be used for a wide variety of applications, such as these:

- Disaster recovery
- Synchronous data replication
- Asynchronous data replication
- Remote tape backup
- Tape vaulting

On the Cisco MDS platform, FCIP is supported on the MDS 9216A and MDS 9216i switches, and all the 9500 series directors. On the MDS 9216A and the MDS 9500 series director switches, FCIP can be configured by adding an IPS-8 or IP- 4 storage services module, or an MPS-14/2 module in the switches or directors. (See [Figure 9-1](#).)The MDS 9216i comes with two IPS ports built in to the switch.

Figure 9-1 FCIP Capable Modules for MDS 9000 Series Switches

| Module     | Compression | Encryption | IPS Ports | Throughput w/ compression/port |
|------------|-------------|------------|-----------|--------------------------------|
| MPS 14 + 2 | Yes (HW)    | Yes (HW)   | 2         | Line Rate (1Gb/s)              |
| IPS 8      | Yes (SW)    | No         | 8         | 155 Mb/s                       |
| IPS 4      | Yes (SW)    | No         | 4         | 155 Mb/s                       |

The MPS-14/2 module and the IPS-8 and IPS-4 storage services modules support both FCIP and iSCSI on the same port at the same time. The MPS-14/2 module uses hardware to compress and encrypt data. The MPS-14/2 module sustains line rate of 1 Gbps throughput on all the IPS ports while compressing and encrypting data. The IPS-8 and IPS-4 storage services modules compress data using software. The IPS-8 and IPS-4 modules do not support encryption. Because these modules use software compression, they can have a maximum throughput of 155 Mb/s per port when compression is used.

The configuration of the FCIP tunnel requires the configuration of the underlying Gigabit Ethernet interfaces. After the Gigabit Ethernet interfaces are configured, the FCIP tunnel is configured over this TCP/IP link. Each of the IPS ports can support up to three FCIP tunnels on the same port. The IPS-8 module can support up to 24 FCIP tunnels.

## Enabling FCIP

Execute the **FCIP enable** command before attempting to configure FCIP on the switch.



### Caution

If you do not run the **fcip enable** command, you cannot do any further FCIP configuration. This command enables FCIP configuration options in the CLI.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# FCIP enable
mds-switch-1(config)#^Z
mds-switch-1#
```

## Configuring FCIP on a Switch with CLI

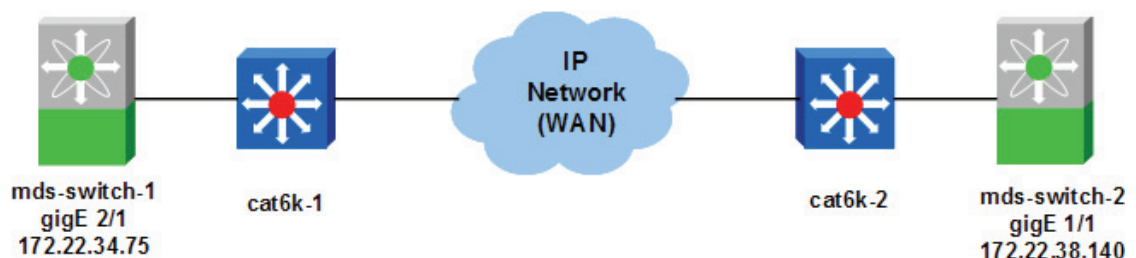
This recipe configures FCIP on a switch using CLI commands. The topology used is shown in [Figure 9-2](#).



### Note

To configure FCIP using Fabric Manager, see [Configuring FCIP Using Fabric Manager](#), page 9-21, but skip the part concerning IPsec.

Figure 9-2 FCIP Topology



The topology shown in [Figure 9-2](#) consists of two MDS 9506 switches. The MDS switch mds-switch-1 has one IPS-4 module and the MDS switch mds-switch-2 has one MPS-14/2 module. Gigabit Ethernet port 2/1 on mds-switch-1 connects to cat6k-1 and Gigabit Ethernet port 1/1 on switch mds-switch-2 connects to cat6k-2. The cat6k-1 and cat6k-2 connect to the wide area network (WAN). The Gigabit

Ethernet port 2/1 on the switch mds-switch-1 has IP address 172.22.34.75 with subnet mask 255.255.254.0 and gateway address 172.22.34.1. The Gigabit Ethernet port 1/1 port on the switch mds-switch-2 has an IP address 172.22.38.140 with a subnet 255.255.254.0 and the gateway address 172.22.38.1. In the recipe, an FCIP tunnel is established between the switch mds-switch-1 (Gigabit Ethernet port 2/1) and mds-switch-2 (Gigabit Ethernet port 1/1).

**Caution**

The IP addresses for the Gigabit Ethernet ports on the IPS-8, IPS-4, and MPS-14/2 modules must be in a different subnet than the management interface. This is a requirement for FCIP to function properly.

The FCIP-related configuration needs to be performed on both the switches.

To configure an FCIP tunnel, follow these steps:

**Step 1** Configure the Gigabit Ethernet interfaces on the MDS switches.

Assign the Gigabit Ethernet interface on the MDS switch mds-switch-1 an IP address and a subnet mask. This allows the Gigabit Ethernet interface to communicate with the network.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface gigabitethernet 2/1
mds-switch-1(config-if)# ip address 172.22.34.75 255.255.254.0
mds-switch-1(config-if)# no shut
mds-switch-1(config-if)# end
mds-switch-1#
```

Assign the Gigabit Ethernet interface on the MDS switch mds-switch-1 an IP address and a subnet mask. This allows the Gigabit Ethernet interface to communicate with the network.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# interface gigabitethernet 1/1
mds-switch-2(config-if)# ip address 172.22.38.140 255.255.254.0
mds-switch-2(config-if)# no shut
mds-switch-2(config-if)# end
mds-switch-2#
```

**Step 2** Configure an IP route so that the two Gigabit Ethernet interfaces can communicate.

An IP route needs to be configured to allow the two Gigabit Ethernet ports on switches mds-switch-1 and mds-switch-2 to communicate with each other. In this recipe, the Gigabit Ethernet ports are in two different subnets, so they must have an explicit route for communication.

**Note**

The recommendation is to create a host route to each of the two Gigabit Ethernet interfaces with a subnet mask of 255.255.255.255. This allows only the two Gigabit Ethernet interfaces to communicate with each other.

For the Gigabit Ethernet port 2/1 on switch mds-switch-1 to communicate with the port Gigabit Ethernet port 1/1 on switch mds-switch-2, create this route configuration on switch mds-switch-1.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# ip route 172.22.38.140 255.255.255.255 172.22.34.1 interface
gigabitethernet 2/1
mds-switch-1(config)# end
mds-switch-1#
```

The preceding configuration provides this information: to reach 172.22.38.140, use the gateway 172.22.34.1 and interface Gigabit Ethernet port 2/1 on switch mds-switch-1.

For the Gigabit Ethernet port 1/1 on switch mds-switch-2 to communicate with Gigabit Ethernet port 2/1 on switch mds-switch-1, create a similar route configuration on switch mds-switch-2.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# ip route 172.22.34.75 255.255.255.255 172.22.38.1 interface
gigabitethernet 1/1
mds-switch-2(config-if)# end
mds-switch-2#
```

The preceding configuration provides this information: to reach 172.22.34.75, use the gateway 172.22.38.1 and interface Gigabit Ethernet port 1/1 on switch mds-switch-2.

**Step 3** Ping the Gigabit Ethernet interfaces to ensure that the Gigabit Ethernet ports can communicate with each other.

From the switch mds-switch-1, ping the IP address of the Gigabit Ethernet interface 1/1 on switch mds-switch-2 using the Gigabit Ethernet interface 2/1. Similarly, ping the IP address of the interface Gigabit Ethernet 2/1 on switch mds-switch-1 from switch mds-switch-2 using the interface Gigabit Ethernet 1/1. Do this from the switch prompt.



#### Note

Before Cisco SAN-OS Release 3.x, you could only ping the IP address of the remote Gigabit Ethernet interface. In Cisco SAN-OS Release 3.x and higher, the interface from which you ping the remote Gigabit Ethernet interface can also be specified.

```
mds-switch-1# ping 172.22.38.140 interface gigabitethernet 2/1
PING 172.22.38.140 (172.22.38.140) from 172.22.34.75 gige2-1: 56(84) bytes of data.
64 bytes from 172.22.38.140: icmp_seq=1 ttl=254 time=0.573 ms
64 bytes from 172.22.38.140: icmp_seq=2 ttl=254 time=0.516 ms
64 bytes from 172.22.38.140: icmp_seq=3 ttl=254 time=0.482 ms
64 bytes from 172.22.38.140: icmp_seq=4 ttl=254 time=0.511 ms
64 bytes from 172.22.38.140: icmp_seq=5 ttl=254 time=0.492 ms

--- 172.22.38.140 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4014ms
rtt min/avg/max/mdev = 0.482/0.514/0.573/0.042 ms
mds-switch-1#
```

```
mds-switch-2# ping 172.22.34.75 interface gigabitethernet 1/1
PING 172.22.34.75 (172.22.34.75) from 172.22.38.140 gige1-1: 56(84) bytes of data.
64 bytes from 172.22.34.75: icmp_seq=1 ttl=254 time=0.593 ms
64 bytes from 172.22.34.75: icmp_seq=2 ttl=254 time=0.507 ms
64 bytes from 172.22.34.75: icmp_seq=3 ttl=254 time=0.509 ms
64 bytes from 172.22.34.75: icmp_seq=4 ttl=254 time=0.529 ms
64 bytes from 172.22.34.75: icmp_seq=5 ttl=254 time=0.527 ms

--- 172.22.34.75 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4041ms
rtt min/avg/max/mdev = 0.507/0.533/0.593/0.031 ms
mds-switch-2#
```



#### Note

It is critical to check the connectivity between the two Gigabit Ethernet ports on the IPS and MPS modules on both the switches before proceeding further. A ping test is sufficient to check connectivity.

**Step 4** Measure the round trip time (RTT) between the two Gigabit Ethernet interfaces. The RTT value is required for configuring the FCIP profile in the next step.



```

mds-switch-1# ips measure-rtt 172.22.38.140 interface gigabitethernet 2/1
Round trip time is 425 micro seconds (0.42 milli seconds)
mds-switch-1#

mds-switch-2# ips measure-rtt 172.22.34.75 interface gigabitethernet 1/1
Round trip time is 425 micro seconds (0.42 milli seconds)
mds-switch-2#

```

**Note**

FCIP by default uses TCP port 3225. If there is a firewall between the two switches that need to be connected through FCIP, then port 3225 needs to be opened up in the firewall for the FCIP tunnel to come up.

**Step 5** Configure FCIP profiles on both switches.

An FCIP profile must be created because the profile defines the characteristics for the FCIP tunnel. The round trip time measured in the previous step is needed for profile configuration. In this case, the time was 425 microseconds.

The IP address used in the profile configuration is the IP address assigned to the Gigabit Ethernet interface on the associated switch.

**Note**

We recommend using the same profile numbers (unique to a tunnel) for the FCIP profiles configured on both switches for a given FCIP tunnel.

```

mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# fcip profile 1
mds-switch-1(config-profile)# ip address 172.22.34.75
mds-switch-1(config-profile)# tcp max-bandwidth-mbps 1000 min-available-bandwidth-mbps
500 round-trip-time-us 425
mds-switch-1(config-profile)# end
mds-switch-1#

mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# fcip profile 1
mds-switch-2(config-profile)# ip address 172.22.38.140
mds-switch-2(config-profile)# tcp max-bandwidth-mbps 1000 min-available-bandwidth-mbps
500 round-trip-time-us 425
mds-switch-2(config-profile)# end
mds-switch-2#

```

The FCIP profile 1 has now been defined on switches mds-switch-1 and mds-switch-2.

**Step 6** Configure the FCIP interface on both switches.

In the FCIP interface configuration, the profile to be used and the peer information (remote Gigabit Ethernet port's IP address) are specified. Additionally, compression and write acceleration can be configured on the FCIP interface.

```

mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface fcip 1
mds-switch-1(config-if)# peer-info ipaddr 172.22.38.140
mds-switch-1(config-if)# use-profile 1
mds-switch-1(config-if)# no shutdown
mds-switch-1(config-if)# end
mds-switch-1#

```

```

mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# int fcip 1
mds-switch-2(config-if)# use-profile 1
mds-switch-2(config-if)# peer-info ipaddr 172.22.34.75
mds-switch-2(config-if)# no shut
mds-switch-2(config-if)# end
mds-switch-2#

```

The FCIP tunnel should be up and running. Use the **show fcip summary** command to see the status of the FCIP link between the two switches.

```

mds-switch-1# show fcip summary

Tun prof Eth-if peer-ip Status T W T Enc Comp Bandwidth rtt
 max/min (us)

1 1 GE2/1 172.22.38.140 TRNK Y N N N N 1000M/500M 425

mds-switch-1#

mds-switch-2# show fcip summary

Tun prof Eth-if peer-ip Status T W T Enc Comp Bandwidth rtt
 max/min (us)

1 1 GE1/1 172.22.34.75 TRNK Y N N N N 1000M/500M 425

mds-switch-2#

```

## Enabling FCIP Write Acceleration

The IPS-8 and IPS-4 modules and the MPS-14/2 module support write acceleration to help alleviate the effects of network latency. Write acceleration helps significantly in improving the write performance of the application over a wide area network (WAN). Write acceleration helps in maximizing the WAN throughput by reducing the effects of WAN latency on the writes that need to go across the WAN.



### Note

Write Acceleration can work with PortChannels only when the PortChannel is managed by PortChannel protocol (PCP). This is available on all Cisco SAN-OS Release 2.0(1b) and above.

Enable write acceleration on the interface FCIP 1 on switch mds-switch-1.

```

mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface fcip 1
mds-switch-1(config)# write-accelerator
mds-switch-1(config)# end
mds-switch-1#

```

Enable write acceleration on the interface FCIP 1 on switch mds-switch-2.

```

mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# interface fcip 1
mds-switch-2(config)# write-accelerator
mds-switch-2(config)# end
mds-switch-2#

```

Use the **show fcip summary** command to see the status of the FCIP link between the two switches with write acceleration turned on.

```
mds-switch-1# show fcip summary
```

| Tun | prof | Eth-if | peer-ip       | Status | T<br>E | W<br>A | T<br>A | Enc | Comp | Bandwidth<br>max/min | rtt<br>(us) |
|-----|------|--------|---------------|--------|--------|--------|--------|-----|------|----------------------|-------------|
| 1   | 1    | GE2/1  | 172.22.38.140 | TRNK   | Y      | Y      | N      | N   | N    | 1000M/500M           | 425         |

```
mds-switch-1#
```

```
mds-switch-2# show fcip summary
```

| Tun | prof | Eth-if | peer-ip      | Status | T<br>E | W<br>A | T<br>A | Enc | Comp | Bandwidth<br>max/min | rtt<br>(us) |
|-----|------|--------|--------------|--------|--------|--------|--------|-----|------|----------------------|-------------|
| 1   | 1    | GE1/1  | 172.22.34.75 | TRNK   | Y      | Y      | N      | N   | N    | 1000M/500M           | 425         |

```
mds-switch-2#
```



#### Note

Enable write acceleration on both sides of the FCIP link. If it is enabled on only one side, the FCIP link will not come up.

## Enabling FCIP Compression

The IPS-8 and IPS-4 modules support software-based compression. The MPS-14/2 module supports hardware-based compression. The IPS-8 and IPS-4 modules do software-based compression at the rate of 155 Mb/s per port, while the MPS-14/2 module's two IPS ports do hardware compression at the line rate of 1 Gbps per port.

In Cisco SAN-OS, there four IP compression modes, which are listed below:

- **Auto** (default) mode picks the appropriate compression scheme based on the bandwidth of the link. (The bandwidth of the link is configured in the FCIP profile's TCP parameters.)
- **Mode1** is a fast compression mode for high-bandwidth links (> 25 Mb/s).
- **Mode2** is a moderate compression mode for moderately low bandwidth links (between 10 and 25 Mbps).
- **Mode3** is a high-compression mode for low bandwidth links (< 10 Mb/s).



#### Caution

A link cannot come up if its compression mode is configured incorrectly. Enable compression on both sides of a link. Also, make sure the compression modes on both sides match.



#### Note

Use the chart in [Figure 9-3](#) as a guide to determine which compression mode is most appropriate for your link. The numbers were derived using the Canterbury Corpus test suite. Your actual performance number may vary depending on device type and data pattern.

The recipe below shows how compression is enabled on the FCIP tunnel.

Enable compression on the interface fcip 1 on switch mds-switch-1. The mode of compression used is model.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface fcip 1
mds-switch-1(config-if)# ip-compression model
mds-switch-1(config-if)#end
mds-switch-1#
```

Enable compression on the interface fcip 1 on switch mds-switch-2. The mode of compression used is model.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# interface fcip 1
mds-switch-2(config-if)# ip-compression model
mds-switch-2(config-if)# end
mds-switch-2#
```

Use the **show fcip summary** command to see the status of the FCIP link between the two switches after compression is enabled.

```
mds-switch-1# show fcip summary
```

| Tun | prof | Eth-if | peer-ip       | Status | T | W | T | Enc | Comp | Bandwidth  | rtt  |
|-----|------|--------|---------------|--------|---|---|---|-----|------|------------|------|
|     |      |        |               |        | E | A | A |     |      | max/min    | (us) |
| 1   | 1    | GE2/1  | 172.22.38.140 | TRNK   | Y | Y | N | N   | M1   | 1000M/500M | 425  |

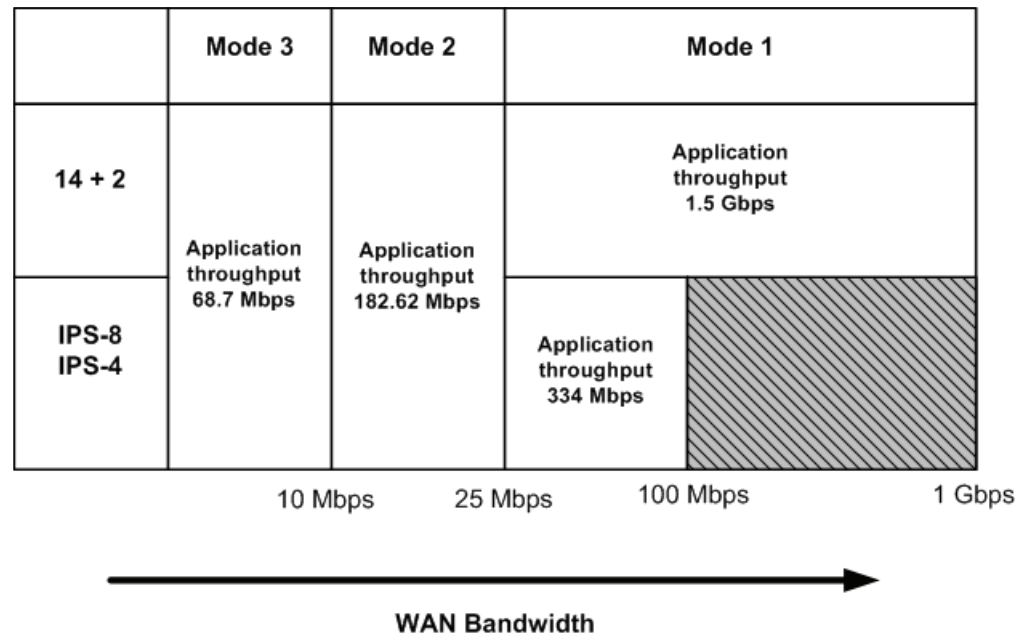
```
mds-switch-1#
```

```
mds-switch-2# show fcip summary
```

| Tun | prof | Eth-if | peer-ip      | Status | T | W | T | Enc | Comp | Bandwidth  | rtt  |
|-----|------|--------|--------------|--------|---|---|---|-----|------|------------|------|
|     |      |        |              |        | E | A | A |     |      | max/min    | (us) |
| 1   | 1    | GE1/1  | 172.22.34.75 | TRNK   | Y | Y | N | N   | M1   | 1000M/500M | 425  |

```
mds-switch-2#
```

Figure 9-3 Approximate Application Throughput with MDS SAN-OS with Compression



## Enabling Tape Acceleration

The tape acceleration feature is similar to write acceleration. Tape acceleration alleviates latency associated issues with backing up to tape drives through FCIP tunnels over long-distance WAN links.



### Caution

FCIP tape acceleration does not work if an FCIP port is part of a PortChannel. It also does not work if there are multiple paths with equal costs between the backup host and the tape device.



### Tip

We recommend that you have multiple paths between the backup server and the tape device, and that you configure the multiple paths with varying costs (such as the FSPF cost of the FCIP link). This ensures that only one link at a time is used for tape acceleration and provides failover if the current link fails.

## Enabling Tape Acceleration from the CLI

Tape acceleration, like write acceleration, needs to be enabled on both sides of the FCIP tunnel. Tape acceleration is a subcommand of write acceleration.

Enable tape acceleration on switch mds-switch-1 for the interface fcip1.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# int fcip1
mds-switch-1(config-if)# write-accelerator tape-accelerator
mds-switch-1(config-if)#end
mds-switch-1#
```

Enable tape acceleration on switch mds-switch-2 for the interface fcip1.

```

mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# int fcipl
mds-switch-2(config-if)# write-accelerator tape-accelerator
mds-switch-2(config-if)#end
mds-switch-2#

```

Use the **show fcip summary** command to see the status of the FCIP link between the two switches after tape acceleration is enabled.

```
mds-switch-1# show fcip summary
```

```

Tun prof Eth-if peer-ip Status T W T Enc Comp Bandwidth rtt
 max/min (us)

1 1 GE2/1 172.22.38.140 TRNK Y Y Y N M1 1000M/500M 425

```

```
mds-switch-1#
```

```
mds-switch-2# show fcip summary
```

```

Tun prof Eth-if peer-ip Status T W T Enc Comp Bandwidth rtt
 max/min (us)

1 1 GE1/1 172.22.34.75 TRNK Y Y Y N M1 1000M/500M 425

```

```
mds-switch-2#
```

Use the **show interface FCIP** command to see the status of the link and the configuration of the FCIP tunnel.

```

mds-switch-1# show interface fcip 1
fcipl is trunking
 Hardware is GigabitEthernet
 Port WWN is 20:42:00:0d:ec:24:5e:c0
 Peer port WWN is 20:10:00:0d:ec:24:5e:80
 Admin port mode is auto, trunk mode is on
 snmp traps are enabled
 Port mode is TE
 Port vsan is 1
 Speed is 1 Gbps
 Trunk vsans (admin allowed and active) (1)
 Trunk vsans (up) (1)
 Trunk vsans (isolated) ()
 Trunk vsans (initializing) ()
 Using Profile id 1 (interface GigabitEthernet2/1)
 Peer Information
 Peer Internet address is 172.22.38.140 and port is 3225
 Write acceleration mode is configured on; operationally on
 Tape acceleration mode is configured on; operationally on
 Tape read acceleration mode is operationally on
 Tape Accelerator flow control buffer size is automatic
 Ficon Tape acceleration configured off for all vsans
 IP Compression is enabled and set for model
 Special Frame is disabled
 Maximum number of TCP connections is 2
 Time Stamp is disabled
 QOS control code point is 0
 QOS data code point is 0
 B-port mode disabled
 TCP Connection Information
 2 Active TCP connections

```

```

Control connection: Local 172.22.34.75:3225, Remote 172.22.38.140:65428
Data connection: Local 172.22.34.75:3225, Remote 172.22.38.140:65430
44 Attempts for active connections, 44 close of connections
TCP Parameters
Path MTU 1500 bytes
Current retransmission timeout is 200 ms
Round trip time: Smoothed 10 ms, Variance: 5 Jitter: 150 us
Advertized window: Current: 55 KB, Maximum: 55 KB, Scale: 4
Peer receive window: Current: 49 KB, Maximum: 49 KB, Scale: 4
Congestion window: Current: 21 KB, Slow start threshold: 50 KB
Current Send Buffer Size: 4151 KB, Requested Send Buffer Size: 0 KB
CWM Burst Size: 50 KB
5 minutes input rate 688 bits/sec, 86 bytes/sec, 0 frames/sec
5 minutes output rate 664 bits/sec, 83 bytes/sec, 0 frames/sec
5853 frames input, 680332 bytes
 5853 Class F frames input, 680332 bytes
 0 Class 2/3 frames input, 0 bytes
 0 Reass frames
 0 Error frames timestamp error 0
5874 frames output, 582104 bytes
 5874 Class F frames output, 582104 bytes
 0 Class 2/3 frames output, 0 bytes
 0 Error frames

mds-switch-1#

```

In this output from mds-switch-1, the link properties are in bold.

```

mds-switch-2# sh int fcip 1
fcip1 is trunking
Hardware is GigabitEthernet
Port WWN is 20:10:00:0d:ec:24:5e:80
Peer port WWN is 20:42:00:0d:ec:24:5e:c0
Admin port mode is auto, trunk mode is on
snmp traps are enabled
Port mode is TE
Port vsan is 1
Speed is 1 Gbps
Trunk vsans (admin allowed and active) (1)
Trunk vsans (up) (1)
Trunk vsans (isolated) ()
Trunk vsans (initializing) ()
Using Profile id 1 (interface GigabitEthernet1/1)
Peer Information
Peer Internet address is 172.22.34.75 and port is 3225
Write acceleration mode is configured on; operationally on
Tape acceleration mode is configured on; operationally on
Tape read acceleration mode is operationally on
Tape Accelerator flow control buffer size is automatic
Ficon Tape acceleration configured off for all vsans
IP Compression is enabled and set for model
Special Frame is disabled
Maximum number of TCP connections is 2
Time Stamp is disabled
QOS control code point is 0
QOS data code point is 0
B-port mode disabled
TCP Connection Information
2 Active TCP connections
Control connection: Local 172.22.38.140:65428, Remote 172.22.34.75:3225
Data connection: Local 172.22.38.140:65430, Remote 172.22.34.75:3225
54 Attempts for active connections, 29 close of connections

```

```

TCP Parameters
 Path MTU 1500 bytes
 Current retransmission timeout is 200 ms
 Round trip time: Smoothed 0 ms, Variance: 0 Jitter: 150 us
 Advertized window: Current: 49 KB, Maximum: 49 KB, Scale: 4
 Peer receive window: Current: 55 KB, Maximum: 55 KB, Scale: 4
 Congestion window: Current: 18 KB, Slow start threshold: 50 KB
 Current Send Buffer Size: 49 KB, Requested Send Buffer Size: 0 KB
 CWM Burst Size: 50 KB
 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
 5775 frames input, 570892 bytes
 5775 Class F frames input, 570892 bytes
 0 Class 2/3 frames input, 0 bytes
 0 Reass frames
 0 Error frames timestamp error 0
 5788 frames output, 673640 bytes
 5788 Class F frames output, 673640 bytes
 0 Class 2/3 frames output, 0 bytes
 0 Error frames

mds-switch-2#

```

In this output from mds-switch-2 the link properties are in bold.

## Tuning FCIP

Configuring and bringing up an FCIP tunnel establishes an ISL between two switches. To achieve greater efficiency and utilization from the link, optimize link parameters. This optimization is specific to each FCIP link between switches. Optimization for a FCIP link over a slow 1.54 Mb/s connection will be different than that done for an FCIP link over a 1Gb/s connection with very low latency. This section provides insight into optimization of FCIP links.



### Note

Individual results may vary due to network conditions (existing link utilization and quality) as well as the storage array and host type using the FCIP tunnel.

## TCP Tuning: Latency and Available Bandwidth

The latency of an FCIP link is the amount of time it takes a packet to go from one end of the link to the other. Latency is affected by many factors including distance and the number of devices that it must traverse. Even the fastest routers and switches incur some latency.

Even though latency cannot be eliminated, protocols can be tuned and the features of an MDS switch (such as FCIP write acceleration) can be enabled to minimize its effect. These features are enabled in the FCIP profile.



### Tip

If the underlying link is dedicated to FCIP, the minimum and maximum available bandwidth values should be the same.

Available bandwidth is the amount of bandwidth that the FCIP link can use on the network. You define a maximum and a minimum value for the FCIP link in the FCIP profile.



- The maximum available bandwidth value is the maximum amount of bandwidth that the FCIP link can use on the network.
- The minimum available bandwidth value is used as a guideline for the minimum value. If there are serious problems on the network (dropped packets, congestion), the link goes slower than the minimum value. We recommend that the minimum value be set (at least) to the minimum accepted by applications (EMC SRDF, IBM PPRC, Hitachi True copy, and so on).

Table 9-1 contains some common WAN links and their speeds. These circuits are most often used as the underlying network for a FCIP link. For example, the underlying network may be an OC3, but you may only be able to use 100 Mb of that link.

Table 9-1 Common WAN Circuit Speeds

| Circuit Name | Link Speed  |
|--------------|-------------|
| T1           | 1.544 Mb/s  |
| T3           | 44.736 Mb/s |
| OC3          | 155 Mb/s    |
| OC12         | 622 Mb/s    |
| OC24         | 1.244 Gb/s  |
| OC48         | 2.488 Gb/s  |
| OC192        | 10 Gb/s     |
| OC 768       | 40 Gb/s     |

**Tip**

When deploying FCIP, you should always involve the LAN and WAN teams to find out about the connections they are providing you. If there are performance issues, they can often help you troubleshoot them from the network standpoint. Involve them earlier rather than later.

## Configuring Multiple FCIP Tunnels Using a Single Gigabit Ethernet Port

This recipe describes the method to create multiple FCIP tunnels using a single Gigabit Ethernet interface on a IPS or MPS module, meaning that a single Gigabit Ethernet port on a MDS switch is used to create two separate tunnels to two different Gigabit Ethernet ports on two remote MDS switches. This is accomplished in a couple of ways.

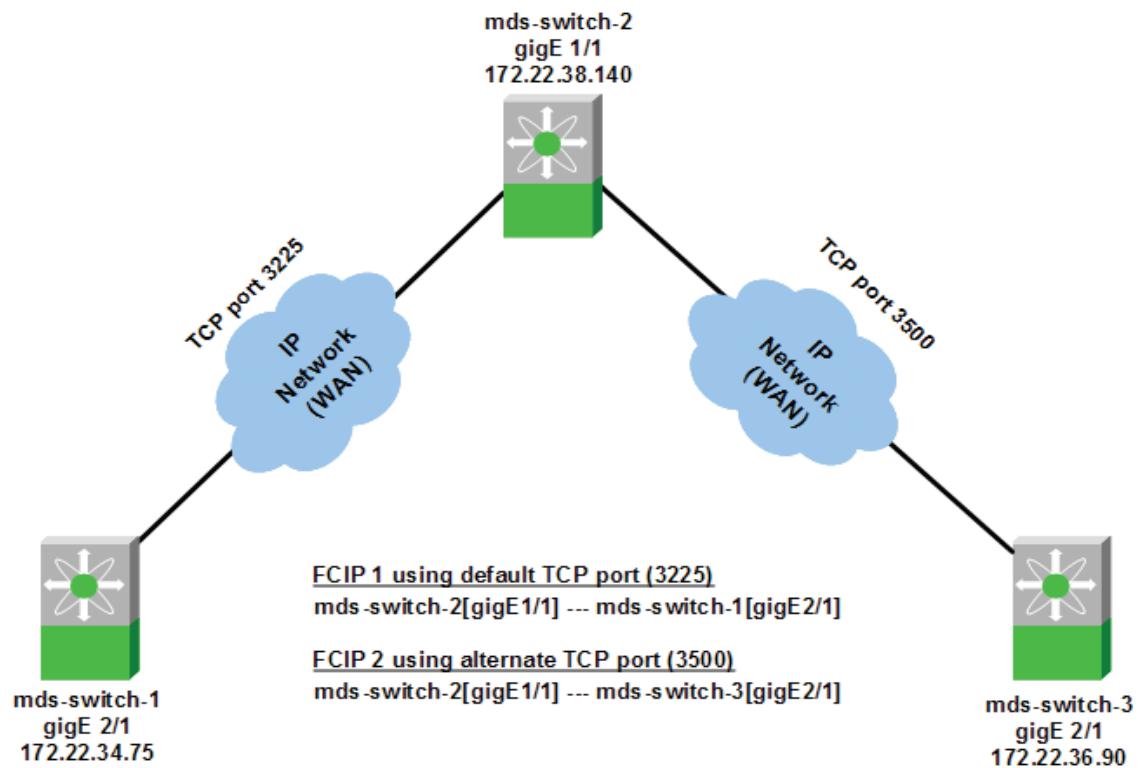
1. Using different TCP ports to establish the separate FCIP tunnels to the local switch to the two remote switches.
2. Using 802.1q interfaces to create separate tunnels from the local switch to the two remote switches.

**Note**

Each Gigabit Ethernet port on the IPS-8, IPS-4 and MPS-14/2 module can support a maximum of three FCIP tunnels.

This recipe describes a method to create two FCIP tunnels using two different TCP ports. On the switch mds-switch-2, the Gigabit Ethernet port 1/1 connects to the Gigabit Ethernet port 2/1 on the switch mds-switch-1 using the default FCIP TCP port 3225. On the switch mds-switch-2, the Gigabit Ethernet port 1/1 connects to the Gigabit Ethernet port 2/1 on mds-switch-3 using the alternate TCP port 3500. This topology is diagrammed in Figure 9-4.

Figure 9-4 FCIP Three-way Topology



The first FCIP tunnel is configured between Gigabit Ethernet port 1/1 on switch mds-switch-2 and the Gigabit Ethernet port 2/1 on switch mds-switch-1. This FCIP connection uses the default FCIP TCP port 3225. The second FCIP tunnel between Gigabit Ethernet port 1/1 on switch mds-switch-2 and Gigabit Ethernet port 2/1 on switch mds-switch-3 is configured to use an alternate TCP port 3500.



#### Note

The downside to this configuration is that the bandwidth of the Gigabit Ethernet port 1/1 on the switch mds-switch-2 has to be shared between the 2 tunnels that will be set up to switches mds-switch-1 and mds-switch-3.

The configuration plan for the three MDS switches is as follows:

- Configure FCIP tunnel fcip1 between mds-switch-2 and mds-switch-1 using the default FCIP port.
- Configure FCIP tunnel fcip2 between mds-switch-2 and mds-switch-3 using TCP port 3500.

To configure the FCIP tunnel, follow these steps:

**Step 1** Configure the Gigabit Ethernet interface on the MDS switches mds-switch-2 and mds-switch-1.

Assign the Gigabit Ethernet interface on the MDS switch mds-switch-1 an IP address and a subnet mask. This allows the Gigabit Ethernet interface to communicate with the network.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface gigabitethernet 2/1
mds-switch-1(config-if)# ip address 172.22.34.75 255.255.254.0
mds-switch-1(config-if)# no shut
mds-switch-1(config-if)# end
mds-switch-1#
```

Assign the Gigabit Ethernet interface on the MDS switch mds-switch-1 an IP address and a subnet mask. This allows the Gigabit Ethernet interface to communicate with the network.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# interface gigabitethernet 1/1
mds-switch-2(config-if)# ip address 172.22.38.140 255.255.254.0
mds-switch-2(config-if)# no shut
mds-switch-2(config-if)# end
mds-switch-1#
```

**Step 2** Configure an IP route so that the two Gigabit Ethernet interfaces can communicate.

An IP route needs to be configured to allow the two Gigabit Ethernet ports on switches mds-switch-1 and mds-switch-2 to communicate with each other. In this recipe, the Gigabit Ethernet ports are in two different subnets, so they must have an explicit route for communication.



**Note**

The recommendation is to create a host route to each of the two Gigabit Ethernet interfaces with a subnet mask of 255.255.255.255. This allows only the two Gigabit Ethernet interfaces to communicate with each other.

For the Gigabit Ethernet port 2/1 on switch mds-switch-1 to communicate with the port Gigabit Ethernet 1/1 on switch mds-switch-2, create this route configuration on switch mds-switch-1.

```
mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# ip route 172.22.38.140 255.255.255.255 172.22.34.1 interface
gigabitethernet 2/1
mds-switch-1(config)# end
mds-switch-1#
```

The configuration provides this information: To reach 172.22.38.140, use the gateway 172.22.34.1 and interface Gigabit Ethernet 2/1 on switch mds-switch-1.

For the Gigabit Ethernet port 1/1 on switch mds-switch-2 to communicate with Gigabit Ethernet port 2/1 on switch mds-switch-1, create a similar route configuration on switch mds-switch-2.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# ip route 172.22.34.75 255.255.255.255 172.22.38.1 interface
gigabitethernet 1/1
mds-switch-2(config-if)# end
mds-switch-2#
```

The configuration above provides this information: to reach 172.22.34.75 use the gateway 172.22.38.1 and interface Gigabit Ethernet 1/1 on switch mds-switch-2.

**Step 3** Ping the Gigabit Ethernet interfaces to ensure that the Gigabit Ethernet ports can communicate with each other.

From the switch mds-switch-1, ping the IP address of the Gigabit Ethernet interface 1/1 on switch mds-switch-2 using the Gigabit Ethernet interface 2/1. Similarly, ping the IP address of the interface Gigabit Ethernet 2/1 on switch mds-switch-1 from switch mds-switch-2 using the interface Gigabit Ethernet 1/1. Do this from the switch prompt.

**Note**

Before Cisco SAN-OS Release 3.x, you could only ping the IP address of the remote Gigabit Ethernet interface. In Cisco SAN-OS Release 3.x and higher, the interface from which you ping the remote Gigabit Ethernet interface can also be specified.

```
mds-switch-1# ping 172.22.38.140 interface gigabitethernet 2/1
PING 172.22.38.140 (172.22.38.140) from 172.22.34.75 gige2-1: 56(84) bytes of data.
64 bytes from 172.22.38.140: icmp_seq=1 ttl=254 time=0.573 ms
64 bytes from 172.22.38.140: icmp_seq=2 ttl=254 time=0.516 ms
64 bytes from 172.22.38.140: icmp_seq=3 ttl=254 time=0.482 ms
64 bytes from 172.22.38.140: icmp_seq=4 ttl=254 time=0.511 ms
64 bytes from 172.22.38.140: icmp_seq=5 ttl=254 time=0.492 ms

--- 172.22.38.140 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4014ms
rtt min/avg/max/mdev = 0.482/0.514/0.573/0.042 ms
mds-switch-1#

mds-switch-2# ping 172.22.34.75 interface gigabitethernet 1/1
PING 172.22.34.75 (172.22.34.75) from 172.22.38.140 gige1-1: 56(84) bytes of data.
64 bytes from 172.22.34.75: icmp_seq=1 ttl=254 time=0.593 ms
64 bytes from 172.22.34.75: icmp_seq=2 ttl=254 time=0.507 ms
64 bytes from 172.22.34.75: icmp_seq=3 ttl=254 time=0.509 ms
64 bytes from 172.22.34.75: icmp_seq=4 ttl=254 time=0.529 ms
64 bytes from 172.22.34.75: icmp_seq=5 ttl=254 time=0.527 ms

--- 172.22.34.75 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4041ms
rtt min/avg/max/mdev = 0.507/0.533/0.593/0.031 ms
mds-switch-2#
```

**Note**

It is critical to check the connectivity between the two Gigabit Ethernet ports on the IPS and MPS modules on both the switches before proceeding further. A ping test is sufficient to check connectivity.

- Step 4** Measure the round trip time (RTT) between the two Gigabit Ethernet interfaces. The RTT value is required for configuring the FCIP profile in the next step.

```
mds-switch-1# ips measure-rtt 172.22.38.140 interface gigabitethernet 2/1
Round trip time is 425 micro seconds (0.42 milli seconds)
mds-switch-1#

mds-switch-2# ips measure-rtt 172.22.34.75 interface gigabitethernet 1/1
Round trip time is 425 micro seconds (0.42 milli seconds)
mds-switch-2#
```

**Note**

FCIP by default uses TCP port 3225. If there is a firewall between the two switches that need to be connected through FCIP, then the port 3225 needs to be opened up in the firewall for FCIP tunnel to come up.

- Step 5** Configure FCIP profiles on both switches.

An FCIP profile must be created. The profile defines the characteristics for the FCIP tunnel. The RTT measured in the previous step is needed for profile configuration. In this case, the time was 425 micro seconds.

The IP address used in the profile configuration is the IP address assigned to the Gigabit Ethernet interface on the associated switch.

**Note**

We recommend using the same profile numbers (unique to a tunnel) for the FCIP profiles configured on both switches for a given FCIP tunnel.

```

mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# fcip profile 1
mds-switch-1(config-profile)# ip address 172.22.34.75
mds-switch-1(config-profile)# tcp max-bandwidth-mbps 1000 min-available-bandwidth-mbps
500 round-trip-time-us 425
mds-switch-1(config-profile)# end
mds-switch-1#

mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# fcip profile 1
mds-switch-2(config-profile)# ip address 172.22.38.140
mds-switch-2(config-profile)# tcp max-bandwidth-mbps 1000 min-available-bandwidth-mbps
500 round-trip-time-us 425
mds-switch-2(config-profile)# end
mds-switch-2#

```

The FCIP profile 1 has now been defined on switches mds-switch-1 and mds-switch-2.

**Step 6** Configure the FCIP interface on both switches.

In the FCIP interface configuration, the profile to be used and the peer information (remote Gigabit Ethernet IP address) are specified. Additionally, compression and write acceleration can be configured on the FCIP interface.

```

mds-switch-1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-1(config)# interface fcip 1
mds-switch-1(config-if)# peer-info ipaddr 172.22.38.140
mds-switch-1(config-if)# use-profile 1
mds-switch-1(config-if)# no shutdown
mds-switch-1(config-if)# end
mds-switch-1#

mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# int fcip 1
mds-switch-2(config-if)# use-profile 1
mds-switch-2(config-if)# peer-info ipaddr 172.22.34.75
mds-switch-2(config-if)# no shut
mds-switch-2(config-if)# end
mds-switch-2#

```

The FCIP tunnel should be up and running. Use the **show fcip summary** command to see the status of the FCIP link between the two switches.

```

mds-switch-1# show fcip summary

Tun prof Eth-if peer-ip Status T W T Enc Comp Bandwidth rtt
 E A A max/min (us)

```

```
1 1 GE2/1 172.22.38.140 TRNK Y N N N N 1000M/500M 425
```

```
mds-switch-1#
```

```
mds-switch-2# show fcip summary
```

```

Tun prof Eth-if peer-ip Status T W T Enc Comp Bandwidth rtt
 E A A max/min (us)

1 1 GE1/1 172.22.34.75 TRNK Y N N N N 1000M/500M 425
```

```
mds-switch-2#
```

The first FCIP tunnel fcip1 between the switches mds-switch-2 and mds-switch-1 using the standard TCP port 3235 is up and running.

The configuration steps for the tunnel fcip2 between switches mds-switch-2 and mds-switch-3 using TCP port 3500 follows.

#### Step 7 Configure the Gigabit Ethernet interface on the MDS switches mds-switch-3.

Give the Gigabit Ethernet interface on the MDS switch mds-switch-3 an IP address and a subnetmask. This allows the Gigabit Ethernet interface to communicate with the network.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# interface gigabitethernet 2/1
mds-switch-3(config-if)# ip address 172.22.36.90 255.255.254.0
mds-switch-3(config-if)# no shut
mds-switch-3(config-if)# end
mds-switch-3#
```

#### Step 8 Configure an IP route so the two Gigabit Ethernet ports can communicate.

An IP route needs to be configured to allow the two Gigabit Ethernet ports on switches mds-switch-2 and mds-switch-3 to communicate with each other. In this recipe, the Gigabit Ethernet ports are in two different subnets, so they must have an explicit route for communication.



#### Tip

We recommend that you create a host route to each of the two Gigabit Ethernet interfaces with a subnet mask of 255.255.255.255. This allows only the two Gigabit Ethernet interfaces to communicate.

The syntax for configuring a route is shown below. For the Gigabit Ethernet port 1/1 on the switch mds-switch-2 to communicate with the port Gigabit Ethernet 2/1 on switch mds-switch-3, the following route configuration must be done on switch mds-switch-2.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# ip route 172.22.36.90 255.255.255.255 172.22.38.1 interface
gigabitethernet 1/1
mds-switch-2(config)# end
mds-switch-2#
```

The preceding configuration provides this information: to reach 172.22.36.90, use the gateway 172.22.38.1 and use interface Gigabit Ethernet 1/1 on switch mds-switch-2.

Similarly, for the Gigabit Ethernet port 2/1 on switch mds-switch-3 to communicate with Gigabit Ethernet port 1/1 on switch mds-switch-2, the following route configuration must be done on switch mds-switch-3.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
```

```
mds-switch-3(config)# ip route 172.22.38.140 255.255.255.255 172.22.36.1 interface
gigabitethernet 2/1
mds-switch-3(config)# end
mds-switch-3#
```

The above configuration provides this information: to reach 172.22.38.140, use the gateway 172.22.36.1 and interface Gigabit Ethernet 2/1 on switch mds-switch-3.

**Step 9** Ping the Gigabit Ethernet interfaces to ensure that the Gigabit Ethernet ports can communicate.

From the switch mds-switch-2, ping the IP address of the Gigabit Ethernet interface 2/1 on switch mds-switch-3. Similarly, ping the IP address of the Gigabit Ethernet interface 1/1 on switch mds-switch-2 from switch mds-switch-3. This can be done from the switch prompt.

```
mds-switch-2# ping 172.22.36.90 interface gigabitethernet 1/1
PING 172.22.36.90 (172.22.36.90) from 172.22.38.140 gige1-1: 56(84) bytes of data.
64 bytes from 172.22.36.90: icmp_seq=1 ttl=254 time=0.605 ms
64 bytes from 172.22.36.90: icmp_seq=2 ttl=254 time=0.510 ms
64 bytes from 172.22.36.90: icmp_seq=3 ttl=254 time=0.552 ms
64 bytes from 172.22.36.90: icmp_seq=4 ttl=254 time=0.511 ms
64 bytes from 172.22.36.90: icmp_seq=5 ttl=254 time=0.542 ms

--- 172.22.36.90 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4030ms
rtt min/avg/max/mdev = 0.510/0.544/0.605/0.034 ms
mds-switch-2#

mds-switch-3# ping 172.22.38.140 interface gigabitethernet 2/1
PING 172.22.38.140 (172.22.38.140) from 172.22.36.90 gige2-1: 56(84) bytes of data.
64 bytes from 172.22.38.140: icmp_seq=1 ttl=254 time=0.595 ms
64 bytes from 172.22.38.140: icmp_seq=2 ttl=254 time=0.531 ms
64 bytes from 172.22.38.140: icmp_seq=3 ttl=254 time=0.525 ms
64 bytes from 172.22.38.140: icmp_seq=4 ttl=254 time=0.540 ms
64 bytes from 172.22.38.140: icmp_seq=5 ttl=254 time=0.529 ms

--- 172.22.38.140 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4039ms
rtt min/avg/max/mdev = 0.525/0.544/0.595/0.025 ms
mds-switch-3#
```

**Step 10** Measure the RTT between the two Gigabit Ethernet interfaces.

Measure the RTT between the two Gigabit Ethernet interfaces. The RTT value is required for configuring the FCIP profile in the next step.

```
mds-switch-2# ips measure-rtt 172.22.36.90 interface gigabitethernet 1/1
Round trip time is 424 micro seconds (0.42 milli seconds)
mds-switch-2#

mds-switch-3# ips measure-rtt 172.22.38.140 interface gigabitethernet 2/1
Round trip time is 424 micro seconds (0.42 milli seconds)
mds-switch-3#
```



**Note**

FCIP uses TCP port 3225 by default. Since the FCIP tunnel to mds-switch-1 (fcip 1) is already up and using TCP port 3225, this FCIP tunnel will use TCP port 3500.

**Step 11** Configure the FCIP profile on both the switches.

An FCIP profile needs to be created. The profile defines the characteristics for the FCIP tunnel. The RTT measured is needed for profile configuration. In this case, the RTT is 425 micro seconds. The IP address used in the profile configuration that follows is the IP address assigned to the Gigabit Ethernet interface on the switch associated with the profile.

**Tip**

We recommend using the same profile numbers (unique to a tunnel) for the FCIP profiles configured on both switches for a given FCIP tunnel.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# fcip profile 2
mds-switch-2(config-profile)# port 3500 <-- TCP port configuration
mds-switch-2(config-profile)# ip address 172.22.38.140
mds-switch-2(config-profile)# tcp max-bandwidth-mbps 1000 min-available-bandwidth-mbps
500 round-trip-time-us 425
mds-switch-2(config-profile)# end
mds-switch-2#

mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# fcip profile 2
mds-switch-3(config-profile)# port 3500 <-- TCP port configuration
mds-switch-3(config-profile)# ip address 172.22.36.90
mds-switch-3(config-profile)# tcp max-bandwidth-mbps 1000 min-available-bandwidth-mbps
500 round-trip-time-us 425
mds-switch-3(config-profile)# end
mds-switch-3#
```

The FCIP profile 2 has now been defined on switches mds-switch-2 and mds-switch-3 using TCP port 3500.

**Step 12** Configure the FCIP interface on both switches.

In the FCIP interface configuration, the profile to be used and the peer information (remote Gigabit Ethernet IP address) are specified. Additionally, compression and write acceleration can be configured on the FCIP interface.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# interface fcip 2
mds-switch-2(config-if)# peer-info ipaddr 172.22.36.90 port 3500 <-- TCP port Config
mds-switch-2(config-if)# use-profile 2
mds-switch-2(config-if)# no shut
mds-switch-2(config-if)# end
mds-switch-2#
m
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# interface fcip 2
mds-switch-3(config-if)# peer-info ipaddr 172.22.38.140 port 3500 <-- TCP port Config
mds-switch-3(config-if)# use-profile 2
mds-switch-3(config-if)# no shut
mds-switch-3(config-if)# end
mds-switch-3#
```

**Step 13** The FCIP tunnel fcip 2 should be up and running. Use the **show fcip summary** command to see the status of the FCIP link between the two switches.

```
mds-switch-2# show fcip summary
```

```

```



```

Tun prof Eth-if peer-ip Status T W T Enc Comp Bandwidth rtt
 E A A max/min (us)

1 1 GE1/1 172.22.34.75 TRNK Y Y Y N M1 1000M/500M 425
2 2 GE1/1 172.22.36.90 TRNK Y N N N N 1000M/500M 425

```

```
mds-switch-2#
```

```
smds-switch-3# show fcip summary
```

```

Tun prof Eth-if peer-ip Status T W T Enc Comp Bandwidth rtt
 E A A max/min (us)

2 2 GE2/1 172.22.38.140 TRNK Y N N N N 1000M/500M 425

```

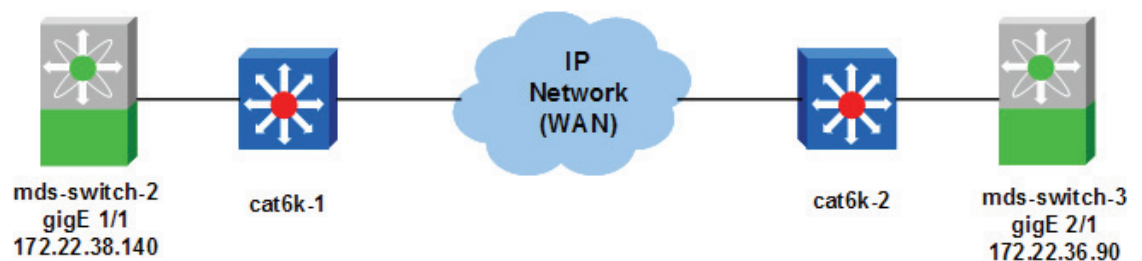
```
mds-switch-3#
```

The preceding output shows the status of the two tunnels IDs 1 and 2 are trunking on switch mds-switch-2 and one tunnel with the ID 2 is trunking on switch mds-switch-3.

## Configuring FCIP Using Fabric Manager

Cisco Fabric Manager can be used to configure FCIP. The following recipe demonstrates how FCIP is configured using Fabric Manager. For this recipe, interface Gigabit Ethernet 1/1 on mds-switch-2 and interface Gigabit Ethernet 2/1 on mds-switch-3 are used. The topology used in this recipe is shown in Figure 9-5.

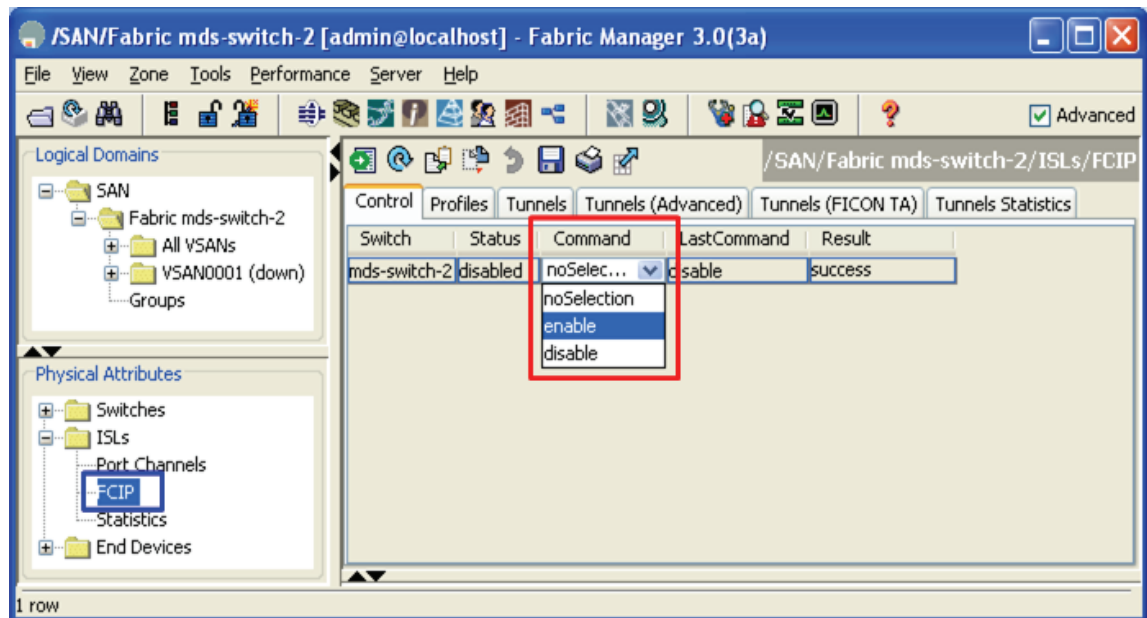
Figure 9-5 FCIP topology for Fabric Manager



To enable FCIP on the switches mds-switch-2 and mds-switch-3 and apply the configuration, follow these steps:

- Step 1** Select FCIP in the Physical Attributes Pane. This selection is highlighted with a blue rectangle in Figure 9-6. Once you select FCIP, the right pane is populated with a list of fabric switches seen by Fabric Manager, including the state of FCIP in each switch.
- Step 2** Select the required switch present in the right-hand side pane. In this case, it is mds-switch-2.
- Step 3** Expand the drop-down box in the **command column** and select the **enable** option. This process is marked using a red rectangle in Figure 9-6.
- Step 4** Apply the changes to the switch mds-switch-2 to enable FCIP on it. Click the green **Apply Changes** icon (see Figure 9-7) to apply the changes to the switch.

Figure 9-6 Enable FCIP Through Fabric Manager



- Step 5** Repeat steps 1 through 3 for the switch mds-switch-3. Click the green **Apply Changes** icon highlighted in blue (see Figure 9-7) to apply the changes to the switch.

Figure 9-7 Apply Changes to Switches Through Fabric Manager

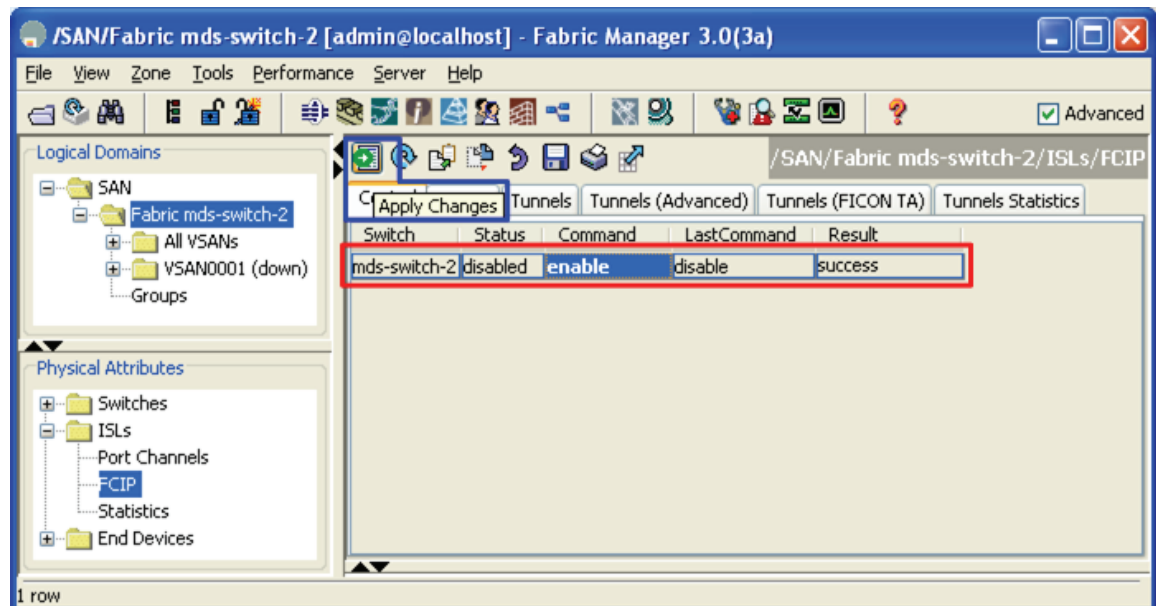
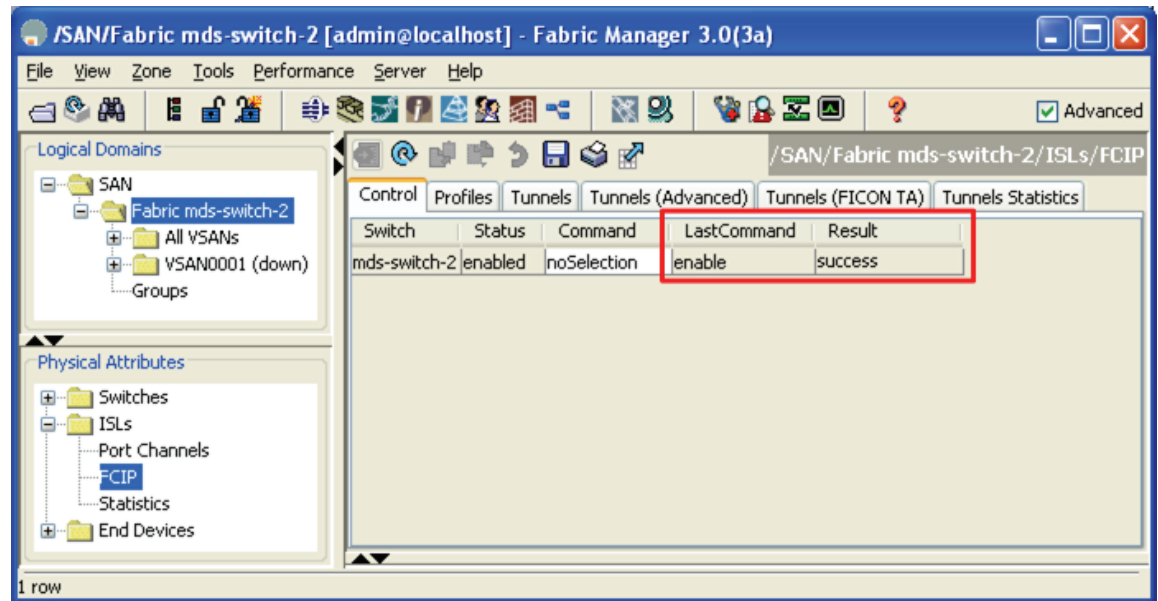


Figure 9-8 Enable Results

**Note**

IPSec is available only when the MPS-14 /2 module is used in the MDS 9216 switch, MDS 9506 switch, MDS 9509 switch, or MDS 9513 switch, or an MDS 9216i switch is used.

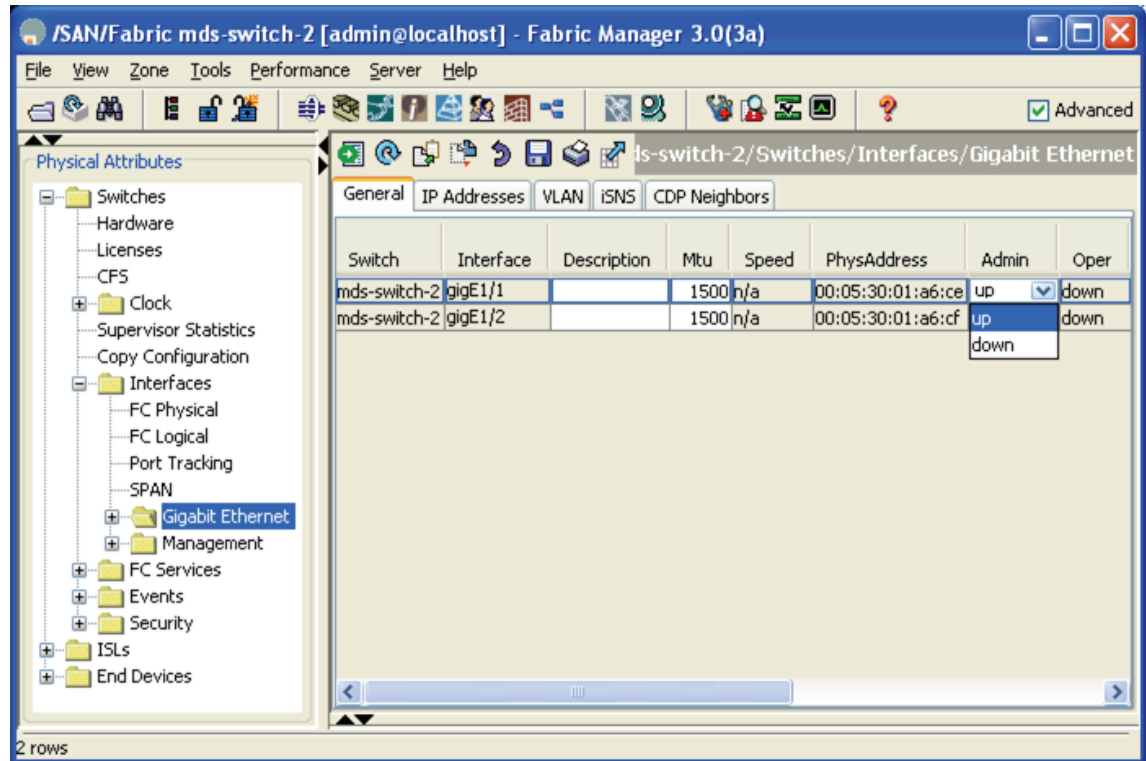
Once the changes are applied, the result of the changes can be seen in the **Last Command** and **Result** Column as shown in [Figure 9-8](#).

**Step 6** Perform steps 1 through 5 on switch mds-switch-3 to enable FCIP on it.

**Step 7** Configure IP addresses for the Gigabit Ethernet interfaces on switches mds-switch-2 and mds-switch-3.

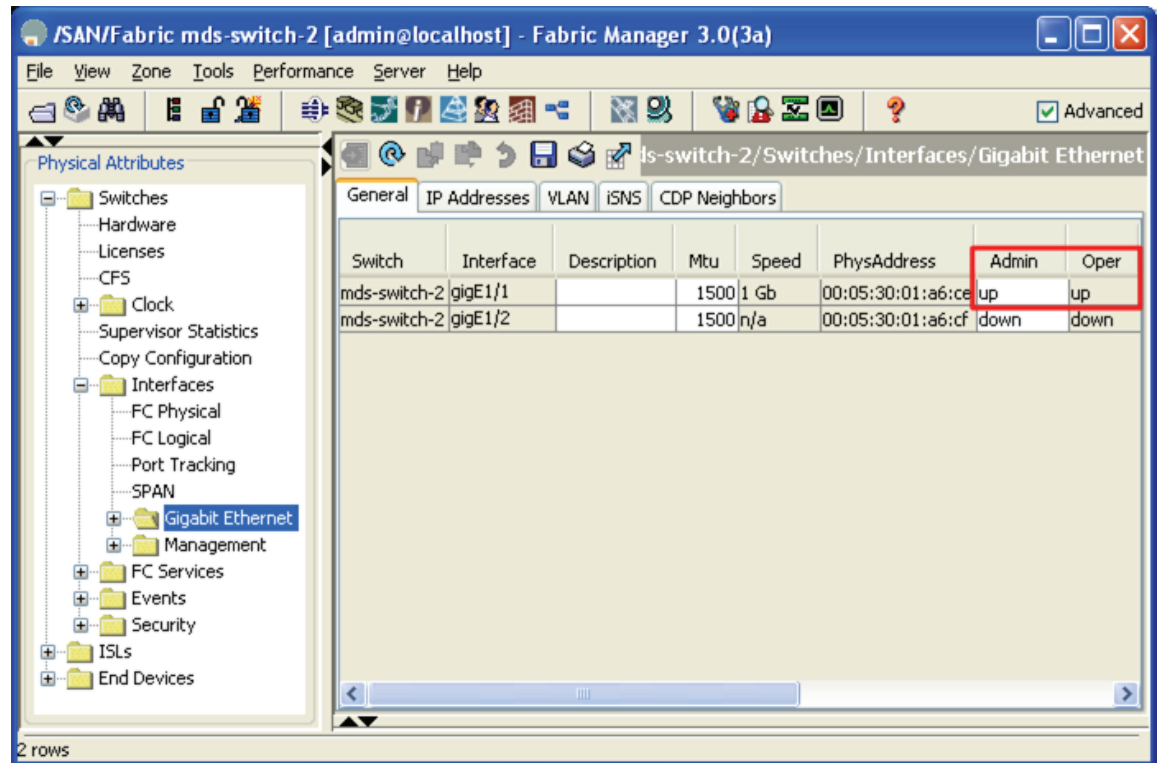
- a. From Physical Attributes pane, expand **Switches**, expand **Interfaces** then select **Gigabit Ethernet** (see [Figure 9-11](#)).
- b. Select the **General** Tab to enable the Gigabit Ethernet interface 1/1. From the **Admin** column pull-down menu, change the port state to **up**. ([Figure 9-9](#))

Figure 9-9 Enabling the Gigabit Ethernet Interface 1/1 on Switch MDS-Switch-2



- c. Click the green **Apply Changes** icon. This enables the interface Gigabit Ethernet1/1 on mds-switch-2. It is operationally up as seen in Figure 9-10 and is highlighted in red.

Figure 9-10 Gigabit Ethernet Ports Operational on MDS-Switch-2



- d. Select the **IP Addresses** tab on the right-hand pane. Then select the **create row** button highlighted in red in Figure 9-11. In the resulting popup dialog box, select the switch mds-switch-2 and add the interface, that is, Gigabit Ethernet port 1/1 and its IP address and net mask. The Gigabit Ethernet port 1/1 on mds-switch-2 in this recipe is assigned an IP address of 172.22.38.140 and a mask of 255.255.254.0 (/23). Click **create** to assign Gigabit Ethernet 1/1 the IP address. After the IP address is assigned to the interface, the IP address is displayed on the right-hand side pane highlighted in blue in Figure 9-12.

**Caution**

The IP address for the ports on the IPS module must be in a different subnet than the management interface. This is critical for FCIP to work on a switch.

Figure 9-11 Assign IP Address to the Gigabit Ethernet Ports on MDS-Switch-2

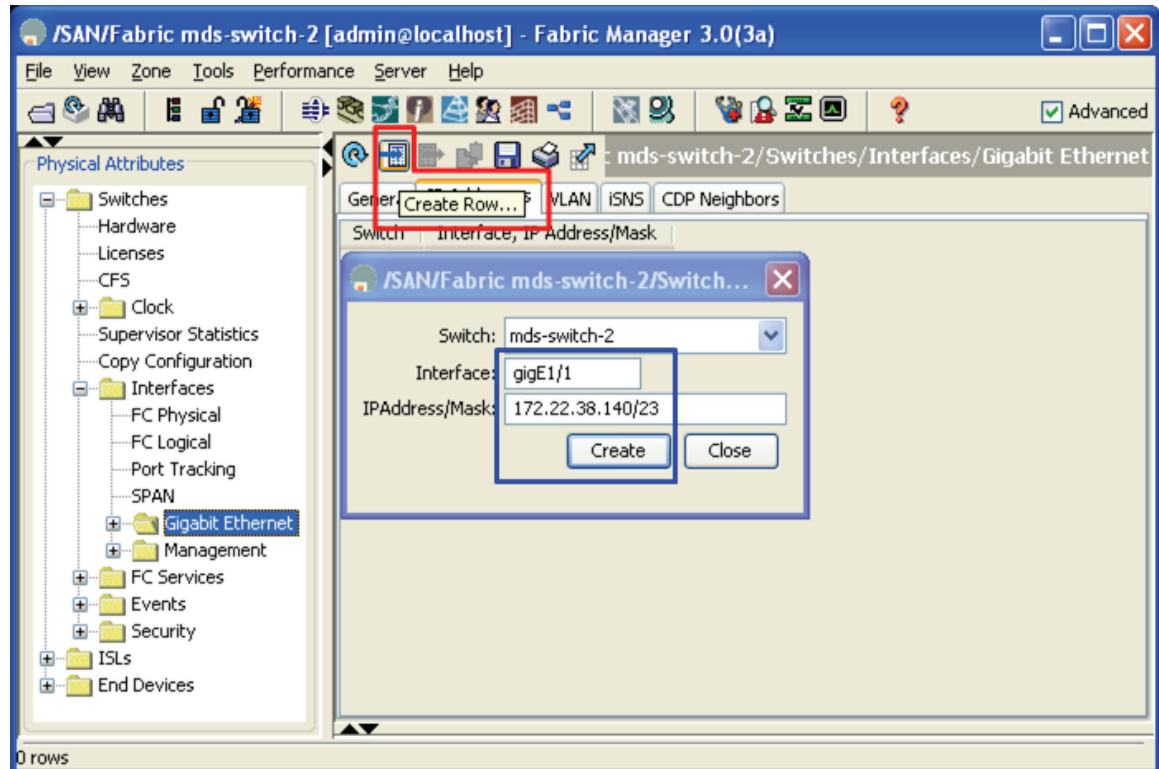
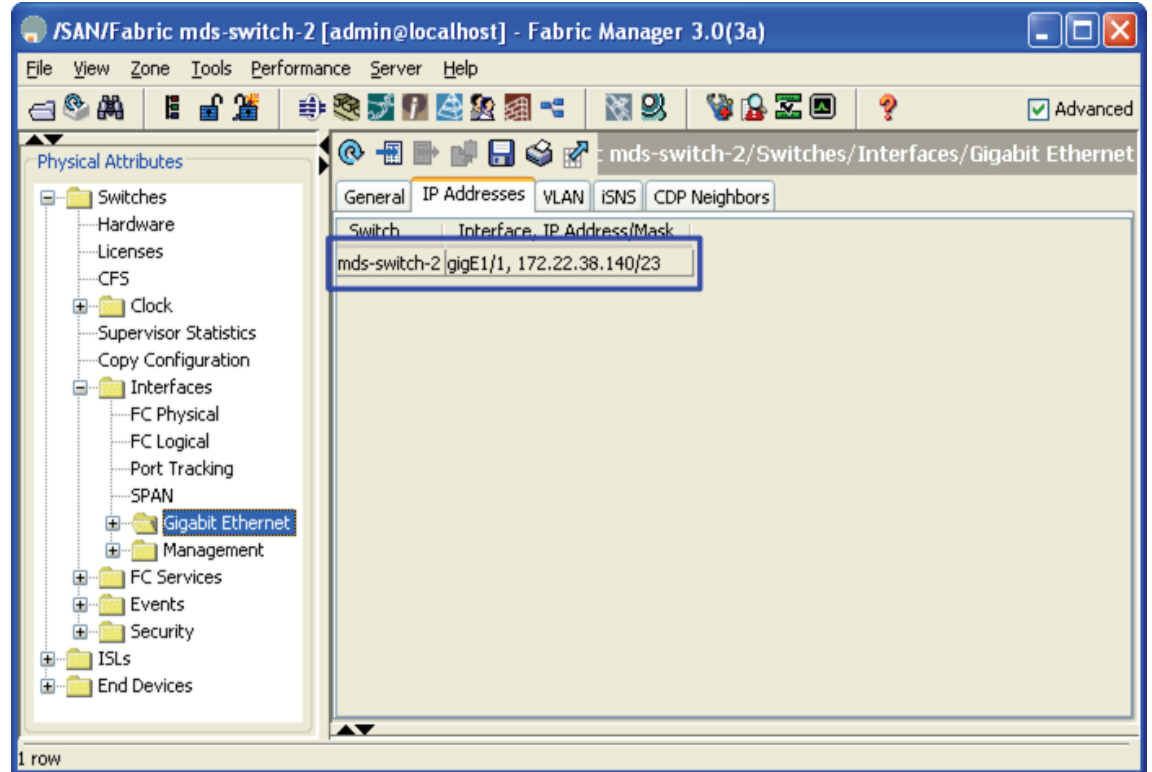


Figure 9-12 Assigned IP Address for the Interface Gigabit Ethernet1/1 on MDS-Switch-2



**Step 8** Repeat step 7 on the switch mds-switch-3, which is the other switch involved in the FCIP configuration. This process enables Gigabit Ethernet 2/1 and assigns it the IP address 172.22.36.90 and net mask of 255.255.255.0 (/23). The end results are shown in [Figure 9-13](#) and [Figure 9-14](#).

Figure 9-13 Gigabit Ethernet Ports Operational on MDS-Switch-3

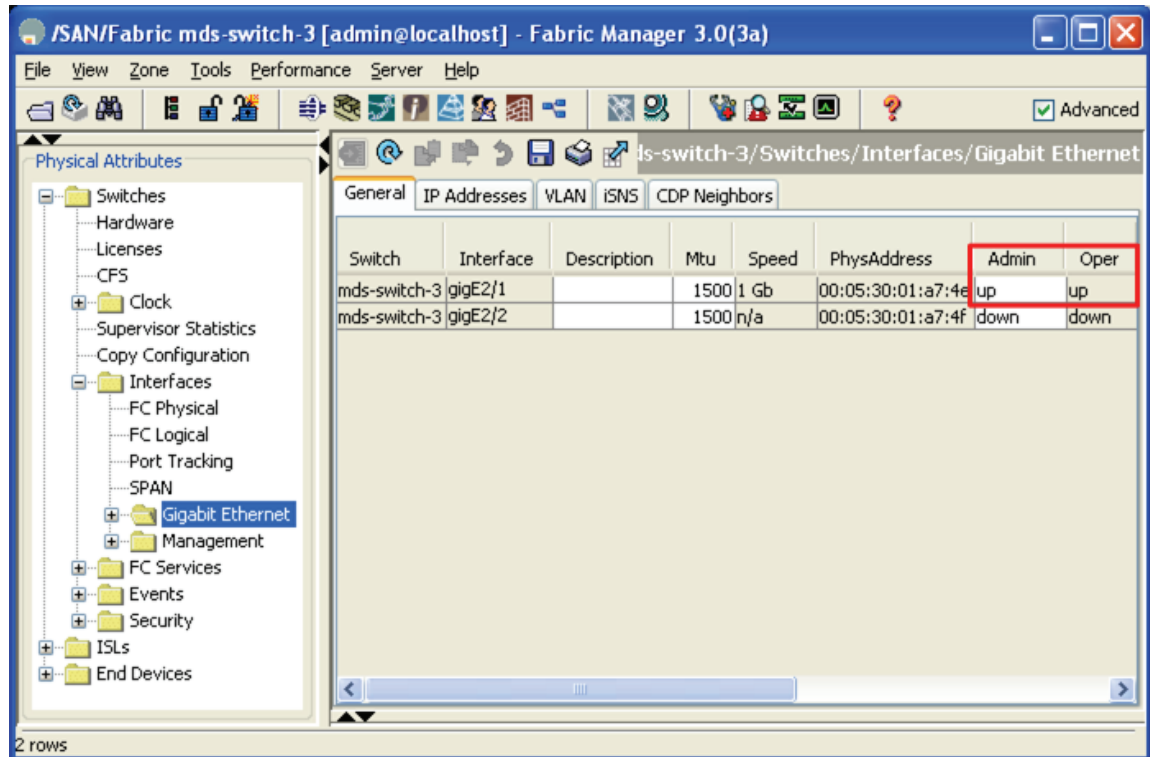
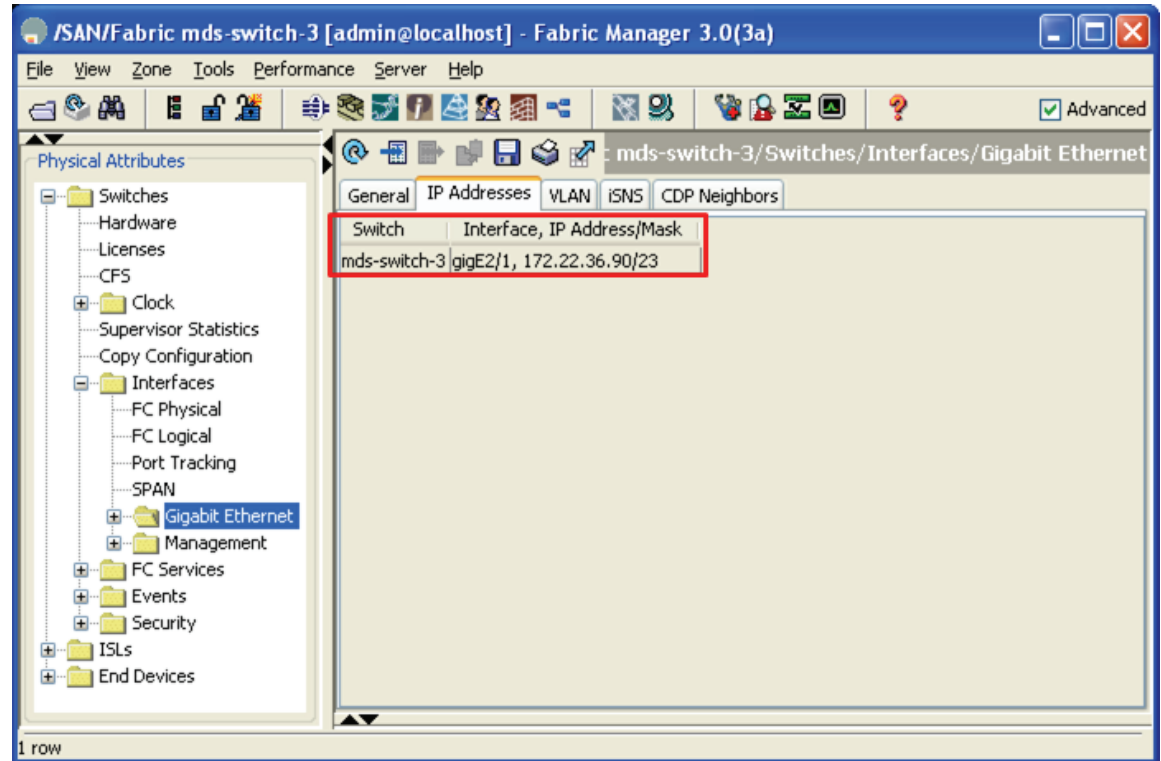




Figure 9-14 Assigned IP Address for the Interface Gigabit Ethernet1/1 on MDS-Switch-3

**Note**

We recommend creating a host route to each of the two Gigabit Ethernet interfaces with a subnet mask of 255.255.255.255. This allows only the two Gigabit Ethernet interfaces to communicate.

**Step 9** Configure IP routes so that the interface Gigabit Ethernet 1/1 on mds-switch2 and Gigabit Ethernet 2/1 on mds-switch-3 can communicate with each other. This is best accomplished using the CLI.

The syntax for configuring a route is shown below. For the Gigabit Ethernet port 1/1 on the switch mds-switch-2 to communicate with the port Gigabit Ethernet 2/1 on switch mds-switch-3 using Gigabit Ethernet 1/1, the following route configuration must be done on switch mds-switch-2.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# ip route 172.22.36.90 255.255.255.255 172.22.38.1 interface
gigabitethernet 1/1
mds-switch-2(config)# end
mds-switch-2#
```

The above configuration provides this information: to reach 172.22.36.90, use the gateway 172.22.38.1 and use interface Gigabit Ethernet 1/1 on switch mds-switch-2.

Similarly, for the Gigabit Ethernet port 2/1 on switch mds-switch-3 to communicate with Gigabit Ethernet port 1/1 on switch mds-switch-2 using Gigabit Ethernet 2/1, the following route configuration must be done on switch mds-switch-3.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# ip route 172.22.38.140 255.255.255.255 172.22.36.1 interface
gigabitethernet 2/1
mds-switch-3(config)# end
mds-switch-3#
```

The preceding configuration provides this information: to reach 172.22.38.140, use the gateway 172.22.36.1 and interface Gigabit Ethernet 2/1 on switch mds-switch-3.

**Step 10** Ping the Gigabit Ethernet interfaces to ensure that the Gigabit Ethernet ports can communicate.

From the switch mds-switch-2, ping the IP address of the Gigabit Ethernet interface 2/1 on switch mds-switch-3. Similarly, ping the IP address of the Gigabit Ethernet interface 1/1 on switch mds-switch-2 from switch mds-switch-3. This can be done from the switch prompt.

```
mds-switch-2# ping 172.22.36.90 interface gigabitethernet 1/1
PING 172.22.36.90 (172.22.36.90) from 172.22.38.140 gige1-1: 56(84) bytes of data.
64 bytes from 172.22.36.90: icmp_seq=1 ttl=254 time=0.605 ms
64 bytes from 172.22.36.90: icmp_seq=2 ttl=254 time=0.510 ms
64 bytes from 172.22.36.90: icmp_seq=3 ttl=254 time=0.552 ms
64 bytes from 172.22.36.90: icmp_seq=4 ttl=254 time=0.511 ms
64 bytes from 172.22.36.90: icmp_seq=5 ttl=254 time=0.542 ms

--- 172.22.36.90 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4030ms
rtt min/avg/max/mdev = 0.510/0.544/0.605/0.034 ms
mds-switch-2#

mds-switch-3# ping 172.22.38.140 interface gigabitethernet 2/1
PING 172.22.38.140 (172.22.38.140) from 172.22.36.90 gige2-1: 56(84) bytes of data.
64 bytes from 172.22.38.140: icmp_seq=1 ttl=254 time=0.595 ms
64 bytes from 172.22.38.140: icmp_seq=2 ttl=254 time=0.531 ms
64 bytes from 172.22.38.140: icmp_seq=3 ttl=254 time=0.525 ms
64 bytes from 172.22.38.140: icmp_seq=4 ttl=254 time=0.540 ms
64 bytes from 172.22.38.140: icmp_seq=5 ttl=254 time=0.529 ms

--- 172.22.38.140 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4039ms
rtt min/avg/max/mdev = 0.525/0.544/0.595/0.025 ms
mds-switch-3#
```



#### Note

It is critical to check the connectivity between the host NIC card and the Gigabit Ethernet port on the switch's IPS module before proceeding further. Use a ping test to check the connectivity.

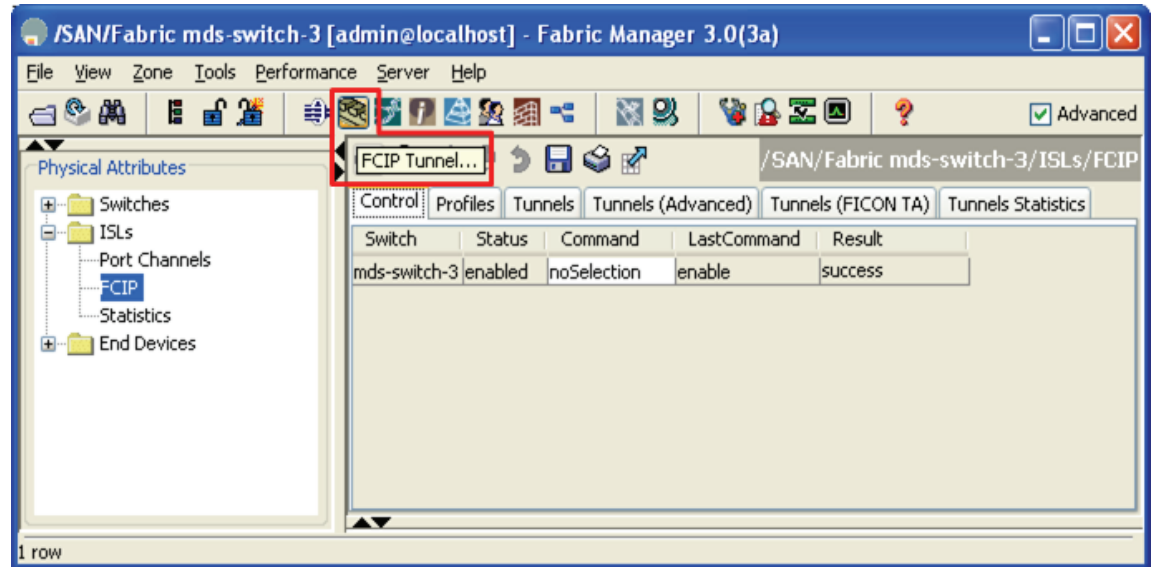
**Step 11** Create the FCIP tunnel between the two Gigabit Ethernet interfaces. Select the **FCIP Tunnel** icon highlighted in red in [Figure 9-15](#). This then launches the FCIP wizard which can be used to configure the FCIP tunnel between mds-switch-2 and mds-switch-3.



#### Note

The wizard is ideal for configuring point-to-point FCIP tunnels, but we recommend using the CLI to configure complex multiway tunnels.

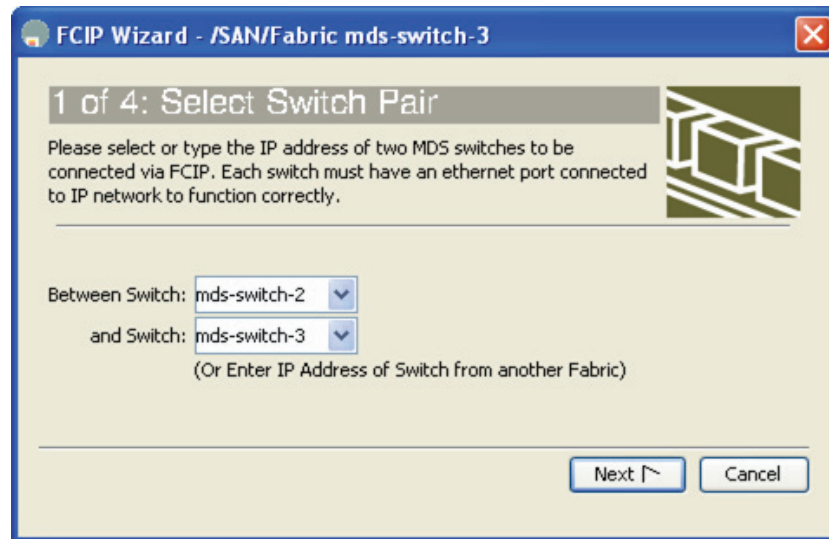
Figure 9-15 Starting FCIP Tunnel Wizard in Fabric Manager



The FCIP Tunnel Wizard launches as shown in Figure 9-16.

- a. In the wizard, enter the two switches between which the FCIP tunnel needs to be configured. In this recipe, the tunnel is between mds-switch-2 and mds-switch-3. Select the **Next** button.

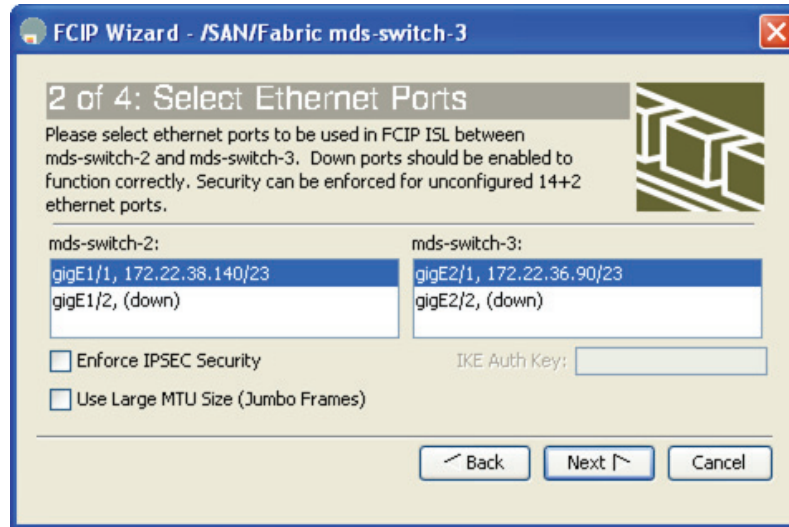
Figure 9-16 . FCIP Configuration Wizard



**Note** If DNS is configured for the MDS switches, the DNS names of the switches can be used. If DNS is not configured, then IP addresses can be used instead.

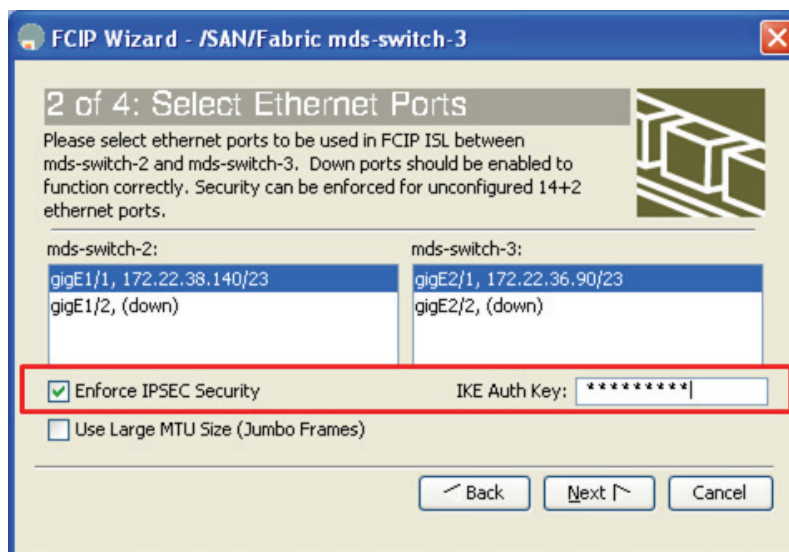
- b. In the next screen, select the Gigabit Ethernet interfaces that have been configured. It is Gigabit Ethernet 1/1 on mds-switch-2 and Gigabit Ethernet 2/1 on mds-switch-3. (See Figure 9-17)

Figure 9-17 Configured and Available Gigabit Ethernet Ports



- c. To protect the FCIP tunnel using IPSec, check the **Enforce IPSEC Security** check box. This is shown in Figure 9-18.

Figure 9-18 Configuration of IPsec for the FCIP Tunnel



- d. The IPsec configuration requires an Internet Key Exchange (IKE) authentication key. Supply an IKE authentication key (a pass phrase or key) as shown in Figure 9-18.

**Tip**

For the IKE authentication key, use an 8- or more character pass phrase or key.

- e. Click **Next**. In this screen the Tunnel properties are specified. These properties include maximum and minimum bandwidth for the tunnel, estimated RTT, write acceleration and compression. The first three properties are mandatory, while the last two are optional.

Figure 9-19 Define Tunnel Properties

**3 of 4: Specify Tunnel Properties**

Please supply the following parameters to tune the TCP connections. If Write Acceleration is enabled, ensure that flows will not load balanced across multiple ISLs.

Max Bandwidth:  1..1000 Mb

Min Bandwidth:  Mb

Estimated RTT (RoundTrip Time):  0..300000 us **Measure**

☐ Write Acceleration

☐ Enable Optimum Compression

Back Next Cancel

- f. Set the maximum bandwidth for the link (see Figure 9-19). Max Bandwidth is the maximum value the FCIP tunnel is allowed to use. If the IP link is dedicated for FCIP, then set Max Bandwidth to maximum available bandwidth of the IP link. For this recipe the maximum bandwidth is set to 1000 Mb (1Gb).
- g. The minimum bandwidth value is based on whether the link is shared or dedicated to FCIP. If the IP link is shared by multiple applications, then we recommend that you set this value to a reasonably low value, say half of the maximum bandwidth value.
- h. Click **Measure** to estimate the RTT circled in red.(see Figure 9-19). The RTT displayed in microseconds should be entered into the estimated RTT dialog box in the **Specify Tunnel** properties screen. In this topology, the RTT was 425 microseconds (see Figure 9-20).

**Note**

Cisco SAN-OS Release 2.1.x and higher strictly enforce bandwidth set by the maximum bandwidth property. Traffic shaper enforces the bandwidth allocated strictly.

Figure 9-20 Measure RTT output screen.

**Measure RTT**

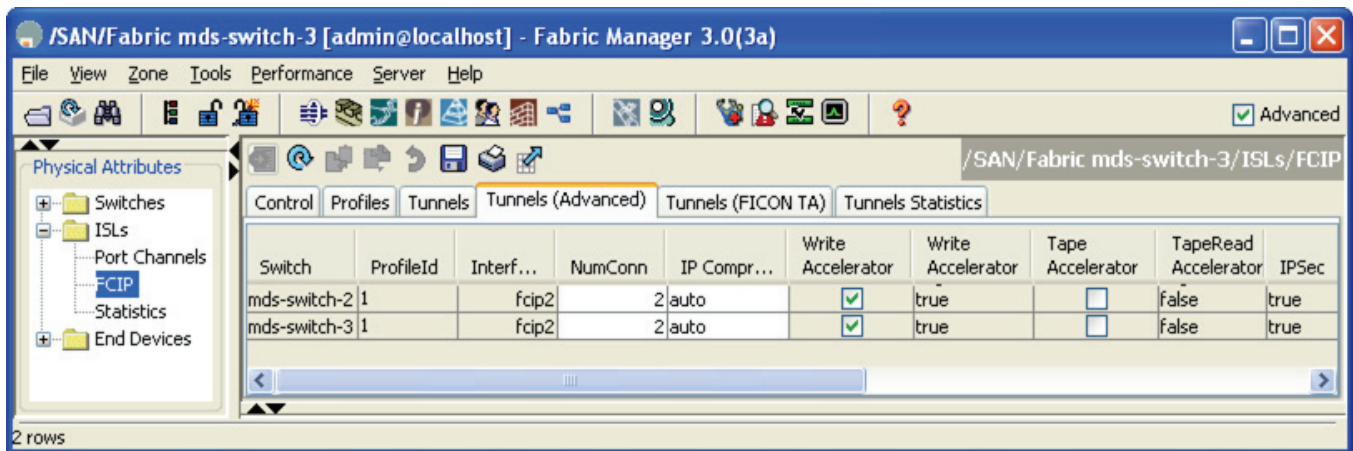
Round trip time is 424 micro seconds (0.42 milli seconds)

Close

Authentication successful

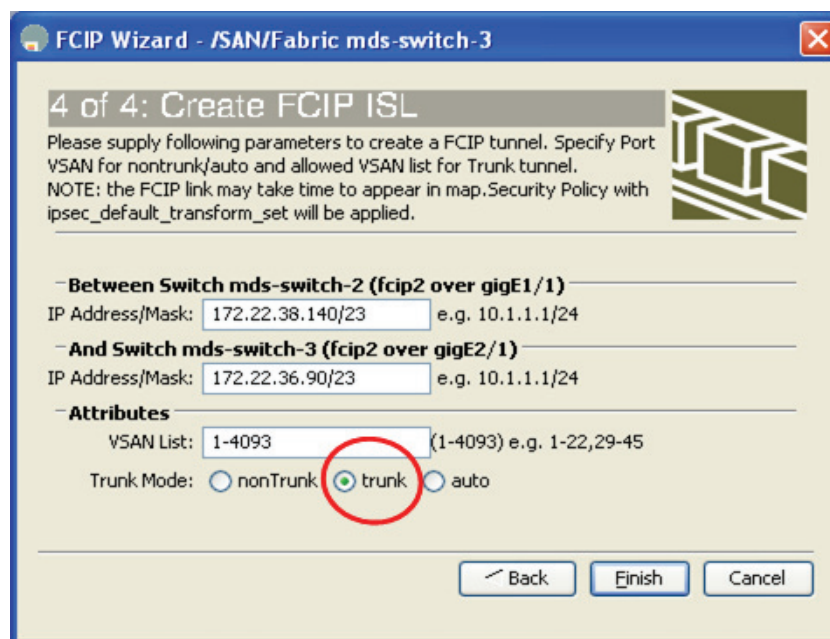
- i. Turn on write acceleration and compression by checking the corresponding check boxes. The final properties of the FCIP tunnel in the recipe with the Max and Min bandwidth, RTT, write acceleration and compression are shown in Figure 9-21. The discovered estimated RTT that was 425 microseconds is circled in red.

Figure 9-21 FCIP Tunnel Properties Configured



- j. Click **Next**. This will bring up the create FCIP ISL dialog box shown in Figure 9-22.

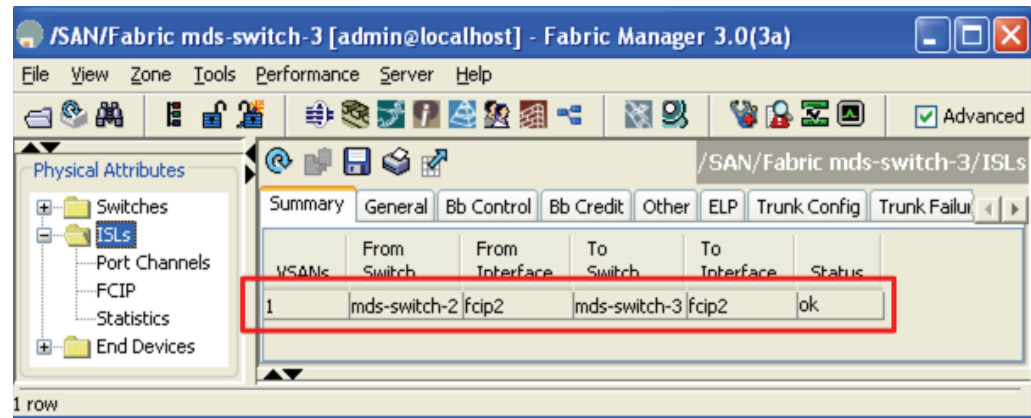
Figure 9-22 Create FCIP ISL



- Step 12** Verify the switches and the interfaces being connected.
- Step 13** Select trunk to turn on trunk mode (see red circle in Figure 9-22). This mode lets multiple VSANs flow through the link.
- Step 14** In this screen, the VSANs allowed through the FCIP link can be changed in the VSAN list dialog box. In the recipe, all VSANs (1 - 4093) are allowed through the FCIP tunnel.
- Step 15** Click **Finish** to complete the FCIP tunnel creation.
- Step 16** Check the tunnel status by selecting the ISL from the Physical Attributes on the left-hand pane. Then check for the FCIP status on the right-hand side pane (see Figure 9-23). The tunnel status is highlighted in red.

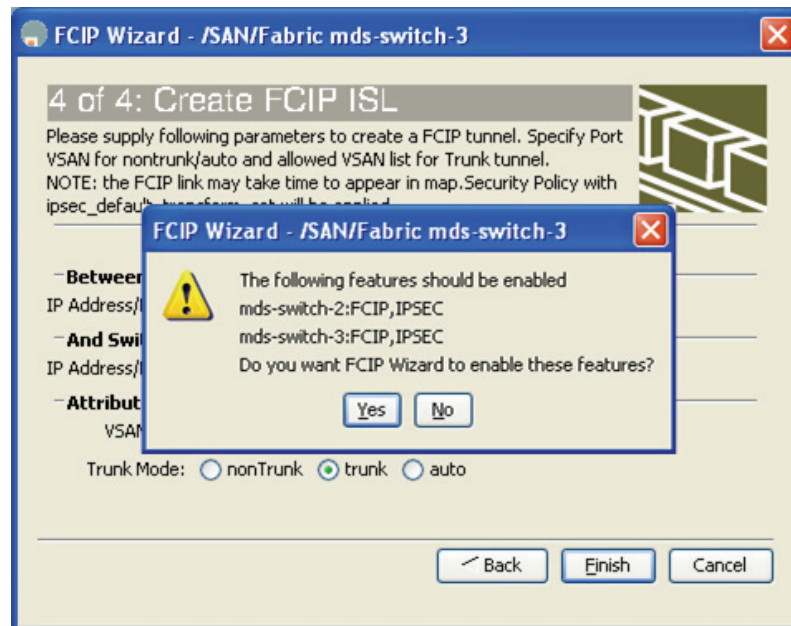


Figure 9-23 FCIP Tunnel Status



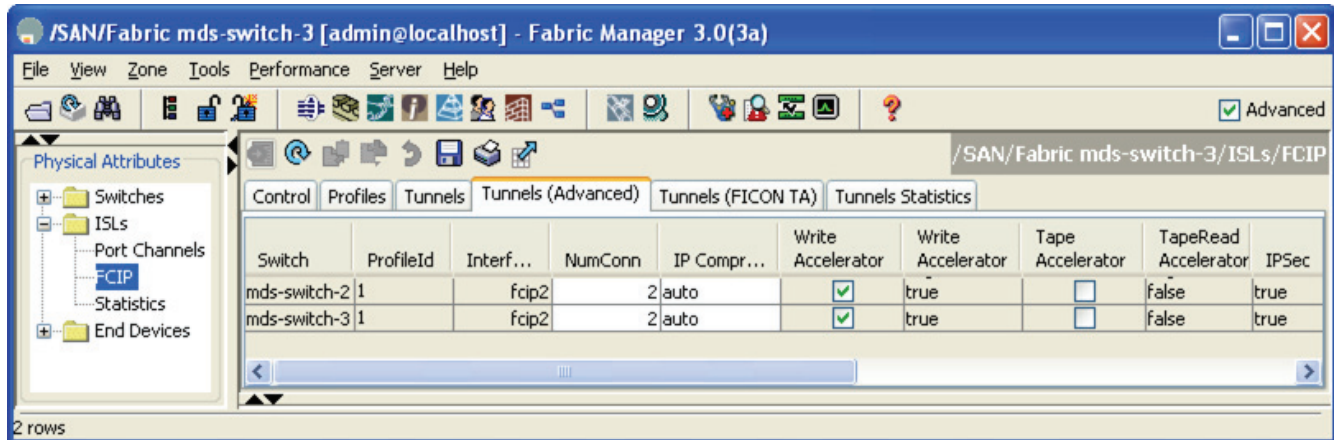
- Step 17** If IPsec is configured, the **Finish** button brings up the prompt to enable IPsec on switches mds-switch-2 and mds-switch-3 if it is not already enabled on the switches. Select **Yes** to complete the tunnel creation (see Figure 9-24).

Figure 9-24 IPSCE Enable Prompt on the Switches



- Step 18** You see the properties of the FCIP tunnel, select **ISL --> FCIP --> Tunnels (Advanced)** as shown in Figure 9-25.

Figure 9-25 FCIP Status with Compression, Write Acceleration Enabled



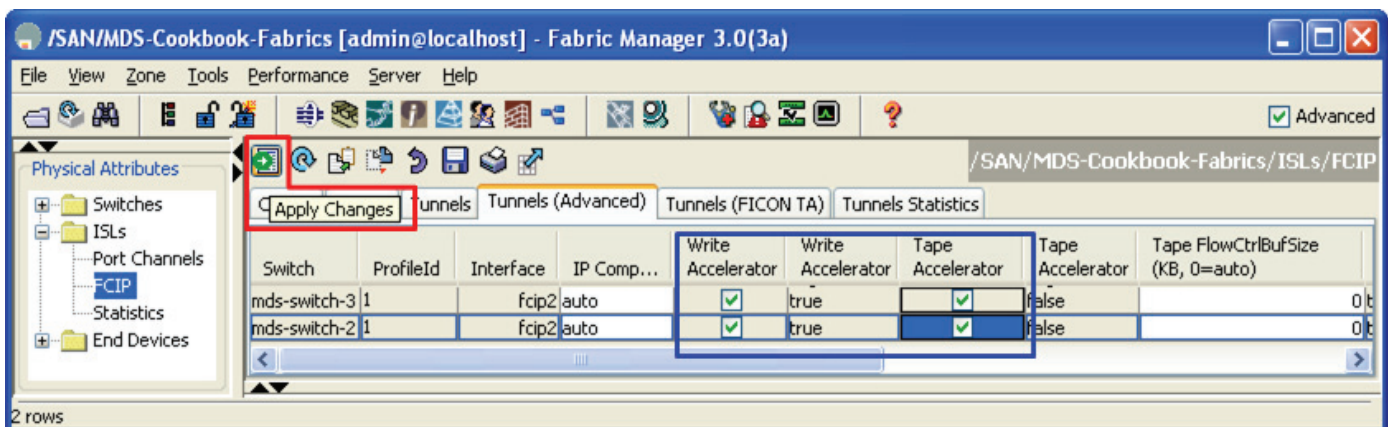
Tunnel creation is complete. The FCIP link secured by IPsec is configured and up.

## Enabling Tape Acceleration

To enable tape acceleration from Fabric Manager, follow these steps:

- Step 1** From the Physical Attributes pane, expand ISLs and select FCIP (see [Figure 9-26](#)).
- Step 2** In the right pane, choose the **Tunnels (Advanced)** tab.
- Step 3** Check the **Tape Accelerator** check boxes on both switches (see the blue box in [Figure 9-26](#)).
- Step 4** Click the green **Apply Changes** icon to apply the config to the switches (see the red box in [Figure 9-26](#)).

Figure 9-26 Activating Tape Acceleration in Fabric Manager



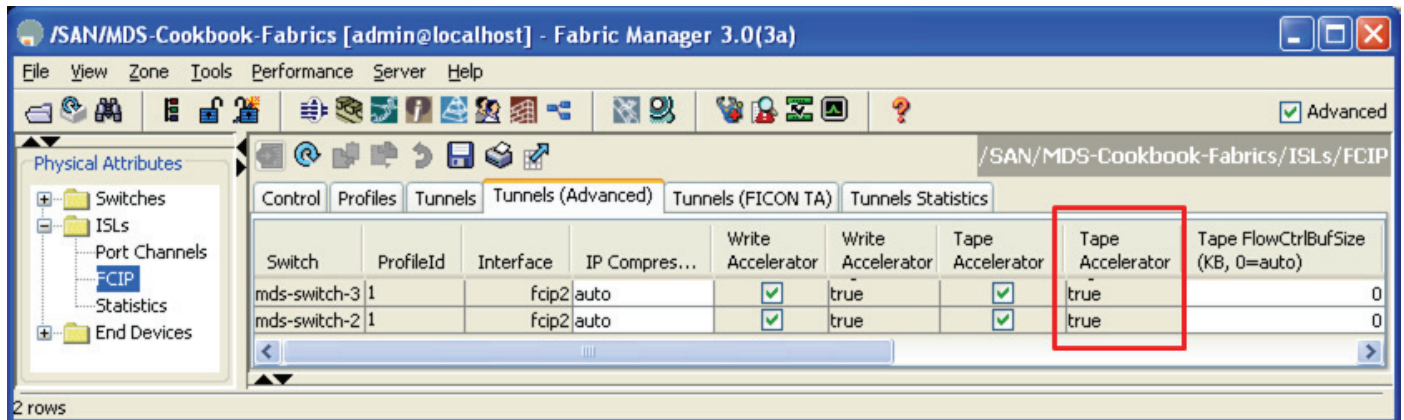


**Note**

For Tape Acceleration to work, write acceleration on the FCIP tunnel must be enabled. Also, tape acceleration will not function if multiple, equal cost paths or a PortChannel exists between the two switches.

**Step 5** Tune the Tape Flow Ctrl buffer size as needed (0=auto). (See the brown box in [Figure 9-26](#)).

Figure 9-27 Tape Acceleration Enable status



**Step 6** You see the properties of the FCIP tunnel select **ISL --> FCIP --> Tunnels (Advanced)** as shown in [Figure 9-27](#). The Tape Accelerator is true for the fcip2 tunnel, as highlighted in [Figure 9-27](#) with a red box.

## Testing and Tuning the FCIP Link with SET

SAN Extension Tuner (SET) is used to test and tune the performance for the FCIP link. SET generates SCSI I/O commands and directs them to virtual targets. SET allows for variation of the I/O type (read/writes) transfer size and the number of concurrent I/Os generated.

SET lets the user determine I/Os and throughput (MB) per second, as well as I/O latency. This helps in the fine tuning of FCIP throughput. The data generated can be used to validate characteristics of the WAN circuit, as well as determining the potential throughput of the FCIP tunnel without involving a host or disk subsystem.

SET is used to create consistent traffic flows, and enable such features as write acceleration, tape acceleration, compression, and encryption to determine their effect on the throughput of the tunnel. Also, SET is used to tune modifications to the FCIP tunnel's round trip time, and view maximum and minimum bandwidth.

SET can also be used to model an array to see if the array is performing up to specifications (number of outstanding I/Os and the size of transfers).

**Tip**

SET requires the SAN\_EXTN\_OVER\_IP license to work.

SET also requires the following resources:

- Two IPS modules.
- An FCIP link between the switches.
- One unused Gigabit Ethernet port per switch to act as a initiator or target.
- The physical layer of the second Gigabit Ethernet port should be up.
- iSCSI enabled on both the switches.
- SAN-EXT-TUNER enabled.

SET works by creating virtual initiators and targets behind two Gigabit Ethernet ports. These virtual devices are created in a VSAN, they have port world-wide names (pWWNs), and they obtain FC IDs just like real end devices do. They are required to be zoned together to communicate, and they send standard Fibre Channel Protocol commands to each other which are handled by the Fibre Channel infrastructure as is normal Fibre Channel traffic. The frames are routed via FSPF to their destinations and can travel on E and TE ports through MDS and nonMDS switches. If the minimum requirements are met, SET can be used to test optical networks not just FCIP links.

While the same Gigabit Ethernet port that is configured for FCIP can be used as a target or initiator, we do not recommend it because this may interfere with the ability to generate sufficient bandwidth. Always use an unused Gigabit Ethernet port for the initiator and target.

**Note**

This SET recipe assumes that the FCIP link is already up and functional. For directions, see [Configuring FCIP on a Switch with CLI, page 9-2](#)

For this recipe, the following resources are used:

- Switch: mds-switch-2:
  - Additional Gigabit Ethernet port: gig1/2
  - VSAN 1000
  - Virtual nWWN: 10:00:00:00:00:00:00:00
  - Virtual pWWN: 20:00:00:00:00:00:00:01
- Switch: mds-switch-3
  - Additional Gigabit Ethernet port: gig2/2:
  - VSAN 1000
  - Virtual nWWN 11:00:00:00:00:00:00:00
  - Virtual pWWN: 30:00:00:00:00:00:00:01

To tune a link, follow these steps:

**Step 1**

Enable the second Gigabit Ethernet port on both the switches. In this recipe Gigabit Ethernet port 2/2 is used on both switches for SET.

**Tip**

This additional Gigabit Ethernet port does not require an IP address to be assigned. Only the physical layer is required to be up.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config-if)#interface gigabitethernet 1/2
mds-switch-2(config-if)# no shut
mds-switch-2(config-if)# end
```

```

mds-switch-2#

mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config-if)#interface gigabitethernet 2/2
mds-switch-3(config-if)# no shut
mds-switch-3(config-if)# end
mds-switch-3#

```

Check to ensure that the physical layer of the Gigabit Ethernet port is up and running.

```

mds-switch-2# show interface gigabitethernet 1/2 brief

Interface Status IP Address Speed MTU Port
 Channel

GigabitEthernet1/2 up -- 1 Gbps 1500 --

mds-switch-3# sh int gigabitethernet 2/2 brief

Interface Status IP Address Speed MTU Port
 Channel

GigabitEthernet2/2 up -- 1 Gbps 1500 --

```

**Step 2** Enable iSCSI on both the switches mds-switch-2 and mds-switch-3 if it is not already enabled.

```

mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# enable iscsi
mds-switch-2(config)# end
mds-switch-2#

mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# iscsi enable
mds-switch-3(config)# end
mds-switch-3#

```

**Step 3** Enable the iSCSI on the second Gigabit Ethernet interface of both switches.

```

mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# interface iscsi 1/2
mds-switch-2(config-if)# no shut
mds-switch-2(config-if)# end
mds-switch-2#

mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# interface iscsi 2/2
mds-switch-3(config-if)# no shut
mds-switch-3(config-if)# end
mds-switch-3#

```

Verify that the iSCSI interface is up and running.

```

mds-switch-2# sh interface iscsi 1/2 brief

Interface Status Oper Mode Oper Speed
 (Gbps)

iscsil/2 up ISCSI 1
mds-switch-2#

```

```
mds-switch-3# sh interface iscsi 2/2 brief
```

| Interface | Status | Oper Mode | Oper Speed (Gbps) |
|-----------|--------|-----------|-------------------|
| iscsi2/2  | up     | ISCSI     | 1                 |

```
mds-switch-3#
```

#### Step 4 Enable SET on both switches.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# san-ext-tuner enable
mds-switch-2(config)# end
mds-switch-2#
```

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# san-ext-tuner enable
mds-switch-3(config)# end
mds-switch-3#
```



#### Tip

Using a separate VSAN for SET ensures that devices in other VSANs are not impacted by the SET traffic.

#### Step 5 Create a separate VSAN for SET N ports.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# vsan database
mds-switch-2(config-vsan-db)# vsan 1000 name SETVSAN
mds-switch-2(config-vsan-db)# end
mds-switch-2#
```

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# vsan database
mds-switch-3(config-vsan-db)# vsan 1000 name SETVSAN
mds-switch-3(config-vsan-db)# end
mds-switch-3#
```

#### Step 6 Configure nWWN and N port on both switches.

The N port pWWN on switch mds-switch-2 is configured as 20:00:00:00:00:00:01 with a nWWN of 10:00:00:00:00:00:00. Similarly, on switch mds-switch-3, the N port pWWN is configured as 30:00:00:00:00:00:01 and nWWN of 11:00:00:00:00:00:00. Both the N ports are made a part of VSAN 1000 which is created just for SET.

```
mds-switch-2# san-ext-tuner
mds-switch-2(san-ext)# nwwn 10:00:00:00:00:00:00
mds-switch-2(san-ext)# nport pwwn 20:00:00:00:00:00:01 vsan 1000 interface
gigabitethernet 1/2
mds-switch-2(san-ext-nport)# end
mds-switch-2#
```

```
mds-switch-3# san-ext-tuner
mds-switch-3(san-ext)# nwwn 11:00:00:00:00:00:00
mds-switch-3(san-ext)# nport pwwn 30:00:00:00:00:00:01 vsan 1000 interface
gigabitethernet 2/2
mds-switch-3(san-ext-nport)# end
mds-switch-3#
```

Verify that the created VSAN has logged on to the fabric.

```
mds-switch-2# sh flogi database

INTERFACE VSAN FCID PORT NAME NODE NAME

iscsi1/2 1000 0x410002 20:00:00:00:00:00:01 11:00:00:00:00:00:00
Total number of flogi = 1.
mds-switch-2#
```

```
mds-switch-3# sh flogi database v 1000

INTERFACE VSAN FCID PORT NAME NODE NAME

iscsi2/2 1000 0x0c0003 30:00:00:00:00:00:01 11:00:00:00:00:00:00
Total number of flogi = 1.
mds-switch-3#
```

The **show fcns database vsan 1000** command should display both the pWWNs in VSAN 1000.

```
mds-switch-2# show fcns database vsan 1000
VSAN 1000:

FCID TYPE PWWN (VENDOR) FC4-TYPE:FEATURE

0x0c0003 N 30:00:00:00:00:00:01 scsi-fcp
0x410002 N 20:00:00:00:00:00:01 scsi-fcp
Total number of entries = 2
mds-switch-2#
```

```
mds-switch-3# show fcns database vsan 1000
VSAN 1000:

FCID TYPE PWWN (VENDOR) FC4-TYPE:FEATURE

0x0c0003 N 30:00:00:00:00:00:01 scsi-fcp
0x410002 N 20:00:00:00:00:00:01 scsi-fcp
Total number of entries = 2
mds-switch-3#
```

**Step 7** Create a zone set and a zone in VSAN 1000 so the SET N ports can communicate. Zone these devices from Fabric Manager with the CLI as shown.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# zoneset name ZS_santune_vsan1000 vsan 1000
mds-switch-2(config-zoneset)# zone name Z_SET_VSAN100
mds-switch-2(config-zoneset-zone)# member pwwn 20:00:00:00:00:00:01
mds-switch-2(config-zoneset-zone)# member pwwn 30:00:00:00:00:00:01
mds-switch-2(config-zoneset-zone)# exit
mds-switch-2(config-zoneset)# exit
```

**Step 8** Activate the zone set. If enhanced zoning is enabled for this VSAN, then commit the zone changes.

```
mds-switch-2(config)# zoneset activate name ZS_santune_vsan1000 vsan 1000
Zoneset activation initiated. check zone status
mds-switch-2(config)# end
mds-switch-2#
```

**Step 9** Verify that the zone set is active and the two N ports are able to communicate.

```
mds-switch-2# show zoneset active vsan 1000
zoneset name santune vsan 1000
```

```

zone name Z_SET_VSAN100 vsan 1000
* fcid 0x410002 [pwwn 20:00:00:00:00:00:01]
* fcid 0x0c0003 [pwwn 30:00:00:00:00:00:01]
mds-switch-2#

```

- Step 10** Create tasks that either read or write from one N port to another. Two tasks, one to read and one to write, are created as examples. In the examples, the N port on switch mds-switch-2 acts as initiator and the N port on switch mds-switch-3 acts as a target.

**Note**

By default, an all-zero pattern is used as the pattern for data generated by the virtual N ports. Optionally specify a file as the data pattern to be generated by selecting a data pattern file.

```

mds-switch-2# san-ext-tuner
mds-switch-2(san-ext)# nport pwwn 20:00:00:00:00:00:01 vsan 1000 interface
gigabitethernet 1/2
mds-switch-2(san-ext-nport)# read command-id 1 target 30:00:00:00:00:00:01
transfer-size 1024000 outstanding-ios 5 continuous <-- read command
mds-switch-2(san-ext-nport)# write command-id 2 target 30:00:00:00:00:00:01
transfer-size 1024000 outstanding-ios 5 continuous <-- write command
mds-switch-2(san-ext-nport)#

```

**Note**

The transfer size should be a multiple of 512. The test can be continuous or can be limited to a certain number of transactions.

- Step 11** Gather throughput and performance data on the switch.

```

mds-switch-2# show san-ext-tuner interface gigabitethernet 1/2 nport pwwn
20:00:00:00:00:00:01 vsan 1000 counters
Statistics for nport
Node name 10:00:00:00:00:00:00 Port name 20:00:00:00:00:00:01
 I/Os per sec : 18
 Reads : 50%
 Writes : 50%
 Egress throughput : 9.06 MBs/sec (Max - 9.12 MBs/sec)
 Ingress throughput: 8.62 MBs/sec (Max - 10.45 MBs/sec)
 Average response time : Read - 572450 us, Write - 568564 us
 Minimum response time : Read - 343728 us, Write - 331788 us
 Maximum response time : Read - 1350666 us, Write - 990794 us
 Errors : 0
mds-switch-2

mds-switch-3# show san-ext-tuner interface gigabitethernet 2/2 nport pwwn
30:00:00:00:00:00:01 vsan 1000 counters
Statistics for nport
Node name 11:00:00:00:00:00:00 Port name 30:00:00:00:00:00:01
 I/Os per sec : 17
 Reads : 58%
 Writes : 41%
 Egress throughput : 8.84 MBs/sec (Max - 10.47 MBs/sec)
 Ingress throughput: 9.02 MBs/sec (Max - 9.42 MBs/sec)
 Average response time : Read - 447611 us, Write - 424872 us
 Minimum response time : Read - 36986 us, Write - 124183 us
 Maximum response time : Read - 1165843 us, Write - 1386055 us
 Errors : 0
mds-switch-3#

```

Collecting the data over the specific period of time will help calibrate the link and further tune the link for optimal throughput and performance.

- Step 12** Stop the data gathering tests.

```
mds-switch-2(san-ext-nport)# stop command-id 1
mds-switch-2(san-ext-nport)# stop command-id 2
mds-switch-2(san-ext-nport)#end
mds-switch-2#
```

---







# CHAPTER 10

## iSCSI

---

Internet SCSI (iSCSI) is a standards-based transport protocol used to transport SCSI packets over TCP/IP. The iSCSI protocol is used to carry SCSI commands and the SCSI response. iSCSI uses the existing TCP/IP infrastructure to provide storage connectivity to servers. It allows hosts to connect and to access storage over a TCP/IP network using a network interface card. iSCSI, like the Fibre Channel Protocol (FCP), provides block-level access to storage arrays.

The iSCSI driver is used on hosts to act as a bridge between the SCSI and the TCP/IP layers. The iSCSI driver translates the SCSI command and responses from the server to the storage devices into an iSCSI payload that can be carried over the regular TCP/IP networks. It also translates the incoming iSCSI payload from the storage to the server into the SCSI command or response on the servers.

Gigabit Ethernet NICs are widely used as a standard connectivity interface for iSCSI on servers. This can provide up to 1 Gbps speed connectivity between host and storage.

There are two types of iSCSI storage devices: the ones that are the native iSCSI devices that connect to the TCP/IP network and directly provide iSCSI storage to iSCSI initiators (such as an iSCSI-enabled storage array), and the iSCSI gateway devices that standard SCSI storage devices (that use the Fibre Channel Protocol) and use a iSCSI gateway to enable them to provide storage to iSCSI initiators (such as the MDS IP storage service modules and the MPS-14/2 module).

Today there is iSCSI driver support for almost all operating systems. The iSCSI driver uses the existing TCP/IP NICs to provide iSCSI storage connectivity. When the host sees the disks, the operating system can then create the required file system to use the disks.

The MDS 9200 and 9500 series switches support iSCSI. The switches require one of the IP storage services modules, IPS-8 or IPS-4, or the MPS-14/2 module to support iSCSI. The MDS switches support multiple configuration methods for iSCSI. The different configuration modes are iSLB (iSCSI server load-balancing), transparent mode, and proxy mode. These different modes of configuration are described in the following sections.

## iSLB Configuration Mode

We recommend iSLB configuration mode, especially for a large scale iSCSI deployment. iSLB allows the incoming iSCSI session from various servers to be load-balanced over a group of Gigabit Ethernet interfaces either on the same switch or across multiple switches. This also allows for failover of an iSCSI session to other Gigabit Ethernet interface in the group of switches used for load-balancing if one or more of the interfaces fail.

This mode of configuration uses Virtual Router Redundancy Protocol (VRRP) for failover and load balancing among the various Gigabit Ethernet interfaces used in the configuration. The load balancing on the switch is based on the load on each of the Gigabit Ethernet interfaces, and is automatically

calculated and enforced on the switches. If the Gigabit Ethernet interfaces using VRRP are located on multiple switches, the switches on which the Gigabit Ethernet interfaces reside have to be interconnected using a PortChannel for failover and load balancing to work. The switches in this type of configuration use CFS to distribute the iSLB configuration between the switches. The iSLB mode can perform automatic zones and zone set activation, if so configured. The recipes below use explicit zones.

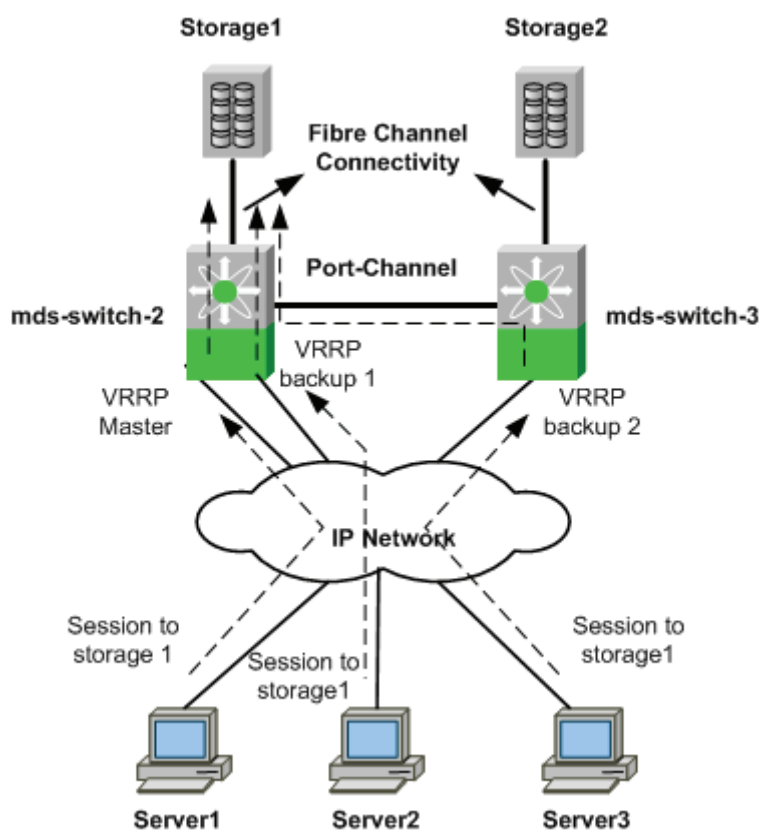
**Note**

iSLB mode is available only in Cisco SAN-OS Release 3.x and higher.

## Configuring iSLB on an MDS Switch

The topology used for this recipe has two MDS switches connected together using a PortChannel. The switches used are mds-switch-2 and mds-switch-3. The mds-switch-2 has two Gigabit Ethernet interfaces and the switch mds-switch-3 has one Gigabit Ethernet interface enabled for iSCSI. A total of three servers are used as the iSCSI client that connects to the target port (see [Figure 10-1](#)).

Figure 10-1 iSLB Topology



The topology has two storage ports, storage1 and storage 2, connected to mds-switch-2 and mds-switch-3 switches using Fibre Channel connects. Both the switches also have one MPS module each. The Gigabit Ethernet port 1/1 and 1/2 on mds-switch-2 and Gigabit Ethernet port 2/1 on mds-switch-3 are used in this recipe. The three iSCSI clients in the configuration talk to the same storage port storage1.

The assumptions are that the two switches mds-switch-2 and mds-switch-3 are linked together through a PortChannel. Refer to [PortChannels, page 5-1](#) on creating and managing PortChannels.

**Note**

If multiple switches are involved in the failover configuration as seen in [Figure 10-1](#) the switches need to be interconnected using a PortChannel for the iSLB load balancing and failover to work correctly.

To configure an iSLB based iSCSI configuration, follow these steps:

**Step 1**

Enable iSCSI in the switches involved in the topology. Do this before attempting to configure it on all the switches involved in this configuration.

**Caution**

If you do not enter the **iSCSI enable** command, further iSCSI configuration is not possible. This command enables any further iSCSI configuration options in the CLI.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# iscsi enable
mds-switch-2(config)# end
mds-switch-2#
```

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# iscsi enable
mds-switch-3(config)# end
mds-switch-3#
```

**Step 2**

Configure the IP address on the Gigabit Ethernet interfaces for the interfaces involved in all the switches involved.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# interface gigabitethernet 1/1
mds-switch-2(config-if)# ip address 172.22.38.140 255.255.254.0
mds-switch-2(config-if)# no shut
mds-switch-2(config-if)# interface gigabitethernet 1/2
mds-switch-2(config-if)# ip address 172.22.38.144 255.255.254.0
mds-switch-2(config-if)# no shut
mds-switch-2(config-if)# end
mds-switch-2#
```

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# interface gigabitethernet 2/1
mds-switch-3(config-if)# ip address 172.22.38.145 255.255.254.0
mds-switch-3(config-if)# no shut
```

**Step 3**

Enable the respective iSCSI interfaces in all the switches involved.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# interface iscsi 1/1-2
mds-switch-2(config-if)# no shut
```

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# interface iscsi 2/1
mds-switch-3(config-if)# no shut
```

- Step 4** Enable CFS distribute for iSLB for all the switches interconnected by an ISL or PortChannels.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# islb distribute
```

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# islb distribute
```

- Step 5** Configure the VRRP group to include all the Gigabit Ethernet interfaces that are part of the configuration. The VRRP group ID 50 is used in this example.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# interface gigabitethernet 1/1
mds-switch-2(config-if)# vrrp 50
mds-switch-2(config-if-vrrp)# address 172.22.38.146 <-- is the VR address of VRRP group 50
mds-switch-2(config-if-vrrp)# no shut
mds-switch-2(config-if-vrrp)#exit
mds-switch-2(config-if)# exit
mds-switch-2(config)# interface gigabitethernet 1/2
mds-switch-2(config-if)# vrrp 50
mds-switch-2(config-if-vrrp)# address 172.22.38.146 <-- is the VR address of VRRP group 50
mds-switch-2(config-if-vrrp)# no shut
mds-switch-2(config-if-vrrp)# end
mds-switch-2#
```

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# interface gigabitethernet 2/1
mds-switch-3(config-if)# vrrp 50
mds-switch-3(config-if-vrrp)# address 172.22.38.146 <-- is the VR address of VRRP group 50
mds-switch-3(config-if-vrrp)# no shut
mds-switch-3(config-if-vrrp)# end
mds-switch-3#
```

- Step 6** Configure VRRP iSLB load balancing. First enable iSLB load balancing for the VRRP group configured in the preceding step in all the switches where the VRRP group resides. In this example, that includes switches mds-switch-2 and mds-switch-3.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# islb vrrp 50 load-balance
```

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# islb vrrp 50 load-balance
```

- Step 7** Enable the iSCSI initiator in dynamic iSLB mode so that the new iSCSI initiators are initialized as iSLB clients and not iSCSI clients.



**Note**

The iSCSI initiators cannot be converted into iSLB initiators automatically. They have to be removed and reconfigured.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-2(config)# iscsi dynamic initiator islb
```

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
```

```
mds-switch-3(config)# iscsi dynamic initiator islb
```

- Step 8** Configure iSLB initiators for all the iSCSI clients statically. This is a requirement for load balancing. Assign the iSLB initiator to a VSAN. The system dynamically assigns a pWWN for the initiators configured.

```
mds-switch-2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# islb initiator name iqn.1991-05.com.microsoft:sjc7-pc-1
mds-switch-3(config-islb-init)# static nWWN system-assign
mds-switch-3(config-islb-init)# vsan 1000
mds-switch-3(config-islb-init)# exit
mds-switch-3(config)# islb initiator name iqn.1991-05.com.microsoft:sjc7-pc-2
mds-switch-3(config-islb-init)# static nWWN system-assign
mds-switch-3(config-islb-init)# vsan 1000
mds-switch-3(config-islb-init)# exit
mds-switch-3(config)# islb initiator name iqn.1991-05.com.microsoft:sjc7-pc-14
mds-switch-3(config-islb-init)# static nWWN system-assign
mds-switch-3(config-islb-init)# vsan 1000
mds-switch-3(config-islb-init)# end
mds-switch-3#
```

- Step 9** Configure iSLB virtual targets that will be zoned to the previously created iSLB initiators. Allow the configured initiators to access this target. The virtual target will be assigned its own pWWN, which can be seen using a **show fcns database** command.

**Note**

The initiators have to be authorized to access the virtual targets. You can do this in two ways. The first method uses the **all-initiator-permit** option for each virtual target configured. *If this method is used to enable initiator access to a virtual target, we strongly recommend that some sort of array-based LUN masking be deployed to prevent unwanted LUN access.* The second method is to individually allow each individual initiator to access the virtual target. This recipe uses the first method.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# islb virtual-target name iqn.storage1-CL3D
mds-switch-3(config-islb-tgt)# pWWN 50:06:0e:80:03:4e:95:23
mds-switch-3(config-islb-tgt)# all-initiator-permit
mds-switch-3(config-islb-tgt)# end
mds-switch-3#
```

- Step 10** Configure the iSCSI initiators on the servers such that the virtual target address is that of the virtual router interface address. In this recipe, configure the iSCSI initiator on the hosts / servers with the VR address of the VRRP group 50 configured previously. In this recipe it is 172.22.38.146. See [“Configuring iSCSI Client Initiators on Hosts” procedure on page 10-15](#) for more information on how to perform this task on different operating systems.
- Step 11** Configure the required zones and add them to the zone set by creating the required zones and zone set to permit the initiators to access the virtual target configured. In the recipe, VSAN 1000 uses enhanced zoning. The zone database has to be committed before activation. Then add the zones to a zone set if one exists or else create a new zone set and add the zones as members to it. Activate the zone set.

**Note**

For enhanced zoning, the zone database had to be committed before activating the zone set.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# zone name Z_islb_pcl vsan 1000
mds-switch-3(config-zone)# member symbolic-nodename iqn.1991-05.com.microsoft:sjc7-pc-1
```

```

mds-switch-3(config-zone)# member device-alias storage-CL3D
mds-switch-3(config-zone)# zone name Z_islb_pc2 vsan 1000
mds-switch-3(config-zone)# member symbolic-nodename iqn.1991-05.com.microsoft:sjc7-pc-2
mds-switch-3(config-zone)# member device-alias storage-CL3D
mds-switch-3(config-zone)# zone name Z_islb_pc14 vsan 1000
mds-switch-3(config-zone)# member symbolic-nodename iqn.1991-05.com.microsoft:sjc7-pc-14
mds-switch-3(config-zone)# member device-alias storage-CL3D
mds-switch-3(config-zone)# zoneset name ZS_islb vsan 1000
mds-switch-3(config-zoneset)# member Z_islb_pc1
mds-switch-3(config-zoneset)# member Z_islb_pc2
mds-switch-3(config-zoneset)# member Z_islb_pc14
mds-switch-3(config-zoneset)# exit
mds-switch-3(config)# zone commit vsan 1000
Commit operation initiated. Check zone status
mds-switch-3(config)# zoneset activate name ZS_islb vsan 1000
 Zoneset activation initiated. check zone status
mds-switch-3(config)#

```

Run the **show zoneset active vsan 1000** command to see the status of the zone set.

```

mds-switch-3# sh zoneset active vsan 1000
zoneset name ZS_islb vsan 1000
 zone name Z_islb_pc1 vsan 1000
 * fcid 0xea0001 [symbolic-nodename iqn.1991-05.com.microsoft:sjc7-pc-1]
 * fcid 0x8b0100 [pwwn 50:06:0e:80:03:4e:95:23] [storage-CL3D]

 zone name Z_islb_pc2 vsan 1000
 * fcid 0x8b0100 [pwwn 50:06:0e:80:03:4e:95:23] [storage-CL3D]
 * fcid 0xea0004 [symbolic-nodename iqn.1991-05.com.microsoft:sjc7-pc-2]

 zone name Z_islb_pc14 vsan 1000
 * fcid 0x8b0102 [symbolic-nodename iqn.1991-05.com.microsoft:sjc7-pc-14]
 * fcid 0x8b0100 [pwwn 50:06:0e:80:03:4e:95:23] [storage-CL3D]
mds-switch-3#

```

Run the **show islb vrrp summary** command to see the load-balance status.

```
mds-switch-3# sh islb vrrp summary
```

```

-- Groups For Load Balance --

 VR Id VRRP Address Type Configured Status

 50 IPv4 Enabled

-- Interfaces For Load Balance --

 VR Id VRRP IP Switch WWN Interface Initiator Load Redirect
 Enabled

 50 172.22.38.146 20:00:00:0d:ec:24:5e:80 GigE1/1 1000 Yes
 50 172.22.38.146 20:00:00:0d:ec:24:5e:80 GigE1/2 1000 Yes
M 50 172.22.38.146 20:00:00:0d:ec:24:5b:c0 GigE2/1 1000 Yes

-- Initiator To Interface Assignment --

Initiator VR Id VRRP IP Switch WWN Interface

iqn.1991-05.com.microsoft:sjc7-pc-1
 50 172.22.38.146 20:00:00:0d:ec:24:5e:80 GigabitEthernet1/1
iqn.1991-05.com.microsoft:sjc7-pc-14
 50 172.22.38.146 20:00:00:0d:ec:24:5b:c0 GigabitEthernet2/1
iqn.1991-05.com.microsoft:sjc7-pc-2
 50 172.22.38.146 20:00:00:0d:ec:24:5e:80 GigabitEthernet1/2

```

```
mds-switch-3#
```

This lists the VRRP group members involved and which interface is the master in the **Interfaces For Load Balance** section. It also lists the way the initiators are distributed to the various Gigabit Ethernet interfaces for load balancing in the **Initiator To Interface Assignment** section of the output.

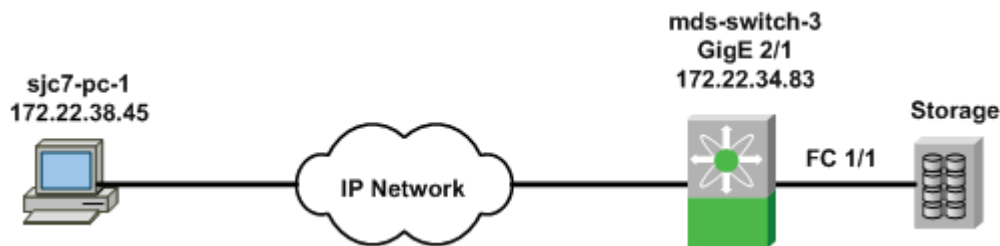
- Step 12** If multiple switches are involved, then do a CFS commit and CFS distribute the configuration to all the switches involved. Then activate the zone set so the initiators and targets can communicate.

## Configuring iSCSI on an MDS Switch in Transparent Mode

This recipe shows the configuration of iSCSI on an MDS switch. Transparent mode configures an equivalent Fibre Channel initiator for each iSCSI initiator. In this process, no logical unit number (LUN) masking or reassignment is done on the switch. For larger installations, iSCSI should be configured using the proxy initiator mode, see [Configuring iSCSI on the MDS Switch in Proxy Initiator Mode, page 10-11](#). Transparent mode is only recommended for very small iSCSI deployments.

The topology for this recipe is shown in [Figure 10-2](#).

Figure 10-2 iSCSI Topology



The topology consists of a Windows 2003 server using a Gigabit Ethernet NIC for iSCSI. It is connected to the network. The NIC dedicated for iSCSI on the host is assigned the IP address 172.22.38.45. The MPS port 2/1 on the switch mds-switch-3 is also connected to the network and has the IP address 172.22.34.83. The storage port from the array is connected to the Fibre Channel port 1/1 on mds-switch-3. This example shows how to configure an iSCSI initiator using IQN (iSCSI qualified name).



### Caution

The IP address for the ports on the IPS module should be in a different subnet than the management interface. This is critical for iSCSI to work on the switch.

To configure iSCSI initiator on the MDS switch, follow these steps:

- Step 1** Configure the Gigabit Ethernet interface on the MDS switch.

The Gigabit Ethernet interface on the MDS switch is given an IP address and a subnet mask. This allows the Gigabit Ethernet interface to communicate with the network.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# interface gigabitethernet 2/1
mds-switch-3(config-if)# ip address 172.22.34.83 255.255.254.0
```

```
mds-switch-3(config-if)# end
mds-switch-3#
```

- Step 2** Configure IP routes as required. In the recipe, the initiator sjc7-pc-1 is in the 172.22.38.0 subnet, while the Gigabit Ethernet interface is in the 172.22.34.0 subnet. To allow the initiator and the Gigabit Ethernet port to communicate, an IP route has to be configured on the switch.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)#ip route 172.22.38.45 255.255.255.255 172.22.34.1 interface
gigabitethernet 2/1
mds-switch-3(config)# end
mds-switch-3#
```

**Note**

It is critical to check the connectivity between the host NIC card and the Gigabit Ethernet port on the switch's IPS module before proceeding further. A ping test is sufficient. We recommend adding host based routes.

- Step 3** Ping the Gigabit Ethernet interface from the host. Similarly ping the host from the switch.

```
mds-switch-3#ping 172.22.38.45 interface gigabit ethernet 2/1
PING 172.22.38.45 (172.22.38.45) 56(84) bytes of data.
64 bytes from 172.22.38.45: icmp_seq=1 ttl=127 time=1.18 ms
64 bytes from 172.22.38.45: icmp_seq=2 ttl=127 time=0.483 ms
64 bytes from 172.22.38.45: icmp_seq=3 ttl=127 time=0.479 ms
--- 172.22.38.45 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2006ms
rtt min/avg/max/mdev = 0.479/0.716/1.186/0.332 ms
mds-switch-3#
```

- Step 4** Enable the iSCSI interface 2/1 on the switch mds-switch-3. Along with enabling the iSCSI interface, you can do additional iSCSI-related TCP tuning. In this recipe the default values are used.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# interface iscsi 2/1
mds-switch-3(config-if)# no shut
mds-switch-3(config-if)# end
mds-switch-3#
```

- Step 5** Configure the iSCSI initiator on the switch mds-switch-3. This configuration can be done in multiple ways. It can be done using the IP address of the initiator or using an IQN (iSCSI Qualified Name). This example uses an IQN name. Most iSCSI drivers or clients can automatically assign an IQN name on the host. The IQN name has to be at least 16 characters long. The IQN name can also be manually assigned. If you assign it manually, ensure that the IQN name is unique.

The Windows 2000 server used in the Microsoft driver, preconfigures the IQN name during installation. You can look up this name. In the iSCSI utility, select **Initiator Settings** to see the IQN name for the host. The driver assigned initiator node name is IQN.1991-05.com.microsoft:sjc7-pc-1 (see [Figure 10-3](#)).



Figure 10-3 iSCSI Initiator Properties

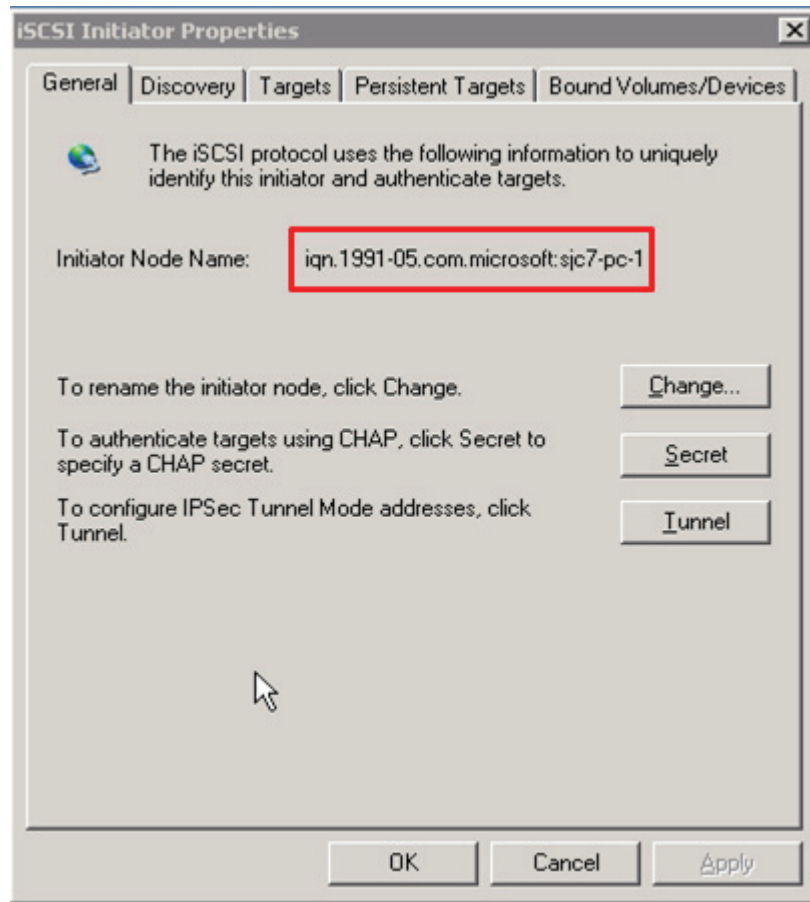


Figure 10-3 shows the IQN name in the iSCSI driver interface highlighted in red. For the Linux OS, this information is found in the `/etc/initiatorname.iscsi` file. You need this information to configure the iSCSI initiator on the switch.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# iscsi initiator name iqn.1991-05.com.microsoft:sjc7-pc-1
mds-switch-3(config-iscsi-init)# static pwwn system-assign 1 <-- system assigned
mds-switch-3(config-iscsi-init)# vsan 1 <-- Must be a member in the Targets VSAN
mds-switch-3(config-iscsi-init)# end
mds-switch-3#
```

**Tip**

If an iSCSI initiator needs to have the same pWWN previously used by a host, such as when you are converting a host from Fibre Channel to iSCSI, assign the pWWN manually with the commands shown. This alleviates the need to modify the LUN masking database on the array.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# iscsi initiator name iqn.1991-05.com.microsoft.sjc7-pc-1
mds-switch-3(config-iscsi-init)# static pwwn 21:05:00:0d:ec:02:2d:82 <-- manually assigned
mds-switch-3(config-iscsi-init)# vsan 1000 <-- Must be a member of the Target VSAN
mds-switch-3(config-iscsi-init)# end
mds-switch-3#
```

Here, the IQN assigned by the driver is used. If it needs to be changed, ensure that the new name is unique and is at least 16 characters long. Optionally, a pWWN can also be assigned to the initiator. The pWWN can be statically assigned by the administrator as shown previously or the system can automatically assign one. The initiator can be part of multiple VSANs. To talk to the target, it has to be a member of the target's VSAN. In the preceding example, the target belongs to VSAN 1000.

**Note**

Alternatively, you can use the IP address of the iSCSI initiator for configuration. Assigning a static pWWN is also an option. While zoning an iSCSI interface, you can use its IP address in place of its pWWN or IQN name.

An example of the using IP address is shown here.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# iscsi initiator ipaddress 172.22.38.45
mds-switch-3(config-iscsi-init)# static pwwn system-assign 1 <-- system assigned
mds-switch-3(config-iscsi-init)# vsan 1000 <-- Must be a member in the Targets VSAN
mds-switch-3(config-iscsi-init)# end
mds-switch-3#
```

**Note**

For the iSCSI initiator to communicate with a target port, the iSCSI initiator has to be a member of the target port's VSAN. The iSCSI initiators can be members of multiple VSANs.

**Step 6** Configure the virtual target on the switch mds-switch-3.

```
mds-switch-3# config t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# iscsi virtual-target name iscsi-sjc7-jbod-1
mds-switch-3(config-iscsi-tgt)# pwwn 22:00:00:20:37:5a:40:26
mds-switch-3(config-iscsi-tgt)# end
mds-switch-3#
```

The virtual target is a name assigned to the storage device. This name has to be 16 characters long. Then the pWWN of the storage port is assigned to the virtual target. This completes the configuration of a virtual target.

**Step 7** Permit the initiator to communicate with the virtual target already configured.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# iscsi virtual-target name iscsi-sjc7-jbod-1
mds-switch-3(config-iscsi-tgt)# initiator iqn.1991-05.com.microsoft.sjc7-pc-1 permit
mds-switch-3(config-iscsi-tgt)# end
mds-switch-3#
```

The virtual target can be configured to allow all initiators to communicate with it. In the preceding example, the virtual target is only configured to communicate with one initiator, iqn.1991-05.com.microsoft.sjc7-pc-1.

This permits the initiator iqn.1991-05.com.microsoft.sjc7-pc-1 to communicate with the virtual target iscsi-sjc7-jbod-1.

**Step 8** Create a zone with the initiator and the virtual target configured. Add the zone to the zone set and activate the zone set. Create a zone with the iSCSI initiator and the virtual targets as members. This enables the initiator and the target to communicate with each other. The zone can be created either with the IQN name or with the IP address or with the pWWN that was assigned to the initiator. In this recipe, it is created using the pWWN.

The following example shows zoning with the pWWN of the virtual target and initiator:

```

mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# zone name Z_iscsi_tst vsan 1000
mds-switch-3(config-zone)# mem pwwn 22:00:00:20:37:39:9c:1f
mds-switch-3(config-zone)# mem pwwn 21:05:00:0d:ec:02:2d:82
mds-switch-3(config-zone)#end
mds-switch-3#

mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# zoneset name ZS_ISCSI vsan 1000
mds-switch-3(config-zoneset)#member Z_iscsi_tst
mds-switch-3(config-zoneset)#end
mds-switch-3#

```

The following example shows zoning with the pWWN of the virtual target and the IQN of the iSCSI initiator:

```

mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# zone name Z_iscsi_tst vsan 1000
mds-switch-3(config-zone)# member pwwn 22:00:00:20:37:39:9c:1f
mds-switch-3(config-zone)# member symbolic-nodename iqn.1991-05.com.microsoft:sjc7-pc-1
mds-switch-3(config-zone)# end
mds-switch-3#

```

Activate the zone set to allow the zone members to communicate.

```

mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# zoneset activate name ZS_ISCSI vsan 1000
Zoneset activation initiated. check zone status
mds-switch-3#

```

## Configuring iSCSI on the MDS Switch in Proxy Initiator Mode

The recipe in this section details the proxy mode configuration for iSCSI on an MDS switch. In proxy initiator mode, one Fibre Channel initiator is used for all iSCSI clients that access the switch through the same iSCSI interface (iscsi3/3 for example). The initiators use the pWWN assigned to the iSCSI interface. The iSCSI interface to which an iSCSI client will log in is configured in the client and must be permitted by the virtual target configured for that initiator.

Proxy mode is advantageous over transparent mode when the configuration requires multiple iSCSI initiators to access the same Fibre Channel target. For example, if 20 iSCSI initiators need to communicate with a Fibre Channel disk, in transparent mode, 20 iSCSI initiators and 20 zones need to be created, and array based LUN masking has to be updated for all 20 initiator instances.

On the other hand, proxy initiator mode is easier to manage than transparent mode as it allows for centralized management of the iSCSI configuration, as all iSCSI clients accessing the same switch interface use the same iSCSI initiator.

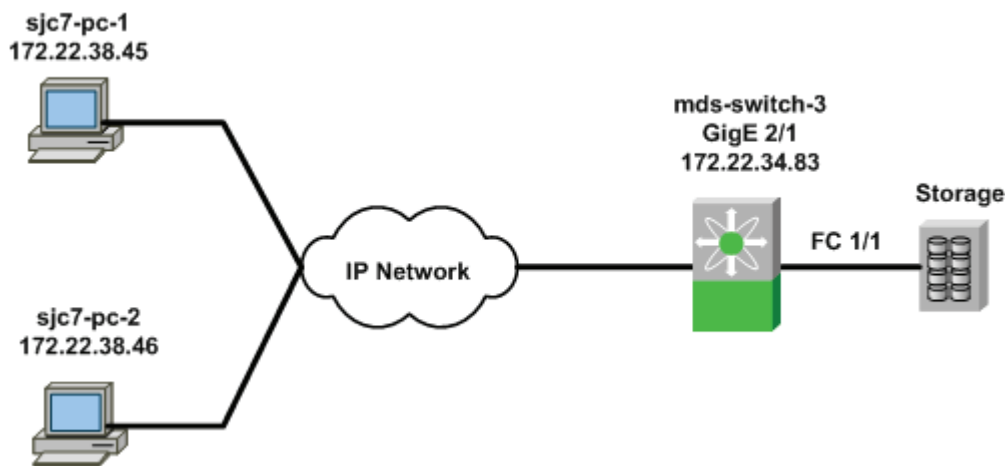
First, a pWWN is assigned to iSCSI interface. Then this pWWN is zoned with the Fibre Channel target so that the proxy initiator can see the LUNs presented by the virtual target. All the LUN masking and zoning are performed only with the proxy initiator. As new hosts (iSCSI clients) are added, they are exposed to only the LUNs they need to see as no new zones are needed and no modifications to the array's LUN masking need to be done.

A typical practice is to create a virtual target for each host and configure the virtual target to only expose the required LUNs to the iSCSI initiator.

The proxy initiator is not restricted to a single VSAN. As iSCSI clients are configured and given access to different VSANs, a proxy initiator is created in the new VSAN. The maximum number of initiators that need to be zoned would be the number of proxy initiators that have iSCSI clients in a particular VSAN. This is far fewer than under transparent mode, where a Fibre Channel initiator is created for every iSCSI client.

The topology used for the iSCSI proxy initiator recipe is shown in Figure 10-4. It has two Windows hosts on the same subnet. Both hosts' iSCSI interfaces are on the 172.22.38.0 network.

Figure 10-4 iSCSI Proxy Topology



To configure proxy mode for iSCSI on an MDS switch, follow these steps:

- Step 1** Configure the Gigabit Ethernet interface on the MDS switch. The Gigabit Ethernet interface on the MDS switch is given an IP address and a subnet mask. This allows the Gigabit Ethernet interface to communicate with the network.

```

mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# interface gigabitethernet 2/1
mds-switch-3(config-if)# ip address 172.22.34.83 255.255.254.0
mds-switch-3(config-if)# end
mds-switch-3#

```

- Step 2** Configure IP routes if required. In the recipe, the initiators sjc7-pc-1 and sjc7-pc-2 are in the 172.22.38.0 subnet, while the Gigabit Ethernet interface is in the 172.22.34.0 subnet. To allow the initiator and the Gigabit Ethernet port to communicate, an IP route has to be configured on the switch. A host-based route is configured to allow the Gigabit Ethernet port to communicate with sjc7-pc-1 and sjc7-pc-2 hosts.

```

mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# ip route 172.22.38.45 255.255.255.255 172.22.34.1 interface
gigabitethernet 2/1
mds-switch-3(config)# ip route 172.22.38.46 255.255.255.255 172.22.34.1 interface
gigabitethernet 2/1
mds-switch-3(config)# end
mds-switch-3#

```

**Step 3** Ping the Gigabit Ethernet interface from the hosts. Similarly ping the host from the switch.

```
mds-switch-3# ping 172.22.38.45 interface gigabit ethernet 2/1
PING 172.22.38.45 (172.22.38.45) 56(84) bytes of data.
64 bytes from 172.22.38.45: icmp_seq=1 ttl=127 time=1.18 ms
64 bytes from 172.22.38.45: icmp_seq=2 ttl=127 time=0.483 ms
64 bytes from 172.22.38.45: icmp_seq=3 ttl=127 time=0.479 ms
--- 172.22.38.45 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2006ms
rtt min/avg/max/mdev = 0.479/0.716/1.186/0.332 ms
mds-switch-3#

mds-switch-3# ping 172.22.38.46 interface gigabit ethernet 2/1
PING 172.22.38.46 (172.22.38.46) 56(84) bytes of data.
64 bytes from 172.22.38.46: icmp_seq=1 ttl=127 time=1.18 ms
64 bytes from 172.22.38.46: icmp_seq=2 ttl=127 time=0.483 ms
64 bytes from 172.22.38.46: icmp_seq=3 ttl=127 time=0.479 ms
--- 172.22.38.46 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2006ms
rtt min/avg/max/mdev = 0.479/0.716/1.186/0.332 ms
mds-switch-3#
```

**Step 4** Enable the iSCSI interface on the switch mds-switch-3. The iSCSI interface 2/1 (which is the same port as the Gigabit Ethernet interface) needs to be enabled. You can do additional iSCSI-related TCP tuning. In this recipe default values are used. The **switchport** command is used to enable proxy-initiator mode for the iSCSI interface 2/1. It is also used to assign a pWWN to the interface.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# interface iscsi 2/1
mds-switch-3(config-if)# switchport proxy-initiator nWWN 21:05:00:0d:ec:02:2d:82 pwwn
21:05:00:0d:ec:02:2d:82
mds-switch-3(config-if)# no shut
mds-switch-3(config-if)# end
mds-switch-3#
```

**Step 5** Add the iSCSI interface to the required VSANs. Do this to allow the iSCSI interface to communicate with the virtual target to see the LUNs in different VSANs. The commands add the iSCSI interface 2/1 into the 1000VSAN. This is the VSAN to which the Fibre Channel target is connected. Once this is complete, the interface iSCSI 2/1 can log on to the fabric.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# iscsi interface vsan-membership
mds-switch-3(config)# vsan database
mds-switch-3(config-vsan-db)# vsan 1000 interface iscsi 2/1
mds-switch-3(config-vsan-db)#end
mds-switch-3#
```



**Note**

The **iscsi interface vsan-membership** command is required to make the iSCSI interface part of multiple VSANs.

**Step 6** Configure a virtual target on the switch mds-switch-3.

```
mds-switch-3# config t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# iscsi virtual-target name sjc7-jbod-iscsi-1
mds-switch-3(config-iscsi-tgt)# pWWN 22:00:00:20:37:39:9c:1f
mds-switch-3(config-iscsi-tgt)# end
mds-switch-3#
```

The virtual target is a name assigned to the storage device. This name has to be 16 characters long. Then the pWWN of the storage port is assigned to the virtual target as shown previously. This completes the configuration of a virtual target.

- Step 7** Create a zone with the iSCSI initiator and virtual targets as members. This enables the initiator and the target to communicate with each other. The zone can be created either with the IQN name, IP address, or pWWN assigned to the initiator. In this recipe, it is created using the pWWN. Add the zone to the zone set.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# zone name Z_iscsi_tst vsan 1000
mds-switch-3(config-zone)# mem pwn 22:00:00:20:37:39:9c:1f
mds-switch-3(config-zone)# mem pwn 21:05:00:0d:ec:02:2d:82
mds-switch-3(config-zone)#end
mds-switch-3#

mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# zoneset name ZS_ISCSI vsan 1000
mds-switch-3(config-zoneset)#member Z_iscsi_tst
mds-switch-3(config-zoneset)#end
mds-switch-3#
```

Activate the zone set to allow the zone members to communicate.

```
mds-switch-3# conf t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# zoneset activate name ZS_ISCSI vsan 1000
Zoneset activation initiated. check zone status
mds-switch-3#
mds-switch-3# show zoneset active vsan 1000
zoneset name ZS_iscsi vsan 1000
 zone name Z_iscsi_tst vsan 1000
 * fcid 0xd90002 [pwn 21:05:00:0d:ec:02:2d:82]
 * fcid 0xd90000 [pwn 22:00:00:20:37:39:9c:1f]
mds-switch-3#
```



**Tip**

To achieve LUN security, create a virtual target with access to specific LUNs for each initiator.

- Step 8** Configure a virtual target for each initiator and configure LUN masking for the initiator. When the zone is successfully activated, the LUNs available on the storage port are visible to the iSCSI interface. As this interface could be a proxy iSCSI interface for many iSCSI initiators, some form of LUN security must be enabled. This recipe creates a virtual target with access to specific LUNs for each initiator. The iSCSI interface can see 10 LUNs (LUN 11 to LUN 20 in decimal). The configuration allows the host sjc7-pc-1 to see LUNs 11 - 14 (decimal) on the array.

```
mds-switch-3# config t
Enter configuration commands, one per line. End with CNTL/Z.
mds-switch-3(config)# iscsi virtual-target name sjc7-jbod-iscsi-1
mds-switch-3(config-(iscsi-tgt))# pwn 22:00:00:20:37:39:9c:1f fc-lun b iscsi-lun 1
mds-switch-3(config-(iscsi-tgt))# pwn 22:00:00:20:37:39:9c:1f fc-lun b iscsi-lun 2
mds-switch-3(config-(iscsi-tgt))# pwn 22:00:00:20:37:39:9c:1f fc-lun d iscsi-lun 3
mds-switch-3(config-(iscsi-tgt))# pwn 22:00:00:20:37:39:9c:1f fc-lun e iscsi-lun 4
mds-switch-3(config-(iscsi-tgt))# initiator ip address 172.22.38.45 permit
mds-switch-3(config-(iscsi-tgt))# end
mds-switch-3#
```

Allow the host sjc7-pc-2 to see LUNs 16 - 20 (decimal).

```
mds-switch-3# config t
```

Enter configuration commands, one per line. End with CNTL/Z.

```
mds-switch-3(config)# iscsi virtual-target name sjc7-jbod-iscsi-1
mds-switch-3(config-iscsi-tgt)# pwwN 22:00:00:20:37:39:9c:1f fc-lun 10 iscsi-lun 1
mds-switch-3(config-iscsi-tgt)# pwwN 22:00:00:20:37:39:9c:1f fc-lun 11 iscsi-lun 2
mds-switch-3(config-iscsi-tgt)# pwwN 22:00:00:20:37:39:9c:1f fc-lun 12 iscsi-lun 3
mds-switch-3(config-iscsi-tgt)# pwwN 22:00:00:20:37:39:9c:1f fc-lun 13 iscsi-lun 4
mds-switch-3(config-iscsi-tgt)# pwwN 22:00:00:20:37:39:9c:1f fc-lun 14 iscsi-lun 5
mds-switch-3(config-iscsi-tgt)# initiator ip address 172.22.38.46 permit
mds-switch-3(config-iscsi-tgt)# end
mds-switch-3#
```

After these changes, both hosts are able to see the LUNs allocated to them through the virtual target created for each. There is no need to create additional zones when new iSCSI clients are added. If the iSCSI clients need access, zone additional targets to the iSCSI interfaces as shown in [Step 7](#) of [Configuring iSCSI on the MDS Switch in Proxy Initiator Mode](#), page 10-11.

## Configuring iSCSI Client Initiators on Hosts

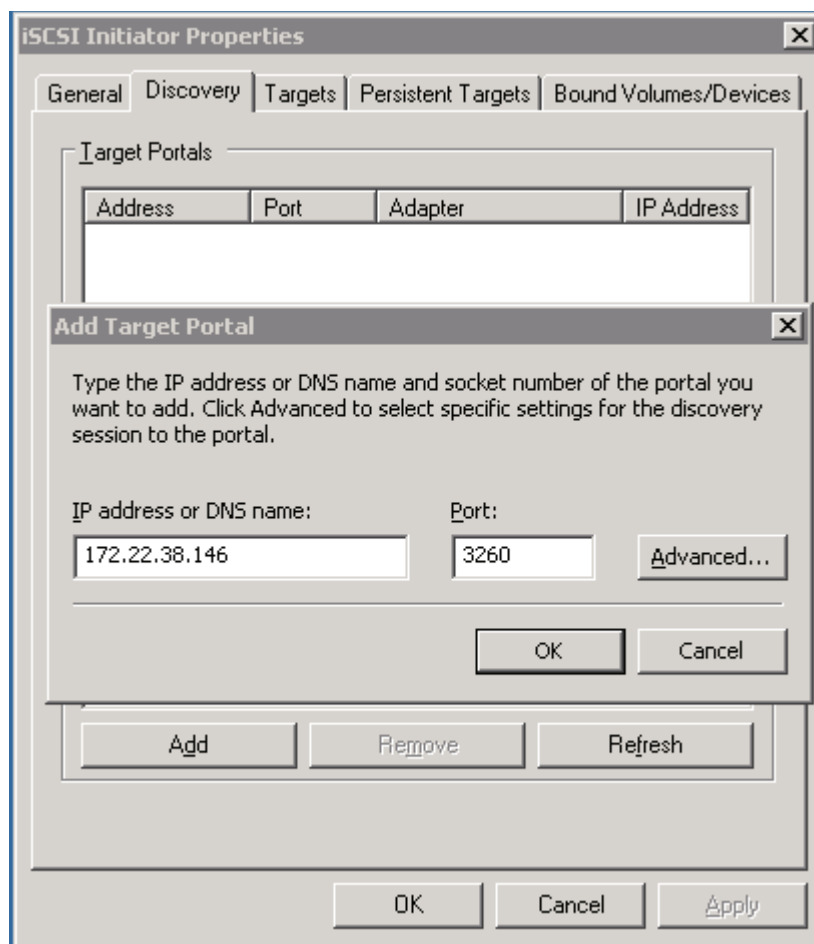
### Configuring iSCSI on Microsoft Windows

This section details configuration of a Microsoft Windows iSCSI driver 2.0 configuration. The example assumes that the iSCSI driver has already been installed on the Windows server.

To add a target portal to the iSCSI driver, follow these steps:

- 
- Step 1** In the iSCSI initiators properties, click the **Discovery** tab (see [Figure 10-5](#)).

Figure 10-5 Add Target Portal in iSCSI Driver

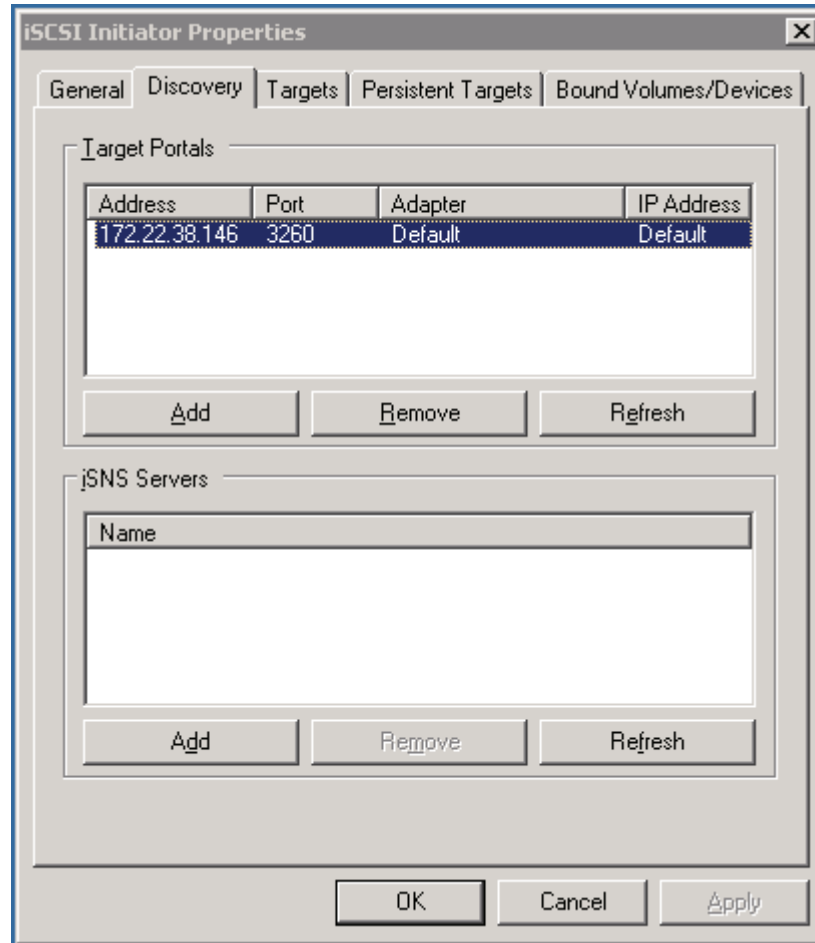


- Step 2** Click **Add**. You see the Add Target Portal dialog box shown in [Figure 10-5](#).
- Step 3** Provide the IP address of the Gigabit Ethernet interface on the switch to which the target is attached to the VR address of a VRRP group, which is 172.22.38.146. Click **OK**.

This adds the target to the iSCSI client. See [Figure 10-6](#).

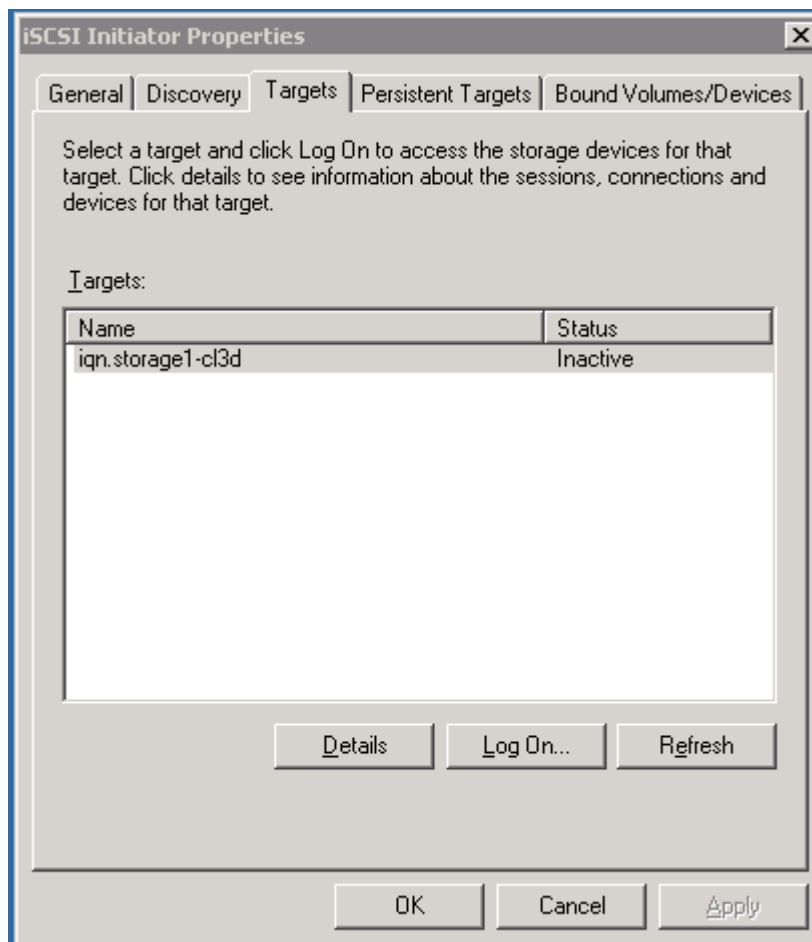


Figure 10-6 iSCSI Target Portal Address Configuration



**Step 4** Select the **Targets** tab in the iSCSI initiator properties (see [Figure 10-7](#)).

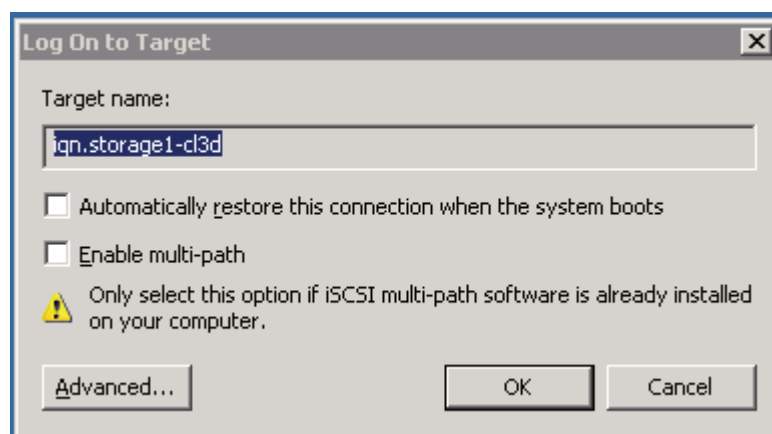
Figure 10-7 Select the iSCSI Target in the Targets Tab



If the virtual target is properly configured and the initiator is permitted to access the virtual target, the configured virtual target is visible under Targets and its status is **Inactive**, as shown in Figure 10-7,

**Step 5** Click **Log On**. A logon options dialog box appears. See Figure 10-8.

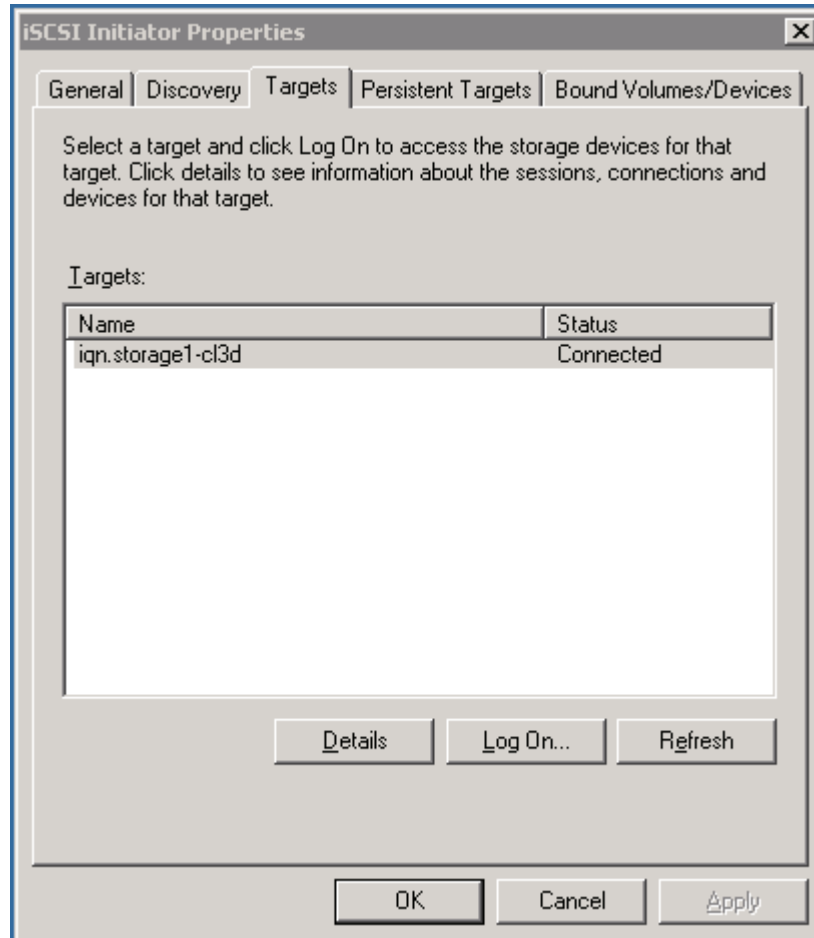
Figure 10-8 Log on to Target



In this dialog box, you can select the automatic reconnect option. If you do not select this option, this operation has to be performed every time the server reboots.

- Step 6** Click **OK** to start the iSCSI login and storage discovery process for the target listed in the window. An iSCSI initiator can see multiple targets.

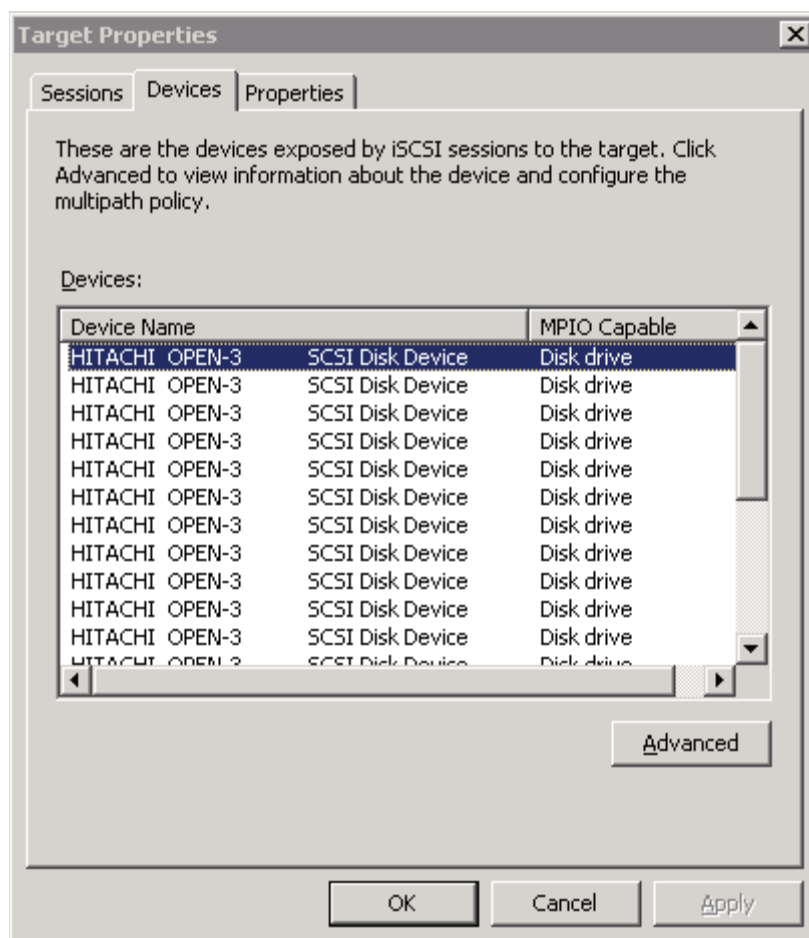
Figure 10-9 Target Logged on Status



If the logon was successful, the status of the target changes from **inactive** to **connected** (see [Figure 10-9](#)). Now the host sees storage through the iSCSI initiator.

- Step 7** Click **Details** in the **Targets** tab and select the **Device** tab in the resulting popup to see the LUNs presented to the initiator. See [Figure 10-10](#).

Figure 10-10 Disk Available to the Initiator on the Target



- Step 8** From Microsoft windows, scan for the new disk and initialize it by selecting **Start > Settings > Control Panel > Administrative Tools > Computer management > Disk Management**.