Catalyst 3550 Multilayer Switch Software Configuration Guide

Cisco IOS Release 12.2(44)SE
February 2008

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-8565-02

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.

CCDE, CCVP, Cisco Eos, Cisco StadiumVision, the Cisco logo, DCE, and Welcome to the Human Network are trademarks; Changing the Way We Work, Live, Play, and Learn is a service mark; and Access Registrar, Aironet, AsyncOS, Bringing the Meeting To You, 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, Collaboration Without Limitation, Enterprise/Solver, EtherChannel, EtherFast, EtherSwitch, Event Center, Fast Step, Follow Me Browsing, FormShare, GigaDrive, HomeLink, Internet Quotient, IOS, iPhone, IP/TV, iQ Expertise, the iQ logo, iQ Net Readiness Scorecard, iQuick Study, IronPort, the IronPort logo, LightStream, Linksys, MediaTone, MeetingPlace, MGX, Networkers, Networking Academy, Network Registrar, PCNow, PIX, PowerPanels, ProConnect, ScriptShare, SenderBase, SMARTnet, Spectrum Expert, StackWise, The Fastest Way to Increase Your Internet Quotient, TransPath, WebEx, and the WebEx logo are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All 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. (0801R)

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

Catalyst 3550 Multilayer Switch Software Configuration Guide
Copyright © 2006–2008 Cisco Systems, Inc. All rights reserved.

CONTENTS

Preface xxxvii

Audience xxxvii

Purpose xxxvii

Conventions xxxviii

Related Publications xxxix

Obtaining Documentation and Submitting a Service Request xxxix

CHAPTER 1 **Overview** 1-1

Features 1-1

Ease of Deployment and Ease of Use 1-1

Performance 1-2

Manageability 1-2

Redundancy 1-3

VLAN Support 1-4

Security 1-5

Quality of Service (QoS) and Class of Service (CoS) 1-6

Layer 3 Support 1-7

Monitoring 1-8

Power over Ethernet Support for the Catalyst 3550-24PWR Switch 1-8

Management Options 1-9

Management Interface Options 1-9

Advantages of Using Network Assistant and Clustering Switches 1-10

Network Configuration Examples 1-10

Design Concepts for Using the Switch 1-11

Small to Medium-Sized Network Using Mixed Switches 1-14

Large Network Using Only Catalyst 3550 Switches 1-15

Multidwelling Network Using Catalyst 3550 Switches 1-17

Long-Distance, High-Bandwidth Transport Configuration 1-18

Where to Go Next 1-19

CHAPTER 2 **Using the Command-Line Interface** 2-1

Cisco IOS Command Modes 2-1

Getting Help 2-3

Abbreviating Commands 2-4

Using no and default Forms of Commands 2-4
Understanding CLI Messages 2-5
Using Configuration Logging 2-5
Using Command History 2-6
Changing the Command History Buffer Size 2-6
Recalling Commands 2-6
Disabling the Command History Feature 2-7
Using Editing Features 2-7
Enabling and Disabling Editing Features 2-7
Editing Commands through Keystrokes 2-7
Editing Command Lines that Wrap 2-9
Searching and Filtering Output of show and more Commands 2-10
Accessing the CLI 2-10

CHAPTER 3 **Assigning the Switch IP Address and Default Gateway** 3-1
Understanding the Boot Process 3-1
Assigning Switch Information 3-2
Default Switch Information 3-3
Understanding DHCP-Based Autoconfiguration 3-3
DHCP Client Request Process 3-4
Configuring DHCP-Based Autoconfiguration 3-5
DHCP Server Configuration Guidelines 3-5
Configuring the TFTP Server 3-6
Configuring the DNS 3-6
Configuring the Relay Device 3-6
Obtaining Configuration Files 3-7
Example Configuration 3-8
Manually Assigning IP Information 3-10
Checking and Saving the Running Configuration 3-10
Modifying the Startup Configuration 3-11
Default Boot Configuration 3-11
Automatically Downloading a Configuration File 3-11
Specifying the Filename to Read and Write the System Configuration 3-11
Booting Manually 3-12
Booting a Specific Software Image 3-13
Controlling Environment Variables 3-13

Scheduling a Reload of the Software Image 3-16
Configuring a Scheduled Reload 3-16
Displaying Scheduled Reload Information 3-17

CHAPTER 4 **Configuring Cisco IOS CNS Agents 4-1**
Understanding Cisco Configuration Engine Software 4-1
Configuration Service 4-2
Event Service 4-3
NameSpace Mapper 4-3
What You Should Know About the CNS IDs and Device Hostnames 4-3
ConfigID 4-3
DeviceID 4-4
Hostname and DeviceID 4-4
Using Hostname, DeviceID, and ConfigID 4-4
Understanding Cisco IOS Agents 4-5
Initial Configuration 4-5
Incremental (Partial) Configuration 4-6
Synchronized Configuration 4-6
Configuring Cisco IOS Agents 4-6
Enabling Automated CNS Configuration 4-6
Enabling the CNS Event Agent 4-7
Enabling the Cisco IOS CNS Agent 4-9
Enabling an Initial Configuration 4-9
Enabling a Partial Configuration 4-13
Upgrading Devices with Cisco IOS Image Agent 4-14
Prerequisites for the CNS Image Agent 4-14
Restrictions for the CNS Image Agent 4-14
Displaying CNS Configuration 4-15

CHAPTER 5 **Clustering Switches 5-1**
Understanding Switch Clusters 5-1
Cluster Command Switch Characteristics 5-3
Standby Cluster Command Switch Characteristics 5-3
Candidate Switch and Member Switch Characteristics 5-3
Planning a Switch Cluster 5-4
Automatic Discovery of Cluster Candidates and Members 5-4
Discovery Through CDP Hops 5-5
Discovery Through Non-CDP-Capable and Noncluster-Capable Devices 5-6
Discovery Through Different VLANs 5-6

Discovery Through Different Management VLANs 5-7
Discovery Through Routed Ports 5-8
Discovery of Newly Installed Switches 5-9
HSRP and Standby Cluster Command Switches 5-10
Virtual IP Addresses 5-10
Other Considerations for Cluster Standby Groups 5-11
Automatic Recovery of Cluster Configuration 5-12
IP Addresses 5-13
Hostnames 5-13
Passwords 5-13
SNMP Community Strings 5-14
TACACS+ and RADIUS 5-14
Catalyst 1900 and Catalyst 2820 CLI Considerations 5-14
Using SNMP to Manage Switch Clusters 5-15

CHAPTER 6 **Administering the Switch** 6-1
Managing the System Time and Date 6-1
Understanding the System Clock 6-1
Understanding Network Time Protocol 6-2
Configuring NTP 6-3
Default NTP Configuration 6-4
Configuring NTP Authentication 6-4
Configuring NTP Associations 6-5
Configuring NTP Broadcast Service 6-6
Configuring NTP Access Restrictions 6-8
Configuring the Source IP Address for NTP Packets 6-10
Displaying the NTP Configuration 6-11
Configuring Time and Date Manually 6-11
Setting the System Clock 6-11
Displaying the Time and Date Configuration 6-12
Configuring the Time Zone 6-12
Configuring Summer Time (Daylight Saving Time) 6-13
Configuring a System Name and Prompt 6-14
Default System Name and Prompt Configuration 6-15
Configuring a System Name 6-15
Understanding DNS 6-15
Default DNS Configuration 6-16
Setting Up DNS 6-16
Displaying the DNS Configuration 6-17

Creating a Banner 6-17
Default Banner Configuration 6-17
Configuring a Message-of-the-Day Login Banner 6-18
Configuring a Login Banner 6-18
Managing the MAC Address Table 6-19
Building the Address Table 6-20
MAC Addresses and VLANs 6-20
Default MAC Address Table Configuration 6-20
Changing the Address Aging Time 6-21
Removing Dynamic Address Entries 6-21
Configuring MAC Address Notification Traps 6-21
Adding and Removing Static Address Entries 6-23
Configuring Unicast MAC Address Filtering 6-24
Displaying Address Table Entries 6-26
Optimizing System Resources for User-Selected Features 6-26
Using the Templates 6-28
Managing the ARP Table 6-29

CHAPTER 7 **Configuring Switch-Based Authentication** 7-1
Preventing Unauthorized Access to Your Switch 7-1
Protecting Access to Privileged EXEC Commands 7-2
Default Password and Privilege Level Configuration 7-2
Setting or Changing a Static Enable Password 7-3
Protecting Enable and Enable Secret Passwords with Encryption 7-3
Disabling Password Recovery 7-5
Setting a Telnet Password for a Terminal Line 7-6
Configuring Username and Password Pairs 7-6
Configuring Multiple Privilege Levels 7-7
Setting the Privilege Level for a Command 7-8
Changing the Default Privilege Level for Lines 7-9
Logging into and Exiting a Privilege Level 7-9
Controlling Switch Access with TACACS+ 7-9
Understanding TACACS+ 7-10
TACACS+ Operation 7-12
Configuring TACACS+ 7-12
Default TACACS+ Configuration 7-13
Identifying the TACACS+ Server Host and Setting the Authentication Key 7-13
Configuring TACACS+ Login Authentication 7-14

Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services 7-16
Starting TACACS+ Accounting 7-17
Displaying the TACACS+ Configuration 7-17
Controlling Switch Access with RADIUS 7-17
Understanding RADIUS 7-18
RADIUS Operation 7-19
Configuring RADIUS 7-19
Default RADIUS Configuration 7-20
Identifying the RADIUS Server Host 7-20
Configuring RADIUS Login Authentication 7-23
Defining AAA Server Groups 7-25
Configuring RADIUS Authorization for User Privileged Access and Network Services 7-27
Starting RADIUS Accounting 7-28
Configuring Settings for All RADIUS Servers 7-29
Configuring the Switch to Use Vendor-Specific RADIUS Attributes 7-29
Configuring the Switch for Vendor-Proprietary RADIUS Server Communication 7-31
Displaying the RADIUS Configuration 7-31
Controlling Switch Access with Kerberos 7-32
Understanding Kerberos 7-32
Kerberos Operation 7-34
Authenticating to a Boundary Switch 7-34
Obtaining a TGT from a KDC 7-35
Authenticating to Network Services 7-35
Configuring Kerberos 7-35
Configuring the Switch for Local Authentication and Authorization 7-36
Configuring the Switch for Secure Shell 7-37
Understanding SSH 7-38
SSH Servers, Integrated Clients, and Supported Versions 7-38
Limitations 7-38
Configuring SSH 7-39
Configuration Guidelines 7-39
Setting Up the Switch to Run SSH 7-39
Configuring the SSH Server 7-40
Displaying the SSH Configuration and Status 7-41
Configuring the Switch for Secure Socket Layer HTTP 7-41
Understanding Secure HTTP Servers and Clients 7-42
Certificate Authority Trustpoints 7-42
CipherSuites 7-43

Configuring Secure HTTP Servers and Clients 7-44
Default SSL Configuration 7-44
SSL Configuration Guidelines 7-44
Configuring a CA Trustpoint 7-44
Configuring the Secure HTTP Server 7-45
Configuring the Secure HTTP Client 7-47
Displaying Secure HTTP Server and Client Status 7-48
Configuring the Switch for Secure Copy Protocol 7-48

CHAPTER 8 **Configuring IEEE 802.1x Port-Based Authentication** 8-1
Understanding IEEE 802.1x Port-Based Authentication 8-1
Device Roles 8-2
Authentication Process 8-3
Authentication Initiation and Message Exchange 8-5
Ports in Authorized and Unauthorized States 8-7
IEEE 802.1x Host Mode 8-7
IEEE 802.1x Accounting 8-8
IEEE 802.1x Accounting Attribute-Value Pairs 8-8
Using IEEE 802.1x Authentication with VLAN Assignment 8-9
Using IEEE 802.1x Authentication with Per-User ACLs 8-10
Using IEEE 802.1x Authentication with Guest VLAN 8-11
Using IEEE 802.1x Authentication with Restricted VLAN 8-13
Using IEEE 802.1x Authentication with Inaccessible Authentication Bypass 8-13
Using IEEE 802.1x Authentication with Voice VLAN Ports 8-15
Using IEEE 802.1x Authentication with Port Security 8-15
Using IEEE 802.1x Authentication with Wake-on-LAN 8-16
Using IEEE 802.1x Authentication with MAC Authentication Bypass 8-17
Network Admission Control Layer 2 IEEE 802.1x Validation 8-18
Web Authentication with Automatic MAC Check 8-18
Configuring IEEE 802.1x Authentication 8-19
Default IEEE 802.1x Authentication Configuration 8-20
IEEE 802.1x Authentication Configuration Guidelines 8-21
IEEE 802.1x Authentication 8-21
VLAN Assignment, Guest VLAN, Restricted VLAN, and Inaccessible Authentication Bypass 8-22
MAC Authentication Bypass 8-23
Upgrading from a Previous Software Release 8-23
Configuring IEEE 802.1x Authentication 8-24
Configuring the Switch-to-RADIUS-Server Communication 8-25

Configuring the Host Mode 8-27
Enabling Periodic Re-Authentication 8-27
Manually Re-Authenticating a Client Connected to a Port 8-28
Changing the Quiet Period 8-28
Changing the Switch-to-Client Retransmission Time 8-29
Setting the Switch-to-Client Frame-Retransmission Number 8-30
Setting the Re-Authentication Number 8-30
Configuring IEEE 802.1x Accounting 8-31
Configuring a Guest VLAN 8-32
Configuring a Restricted VLAN 8-33
Configuring the Inaccessible Authentication Bypass Feature 8-34
Configuring IEEE 802.1x Authentication with WoL 8-37
Configuring MAC Authentication Bypass 8-37
Configuring NAC Layer 2 IEEE 802.1x Validation 8-38
Configuring Web Authentication 8-39
Disabling IEEE 802.1x on the Port 8-41
Resetting the IEEE 802.1x Configuration to the Default Values 8-42
Displaying IEEE 802.1x Statistics and Status 8-42

CHAPTER 9 **Configuring Interface Characteristics** 9-1
Understanding Interface Types 9-1
Port-Based VLANs 9-2
Switch Ports 9-2
Access Ports 9-2
Trunk Ports 9-3
Tunnel Ports 9-3
Switch Virtual Interfaces 9-4
Routed Ports 9-4
EtherChannel Port Groups 9-5
Power Over Ethernet Ports 9-5
Supported Protocols and Standards 9-6
Powered-Device Detection and Initial Power Allocation 9-6
Power Management Modes 9-7
Connecting Interfaces 9-7
Using the Interface Command 9-9
Procedures for Configuring Interfaces 9-9
Configuring a Range of Interfaces 9-10
Configuring and Using Interface Range Macros 9-12
Configuring Ethernet Interfaces 9-13

Default Ethernet Interface Configuration 9-14
Configuring Interface Speed and Duplex Mode 9-15
Configuration Guidelines 9-15
Setting the Interface Speed and Duplex Parameters 9-16
Configuring Power over Ethernet on the Catalyst 3550-24PWR Ports 9-16
Configuring IEEE 802.3x Flow Control 9-17
Adding a Description for an Interface 9-19
Configuring Layer 3 Interfaces 9-19
Monitoring and Maintaining the Interfaces 9-21
Monitoring Interface and Controller Status 9-21
Clearing and Resetting Interfaces and Counters 9-22
Shutting Down and Restarting the Interface 9-22

CHAPTER 10 **Configuring Smartports Macros** 10-1
Understanding Smartports Macros 10-1
Configuring Smartports Macros 10-2
Default Smartports Macro Configuration 10-2
Smartports Macro Configuration Guidelines 10-3
Creating Smartports Macros 10-4
Applying Smartports Macros 10-4
Applying Cisco-Default Smartports Macros 10-6
Displaying Smartports Macros 10-8

CHAPTER 11 **Configuring VLANs** 11-1
Understanding VLANs 11-1
Supported VLANs 11-2
VLAN Port Membership Modes 11-3
Configuring Normal-Range VLANs 11-4
Token Ring VLANs 11-5
Normal-Range VLAN Configuration Guidelines 11-5
VLAN Configuration Mode Options 11-6
VLAN Configuration in config-vlan Mode 11-6
VLAN Configuration in VLAN Configuration Mode 11-6
Saving VLAN Configuration 11-7
Default Ethernet VLAN Configuration 11-7
Creating or Modifying an Ethernet VLAN 11-8
Deleting a VLAN 11-10
Assigning Static-Access Ports to a VLAN 11-10

Configuring Extended-Range VLANs 11-11
Default VLAN Configuration 11-12
Extended-Range VLAN Configuration Guidelines 11-12
Creating an Extended-Range VLAN 11-13
Creating an Extended-Range VLAN with an Internal VLAN ID 11-14
Displaying VLANs 11-14
Configuring VLAN Trunks 11-15
Trunking Overview 11-15
Encapsulation Types 11-17
IEEE 802.1Q Configuration Considerations 11-18
Default Layer 2 Ethernet Interface VLAN Configuration 11-18
Configuring an Ethernet Interface as a Trunk Port 11-18
Interaction with Other Features 11-19
Configuring a Trunk Port 11-20
Defining the Allowed VLANs on a Trunk 11-21
Changing the Pruning-Eligible List 11-22
Configuring the Native VLAN for Untagged Traffic 11-22
Load Sharing Using STP 11-23
Load Sharing Using STP Port Priorities 11-23
Load Sharing Using STP Path Cost 11-25
Configuring VMPS 11-26
Understanding VMPS 11-27
Dynamic Port VLAN Membership 11-27
VMPS Database Configuration File 11-28
Default VMPS Client Configuration 11-28
VMPS Configuration Guidelines 11-28
Configuring the VMPS Client 11-29
Entering the IP Address of the VMPS 11-29
Configuring Dynamic Access Ports on VMPS Clients 11-30
Reconfirming VLAN Memberships 11-31
Changing the Reconfirmation Interval 11-31
Changing the Retry Count 11-31
Monitoring the VMPS 11-32
Troubleshooting Dynamic Port VLAN Membership 11-32
VMPS Configuration Example 11-33

CHAPTER 12 **Configuring VTP** 12-1

Understanding VTP 12-1

The VTP Domain 12-2

VTP Modes 12-2

VTP Advertisements 12-3

VTP Version 2 12-4

VTP Pruning 12-4

Configuring VTP 12-6

Default VTP Configuration 12-6

VTP Configuration Options 12-7

VTP Configuration in Global Configuration Mode 12-7

VTP Configuration in VLAN Configuration Mode 12-7

VTP Configuration Guidelines 12-8

Domain Names 12-8

Passwords 12-8

VTP Version 12-8

Configuration Requirements 12-9

Configuring a VTP Server 12-9

Configuring a VTP Client 12-10

Disabling VTP (VTP Transparent Mode) 12-12

Enabling VTP Version 2 12-12

Enabling VTP Pruning 12-13

Adding a VTP Client Switch to a VTP Domain 12-14

Monitoring VTP 12-15

CHAPTER 13 **Configuring Voice VLAN** 13-1

Understanding Voice VLAN 13-1

Configuring Voice VLAN 13-2

Default Voice VLAN Configuration 13-2

Voice VLAN Configuration Guidelines 13-3

Configuring a Port to Connect to a Cisco 7960 IP Phone 13-3

Configuring Ports to Carry Voice Traffic in IEEE 802.1Q Frames 13-4

Configuring Ports to Carry Voice Traffic in IEEE 802.1p Priority-Tagged Frames 13-4

Overriding the CoS Priority of Incoming Data Frames 13-5

Configuring the IP Phone to Trust the CoS Priority of Incoming Data Frames 13-5

Displaying Voice VLAN 13-6

CHAPTER 14 **Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling** 14-1

Understanding IEEE 802.1Q Tunneling 14-1

Configuring IEEE 802.1Q Tunneling 14-4

Default IEEE 802.1Q Tunneling Configuration 14-4

IEEE 802.1Q Tunneling Configuration Guidelines 14-4

Native VLANs 14-4

System MTU 14-5

IEEE 802.1Q Tunneling and Other Features 14-5

Configuring an IEEE 802.1Q Tunneling Port 14-6

Understanding Layer 2 Protocol Tunneling 14-7

Configuring Layer 2 Protocol Tunneling 14-9

Default Layer 2 Protocol Tunneling Configuration 14-10

Layer 2 Protocol Tunneling Configuration Guidelines 14-11

Configuring Layer 2 Tunneling 14-12

Configuring Layer 2 Tunneling for EtherChannels 14-13

Configuring the SP Edge Switch 14-13

Configuring the Customer Switch 14-15

Monitoring and Maintaining Tunneling Status 14-17

CHAPTER 15 **Configuring STP** 15-1

Understanding Spanning-Tree Features 15-1

STP Overview 15-2

Spanning-Tree Topology and BPDUs 15-3

Bridge ID, Switch Priority, and Extended System ID 15-4

Spanning-Tree Interface States 15-4

Blocking State 15-5

Listening State 15-6

Learning State 15-6

Forwarding State 15-6

Disabled State 15-7

How a Switch or Port Becomes the Root Switch or Root Port 15-7

Spanning Tree and Redundant Connectivity 15-8

Spanning-Tree Address Management 15-8

Accelerated Aging to Retain Connectivity 15-8

Spanning-Tree Modes and Protocols 15-9

Supported Spanning-Tree Instances 15-9

Spanning-Tree Interoperability and Backward Compatibility 15-10

STP and IEEE 802.1Q Trunks 15-10

VLAN-Bridge Spanning Tree 15-10

Configuring Spanning-Tree Features 15-11
Default Spanning-Tree Configuration 15-11
Spanning-Tree Configuration Guidelines 15-12
Changing the Spanning-Tree Mode 15-13
Disabling Spanning Tree 15-14
Configuring the Root Switch 15-15
Configuring a Secondary Root Switch 15-16
Configuring the Port Priority 15-17
Configuring the Path Cost 15-18
Configuring the Switch Priority of a VLAN 15-19
Configuring Spanning-Tree Timers 15-20
Configuring the Hello Time 15-21
Configuring the Forwarding-Delay Time for a VLAN 15-21
Configuring the Maximum-Aging Time for a VLAN 15-22
Configuring Spanning Tree for Use in a Cascaded Stack 15-22
Configuring the Transmit Hold Count 15-23
Displaying the Spanning-Tree Status 15-24

CHAPTER 16 **Configuring MSTP** 16-1
Understanding MSTP 16-2
Multiple Spanning-Tree Regions 16-2
IST, CIST, and CST 16-2
Operations Within an MST Region 16-3
Operations Between MST Regions 16-3
IEEE 802.1s Terminology 16-5
Hop Count 16-5
Boundary Ports 16-6
IEEE 802.1s Implementation 16-6
Port Role Naming Change 16-6
Interoperation Between Legacy and Standard Switches 16-7
Detecting Unidirectional Link Failure 16-7
Interoperability with IEEE 802.1D STP 16-8
Understanding RSTP 16-8
Port Roles and the Active Topology 16-9
Rapid Convergence 16-10
Synchronization of Port Roles 16-11

Bridge Protocol Data Unit Format and Processing 16-12
Processing Superior BPDU Information 16-13
Processing Inferior BPDU Information 16-13
Topology Changes 16-13
Configuring MSTP Features 16-14
Default MSTP Configuration 16-14
MSTP Configuration Guidelines 16-15
Specifying the MST Region Configuration and Enabling MSTP 16-16
Configuring the Root Switch 16-17
Configuring a Secondary Root Switch 16-19
Configuring the Port Priority 16-20
Configuring the Path Cost 16-21
Configuring the Switch Priority 16-21
Configuring the Hello Time 16-22
Configuring the Forwarding-Delay Time 16-23
Configuring the Maximum-Aging Time 16-23
Configuring the Maximum-Hop Count 16-24
Specifying the Link Type to Ensure Rapid Transitions 16-24
Designating the Neighbor Type 16-25
Restarting the Protocol Migration Process 16-26
Displaying the MST Configuration and Status 16-26

CHAPTER 17 **Configuring Optional Spanning-Tree Features** 17-1
Understanding Optional Spanning-Tree Features 17-1
Understanding Port Fast 17-2
Understanding BPDU Guard 17-2
Understanding BPDU Filtering 17-3
Understanding UplinkFast 17-3
Understanding Cross-Stack UplinkFast 17-5
How CSUF Works 17-6
Events that Cause Fast Convergence 17-7
Limitations 17-8
Connecting the Stack Ports 17-8
Understanding BackboneFast 17-9
Understanding EtherChannel Guard 17-11
Understanding Root Guard 17-12
Understanding Loop Guard 17-13

Configuring Optional Spanning-Tree Features 17-13
Default Optional Spanning-Tree Configuration 17-14
Optional Spanning-Tree Configuration Guidelines 17-14
Enabling Port Fast 17-14
Enabling BPDU Guard 17-15
Enabling BPDU Filtering 17-16
Enabling UplinkFast for Use with Redundant Links 17-17
Enabling Cross-Stack UplinkFast 17-18
Enabling BackboneFast 17-19
Enabling EtherChannel Guard 17-20
Enabling Root Guard 17-20
Enabling Loop Guard 17-21
Displaying the Spanning-Tree Status 17-21

CHAPTER 18 **Configuring Flex Links and the MAC Address-Table Move Update Feature** 18-1
Understanding Flex Links and the MAC Address-Table Move Update 18-1
Flex Links 18-1
MAC Address-Table Move Update 18-3
Configuring Flex Links and MAC Address-Table Move Update 18-4
Configuration Guidelines 18-4
Default Configuration 18-5
Configuring Flex Links and MAC Address-Table Move Update 18-5
Configuring Flex Links 18-5
Configuring the MAC Address-Table Move Update Feature 18-7
Monitoring Flex Links and the MAC Address-Table Move Update 18-9

CHAPTER 18 **Configuring DHCP Features and IP Source Guard** 18-1
Understanding DHCP Features 18-1
DHCP Server 18-2
DHCP Relay Agent 18-2
DHCP Snooping 18-2
Option-82 Data Insertion 18-3
Cisco IOS DHCP Server Database 18-7
DHCP Snooping Binding Database 18-7
Configuring DHCP Features 18-8
Default DHCP Configuration 18-9
DHCP Snooping Configuration Guidelines 18-10
Upgrading from a Previous Software Release 18-10
Configuring the DHCP Server 18-11

Enabling Only the DHCP Relay Agent 18-11
Enabling the DHCP Relay Agent and Option 82 18-11
Validating the Relay Agent Information Option 82 18-12
Configuring the Reforwarding Policy 18-12
Specifying the Packet Forwarding Address 18-13
Enabling DHCP Snooping and Option 82 18-15
Enabling DHCP Snooping on Private VLANs 18-16
Enabling the Cisco IOS DHCP Server Database 18-17
Enabling the DHCP Snooping Binding Database Agent 18-17
Displaying DHCP Information 18-18
Understanding IP Source Guard 18-18
Source IP Address Filtering 18-19
Source IP and MAC Address Filtering 18-19
Configuring IP Source Guard 18-19
Default IP Source Guard Configuration 18-19
IP Source Guard Configuration Guidelines 18-20
Enabling IP Source Guard 18-20
Displaying IP Source Guard Information 18-22

CHAPTER 19 **Configuring Dynamic ARP Inspection** 19-1
Understanding Dynamic ARP Inspection 19-1
Interface Trust States and Network Security 19-3
Rate Limiting of ARP Packets 19-4
Relative Priority of ARP ACLs and DHCP Snooping Entries 19-4
Logging of Dropped Packets 19-4
Configuring Dynamic ARP Inspection 19-5
Default Dynamic ARP Inspection Configuration 19-5
Dynamic ARP Inspection Configuration Guidelines 19-6
Configuring Dynamic ARP Inspection in DHCP Environments 19-7
Configuring ARP ACLs for Non-DHCP Environments 19-8
Limiting the Rate of Incoming ARP Packets 19-10
Performing Validation Checks 19-11
Configuring the Log Buffer 19-12
Displaying Dynamic ARP Inspection Information 19-14

CHAPTER 20 **Configuring IGMP Snooping and MVR** 20-1
Understanding IGMP Snooping 20-1
IGMP Versions 20-2
Joining a Multicast Group 20-3

Leaving a Multicast Group 20-4
Immediate-Leave Processing 20-5
IGMP Configurable-Leave Timer 20-5
IGMP Report Suppression 20-5
Source-Only Networks 20-6
Configuring IGMP Snooping 20-6
Default IGMP Snooping Configuration 20-7
Enabling or Disabling IGMP Snooping 20-7
Setting the Snooping Method 20-8
Configuring a Multicast Router Port 20-9
Configuring a Host Statically to Join a Group 20-9
Enabling IGMP Immediate-Leave Processing 20-10
Configuring the IGMP Leave Timer 20-11
Configuring TCN-Related Commands 20-11
Controlling the Multicast Flooding Time After a TCN Event 20-12
Recovering from Flood Mode 20-12
Disabling Multicast Flooding During a TCN Event 20-13
Disabling IGMP Report Suppression 20-13
Configuring the Aging Time 20-14
Displaying IGMP Snooping Information 20-14
Understanding Multicast VLAN Registration 20-15
Using MVR in a Multicast Television Application 20-16
Configuring MVR 20-18
Default MVR Configuration 20-18
MVR Configuration Guidelines and Limitations 20-18
Configuring MVR Global Parameters 20-19
Configuring MVR Interfaces 20-20
Displaying MVR Information 20-21
Configuring IGMP Filtering and Throttling 20-22
Default IGMP Filtering and Throttling Configuration 20-22
Configuring IGMP Profiles 20-23
Applying IGMP Profiles 20-24
Setting the Maximum Number of IGMP Groups 20-25
Configuring the IGMP Throttling Action 20-25
Displaying IGMP Filtering and Throttling Configuration 20-27

CHAPTER 21 **Configuring Port-Based Traffic Control** 21-1

Configuring Storm Control 21-1

Understanding Storm Control 21-1

Default Storm Control Configuration 21-3

Configuring Storm Control and Threshold Levels 21-3

Configuring Protected Ports 21-5

Configuring Port Blocking 21-6

Blocking Flooded Traffic on an Interface 21-6

Resuming Normal Forwarding on a Port 21-7

Configuring Port Security 21-7

Understanding Port Security 21-8

Secure MAC Addresses 21-8

Security Violations 21-8

Default Port Security Configuration 21-9

Port Security Configuration Guidelines 21-10

Enabling and Configuring Port Security 21-11

Enabling and Configuring Port Security Aging 21-15

Displaying Port-Based Traffic Control Settings 21-17

CHAPTER 22 **Configuring CDP** 22-1

Understanding CDP 22-1

Configuring CDP 22-2

Default CDP Configuration 22-2

Configuring the CDP Characteristics 22-2

Disabling and Enabling CDP 22-3

Disabling and Enabling CDP on an Interface 22-4

Monitoring and Maintaining CDP 22-4

CHAPTER 23 **Configuring UDLD** 23-1

Understanding UDLD 23-1

Modes of Operation 23-1

Methods to Detect Unidirectional Links 23-2

Configuring UDLD 23-3

Default UDLD Configuration 23-4

Configuration Guidelines 23-4

Enabling UDLD Globally 23-4

Enabling UDLD on an Interface 23-5
Resetting an Interface Shut Down by UDLD 23-6
Displaying UDLD Status 23-7

CHAPTER 24 **Configuring SPAN and RSPAN** 24-1
Understanding SPAN and RSPAN 24-1
SPAN and RSPAN Concepts and Terminology 24-2
SPAN Session 24-3
Traffic Types 24-3
Source Port 24-4
Destination Port 24-5
Reflector Port 24-5
VLAN-Based SPAN 24-6
SPAN Traffic 24-6
SPAN and RSPAN Interaction with Other Features 24-7
SPAN and RSPAN Session Limits 24-8
Default SPAN and RSPAN Configuration 24-8
Configuring SPAN 24-8
SPAN Configuration Guidelines 24-9
Creating a SPAN Session and Specifying Ports to Monitor 24-10
Creating a SPAN Session and Enabling Ingress Traffic 24-11
Removing Ports from a SPAN Session 24-13
Specifying VLANs to Monitor 24-14
Specifying VLANs to Filter 24-15
Configuring RSPAN 24-16
RSPAN Configuration Guidelines 24-16
Configuring a VLAN as an RSPAN VLAN 24-17
Creating an RSPAN Source Session 24-18
Creating an RSPAN Destination Session 24-19
Creating an RSPAN Destination Session and Enabling Ingress Traffic 24-20
Removing Ports from an RSPAN Session 24-21
Specifying VLANs to Monitor 24-21
Specifying VLANs to Filter 24-23
Displaying SPAN and RSPAN Status 24-24

CHAPTER 25 **Configuring RMON** 25-1
Understanding RMON 25-1
Configuring RMON 25-2
Default RMON Configuration 25-3
Configuring RMON Alarms and Events 25-3
Configuring RMON Collection on an Interface 25-4
Displaying RMON Status 25-6

CHAPTER 26 **Configuring System Message Logging** 26-1
Understanding System Message Logging 26-1
Configuring System Message Logging 26-2
System Log Message Format 26-2
Default System Message Logging Configuration 26-3
Disabling and Enabling Message Logging 26-4
Setting the Message Display Destination Device 26-5
Synchronizing Log Messages 26-6
Enabling and Disabling Timestamps on Log Messages 26-7
Enabling and Disabling Sequence Numbers in Log Messages 26-8
Defining the Message Severity Level 26-8
Limiting Syslog Messages Sent to the History Table and to SNMP 26-10
Configuring UNIX Syslog Servers 26-10
Logging Messages to a UNIX Syslog Daemon 26-11
Configuring the UNIX System Logging Facility 26-11
Displaying the Logging Configuration 26-12

CHAPTER 27 **Configuring SNMP** 27-1
Understanding SNMP 27-1
SNMP Versions 27-2
SNMP Manager Functions 27-3
SNMP Agent Functions 27-4
SNMP Community Strings 27-4
Using SNMP to Access MIB Variables 27-4
SNMP Notifications 27-5
SNMP ifIndex MIB Object Values 27-5
Configuring SNMP 27-6
Default SNMP Configuration 27-6
SNMP Configuration Guidelines 27-6
Disabling the SNMP Agent 27-7

Configuring Community Strings 27-8
Configuring SNMP Groups and Users 27-9
Configuring SNMP Notifications 27-11
Configuring SNMP Trap Notification Priority 27-14
Setting the Agent Contact and Location Information 27-15
Limiting TFTP Servers Used Through SNMP 27-16
SNMP Examples 27-16
Displaying SNMP Status 27-17

CHAPTER 28 **Configuring QoS** 28-1
Understanding QoS 28-2
Basic QoS Model 28-4
Classification 28-5
Classification Based on QoS ACLs 28-7
Classification Based on Class Maps and Policy Maps 28-7
Policing and Marking 28-8
Mapping Tables 28-10
Queueing and Scheduling 28-11
Queueing and Scheduling on Gigabit-Capable Ports 28-11
Queueing and Scheduling on 10/100 Ethernet Ports 28-14
Packet Modification 28-16
Configuring Auto-QoS 28-17
Generated Auto-QoS Configuration 28-17
Effects of Auto-QoS on the Configuration 28-20
Configuration Guidelines 28-21
Upgrading from a Previous Software Release 28-21
Enabling Auto-QoS for VoIP 28-22
Displaying Auto-QoS Information 28-23
Auto-QoS Configuration Example 28-23
Configuring Standard QoS 28-25
Default Standard QoS Configuration 28-26
Standard QoS Configuration Guidelines 28-27
Enabling QoS Globally 28-29
Configuring Classification By Using Port Trust States 28-30
Configuring the Trust State on Ports within the QoS Domain 28-30
Configuring the CoS Value for an Interface 28-32
Configuring a Trusted Boundary to Ensure Port Security 28-32
Enabling Pass-Through Mode 28-34
Configuring the DSCP Trust State on a Port Bordering Another QoS Domain 28-34

Configuring a QoS Policy 28-36
Classifying Traffic by Using ACLs 28-36
Classifying Traffic on a Physical-Port Basis by Using Class Maps 28-40
Classifying Traffic on a Per-Port Per-VLAN Basis by Using Class Maps 28-41
Classifying, Policing, and Marking Traffic by Using Policy Maps 28-44
Classifying, Policing, and Marking Traffic by Using Aggregate Policers 28-50
Configuring DSCP Maps 28-52
Configuring the CoS-to-DSCP Map 28-52
Configuring the IP-Precedence-to-DSCP Map 28-53
Configuring the Policed-DSCP Map 28-54
Configuring the DSCP-to-CoS Map 28-55
Configuring the DSCP-to-DSCP-Mutation Map 28-57
Configuring Egress Queues on Gigabit-Capable Ethernet Ports 28-58
Mapping CoS Values to Select Egress Queues 28-59
Configuring the Egress Queue Size Ratios 28-59
Configuring Tail-Drop Threshold Percentages 28-60
Configuring WRED Drop Thresholds Percentages 28-62
Configuring the Egress Expedite Queue 28-63
Allocating Bandwidth among Egress Queues 28-64
Configuring Egress Queues on 10/100 Ethernet Ports 28-65
Mapping CoS Values to Select Egress Queues 28-65
Configuring the Minimum-Reserve Levels 28-66
Configuring the Egress Expedite Queue 28-67
Allocating Bandwidth among Egress Queues 28-68
Displaying Standard QoS Information 28-69
Standard QoS Configuration Examples 28-70
QoS Configuration for the Existing Wiring Closet 28-71
QoS Configuration for the Intelligent Wiring Closet 28-72
QoS Configuration for the Distribution Layer 28-73

CHAPTER 29 **Configuring Network Security with ACLs** 29-1
Understanding ACLs 29-1
Supported ACLs 29-2
Router ACLs 29-3
Port ACLs 29-4
VLAN Maps 29-4
Handling Fragmented and Unfragmented Traffic 29-5

Configuring IP ACLs 29-6
Hardware and Software Handling of Router ACLs 29-6
Configuration Guidelines for Input Router ACLs 29-7
Unsupported Features 29-8
Creating Standard and Extended IP ACLs 29-8
Access List Numbers 29-9
Creating a Numbered Standard ACL 29-10
Creating a Numbered Extended ACL 29-11
Resequencing ACEs in an ACL 29-16
Creating Named Standard and Extended IP ACLs 29-16
Using Time Ranges with ACLs 29-18
Including Comments in ACLs 29-20
Applying an IP ACL to an Interface or Terminal Line 29-21
IP ACL Configuration Examples 29-23
Numbered ACLs 29-24
Extended ACLs 29-24
Named ACLs 29-25
Time Range Applied to an IP ACL 29-25
Commented IP ACL Entries 29-26
ACL Logging 29-26
Configuring Named MAC Extended ACLs 29-27
Applying a MAC ACL to a Layer 2 Interface 29-29
Configuring VLAN Maps 29-30
VLAN Map Configuration Guidelines 29-31
Creating a VLAN Map 29-31
Examples of ACLs and VLAN Maps 29-32
Applying a VLAN Map to a VLAN 29-34
Using VLAN Maps in Your Network 29-34
Wiring Closet Configuration 29-34
Denying Access to a Server on Another VLAN 29-35
Using VLAN Maps with Router ACLs 29-36
Guidelines for Using Router ACLs and VLAN Maps 29-37
Examples of Router ACLs and VLAN Maps Applied to VLANs 29-38
ACLs and Switched Packets 29-38
ACLs and Bridged Packets 29-38
ACLs and Routed Packets 29-39
ACLs and Multicast Packets 29-40

Displaying ACL Information 29-40
Displaying ACL Configuration 29-41
Displaying ACL Resource Usage and Configuration Problems 29-42
Configuration Conflicts 29-43
ACL Configuration Fitting in Hardware 29-44
TCAM Usage 29-47

CHAPTER 30 **Configuring EtherChannels** 30-1
Understanding EtherChannels 30-1
Understanding Port-Channel Interfaces 30-3
Understanding the Port Aggregation Protocol and Link Aggregation Protocol 30-3
PAgP and LACP Modes 30-4
Physical Learners and Aggregate-Port Learners 30-5
PAgP and LACP Interaction with Other Features 30-5
EtherChannel On Mode 30-6
Understanding Load Balancing and Forwarding Methods 30-6
Configuring EtherChannels 30-7
Default EtherChannel Configuration 30-8
EtherChannel Configuration Guidelines 30-8
Configuring Layer 2 EtherChannels 30-9
Configuring Layer 3 EtherChannels 30-12
Creating Port-Channel Logical Interfaces 30-12
Configuring the Physical Interfaces 30-13
Configuring EtherChannel Load Balancing 30-15
Configuring the PAgP Learn Method and Priority 30-15
Configuring the LACP Port Priority 30-17
Configuring Hot Standby Ports 30-17
Configuring the LACP System Priority 30-18
Displaying EtherChannel, PAgP, and LACP Status 30-18

CHAPTER 31 **Configuring IP Unicast Routing** 31-1
Understanding IP Routing 31-2
Steps for Configuring Routing 31-3
Configuring IP Addressing on Layer 3 Interfaces 31-4
Default Addressing Configuration 31-4
Assigning IP Addresses to Network Interfaces 31-5
Use of Subnet Zero 31-6
Classless Routing 31-7

Configuring Address Resolution Methods 31-8
Define a Static ARP Cache 31-9
Set ARP Encapsulation 31-10
Enable Proxy ARP 31-10
Routing Assistance When IP Routing is Disabled 31-11
Proxy ARP 31-11
Default Gateway 31-11
ICMP Router Discovery Protocol (IRDP) 31-12
Configuring Broadcast Packet Handling 31-13
Enabling Directed Broadcast-to-Physical Broadcast Translation 31-13
Forwarding UDP Broadcast Packets and Protocols 31-14
Establishing an IP Broadcast Address 31-15
Flooding IP Broadcasts 31-16
Monitoring and Maintaining IP Addressing 31-17
Enabling IP Unicast Routing 31-18
Configuring RIP 31-19
Default RIP Configuration 31-19
Configuring Basic RIP Parameters 31-20
Configuring RIP Authentication 31-22
Configuring Summary Addresses and Split Horizon 31-22
Configuring OSPF 31-24
Default OSPF Configuration 31-25
Nonstop Forwarding Awareness 31-26
Configuring Basic OSPF Parameters 31-26
Configuring OSPF Interfaces 31-27
Configuring OSPF Area Parameters 31-28
Configuring Other OSPF Parameters 31-30
Changing LSA Group Pacing 31-32
Configuring a Loopback Interface 31-32
Monitoring OSPF 31-33
Configuring EIGRP 31-34
Default EIGRP Configuration 31-35
Nonstop Forwarding Awareness 31-37
Configuring Basic EIGRP Parameters 31-37
Configuring EIGRP Interfaces 31-38
Configuring EIGRP Route Authentication 31-39
Monitoring and Maintaining EIGRP 31-40

Configuring BGP 31-40
Default BGP Configuration 31-42
Nonstop Forwarding Awareness 31-44
Enabling BGP Routing 31-45
Managing Routing Policy Changes 31-47
Configuring BGP Decision Attributes 31-49
Configuring BGP Filtering with Route Maps 31-51
Configuring BGP Filtering by Neighbor 31-51
Configuring Prefix Lists for BGP Filtering 31-53
Configuring BGP Community Filtering 31-54
Configuring BGP Neighbors and Peer Groups 31-55
Configuring Aggregate Addresses 31-57
Configuring a Routing Domain Confederation 31-58
Configuring BGP Route Reflectors 31-58
Configuring Route Dampening 31-59
Monitoring and Maintaining BGP 31-60
Configuring Multi-VRF CE 31-61
Understanding Multi-VRF CE 31-62
Default Multi-VRF CE Configuration 31-64
Multi-VRF CE Configuration Guidelines 31-64
Configuring VRFs 31-65
Configuring a VPN Routing Session 31-66
Configuring BGP PE to CE Routing Sessions 31-67
Multi-VRF CE Configuration Example 31-67
Displaying Multi-VRF CE Status 31-71
Configuring Protocol-Independent Features 31-72
Configuring Cisco Express Forwarding 31-72
Configuring the Number of Equal-Cost Routing Paths 31-73
Configuring Static Unicast Routes 31-74
Specifying Default Routes and Networks 31-75
Using Route Maps to Redistribute Routing Information 31-76
Configuring Policy-Based Routing 31-79
PBR Configuration Guidelines 31-79
Enabling PBR 31-80
Filtering Routing Information 31-81
Setting Passive Interfaces 31-81
Controlling Advertising and Processing in Routing Updates 31-82
Filtering Sources of Routing Information 31-82
Managing Authentication Keys 31-83

Monitoring and Maintaining the IP Network 31-84

CHAPTER 32 **Configuring HSRP** 32-1

Understanding HSRP 32-1

Configuring HSRP 32-4

Default HSRP Configuration 32-4

HSRP Configuration Guidelines and Limitations 32-4

Enabling HSRP 32-5

Configuring HSRP Priority 32-6

Configuring HSRP Authentication and Timers 32-8

Configuring HSRP Groups and Clustering 32-10

Displaying HSRP Configurations 32-10

CHAPTER 33 **Configuring Cisco IOS IP SLAs Operations** 33-1

Understanding Cisco IOS IP SLAs 33-1

Using Cisco IOS IP SLAs to Measure Network Performance 33-2

IP SLAs Responder and IP SLAs Control Protocol 33-3

Response Time Computation for IP SLAs 33-4

IP SLAs Operation Scheduling 33-5

IP SLAs Operation Threshold Monitoring 33-5

Configuring IP SLAs Operations 33-6

Default Configuration 33-6

Configuration Guidelines 33-6

Configuring the IP SLAs Responder 33-7

Analyzing IP Service Levels by Using the UDP Jitter Operation 33-8

Analyzing IP Service Levels by Using the ICMP Echo Operation 33-10

Monitoring IP SLAs Operations 33-13

CHAPTER 34 **Configuring Web Cache Services By Using WCCP** 34-1

Understanding WCCP 34-1

WCCP Message Exchange 34-2

WCCP Negotiation 34-3

MD5 Security 34-3

Packet Redirection 34-3

Unsupported WCCPv2 Features 34-4

Configuring WCCP 34-4
Default WCCP Configuration 34-4
WCCP Configuration Guidelines 34-4
Enabling the Web Cache Service, Setting the Password, and Redirecting Traffic Received From a Client 34-5
Monitoring and Maintaining WCCP 34-8

CHAPTER 35 **Configuring IP Multicast Routing** 35-1
Understanding Cisco's Implementation of IP Multicast Routing 35-2
Understanding IGMP 35-2
IGMP Version 1 35-3
IGMP Version 2 35-3
Understanding PIM 35-4
PIM Versions 35-4
PIM Modes 35-4
Auto-RP 35-5
Bootstrap Router 35-5
Multicast Forwarding and Reverse Path Check 35-6
Understanding DVMRP 35-7
Understanding CGMP 35-8
Configuring IP Multicast Routing 35-8
Default Multicast Routing Configuration 35-9
Multicast Routing Configuration Guidelines 35-9
PIMv1 and PIMv2 Interoperability 35-9
Auto-RP and BSR Configuration Guidelines 35-10
Configuring Basic Multicast Routing 35-10
Configuring a Rendezvous Point 35-12
Manually Assigning an RP to Multicast Groups 35-12
Configuring Auto-RP 35-14
Configuring PIMv2 BSR 35-18
Using Auto-RP and a BSR 35-22
Monitoring the RP Mapping Information 35-23
Troubleshooting PIMv1 and PIMv2 Interoperability Problems 35-23
Configuring Advanced PIM Features 35-23
Understanding PIM Shared Tree and Source Tree 35-23
Delaying the Use of PIM Shortest-Path Tree 35-25
Modifying the PIM Router-Query Message Interval 35-26

Configuring Optional IGMP Features 35-26
Default IGMP Configuration 35-27
Configuring the Multilayer Switch as a Member of a Group 35-27
Controlling Access to IP Multicast Groups 35-28
Changing the IGMP Version 35-29
Modifying the IGMP Host-Query Message Interval 35-29
Changing the IGMP Query Timeout for IGMPv2 35-30
Changing the Maximum Query Response Time for IGMPv2 35-31
Configuring the Multilayer Switch as a Statically Connected Member 35-31
Configuring Optional Multicast Routing Features 35-32
Enabling CGMP Server Support 35-32
Configuring sdr Listener Support 35-33
Enabling sdr Listener Support 35-34
Limiting How Long an sdr Cache Entry Exists 35-34
Configuring the TTL Threshold 35-34
Configuring an IP Multicast Boundary 35-36
Configuring Basic DVMRP Interoperability Features 35-38
Configuring DVMRP Interoperability 35-38
Configuring a DVMRP Tunnel 35-40
Advertising Network 0.0.0.0 to DVMRP Neighbors 35-42
Responding to mrinfo Requests 35-43
Configuring Advanced DVMRP Interoperability Features 35-43
Enabling DVMRP Unicast Routing 35-43
Rejecting a DVMRP Nonpruning Neighbor 35-44
Controlling Route Exchanges 35-46
Limiting the Number of DVMRP Routes Advertised 35-47
Changing the DVMRP Route Threshold 35-47
Configuring a DVMRP Summary Address 35-48
Disabling DVMRP Autosummarization 35-50
Adding a Metric Offset to the DVMRP Route 35-50
Monitoring and Maintaining IP Multicast Routing 35-51
Clearing Caches, Tables, and Databases 35-51
Displaying System and Network Statistics 35-52
Monitoring IP Multicast Routing 35-53

CHAPTER 36 **Configuring MSDP** 36-1
Understanding MSDP 36-1
MSDP Operation 36-2
MSDP Benefits 36-3

Configuring MSDP 36-3
Default MSDP Configuration 36-4
Configuring a Default MSDP Peer 36-4
Caching Source-Active State 36-6
Requesting Source Information from an MSDP Peer 36-8
Controlling Source Information that Your Switch Originates 36-8
Redistributing Sources 36-9
Filtering Source-Active Request Messages 36-10
Controlling Source Information that Your Switch Forwards 36-11
Using a Filter 36-11
Using TTL to Limit the Multicast Data Sent in SA Messages 36-13
Controlling Source Information that Your Switch Receives 36-13
Configuring an MSDP Mesh Group 36-15
Shutting Down an MSDP Peer 36-15
Including a Bordering PIM Dense-Mode Region in MSDP 36-16
Configuring an Originating Address other than the RP Address 36-17
Monitoring and Maintaining MSDP 36-17

CHAPTER 37 **Configuring Fallback Bridging** 37-1
Understanding Fallback Bridging 37-1
Configuring Fallback Bridging 37-2
Default Fallback Bridging Configuration 37-3
Fallback Bridging Configuration Guidelines 37-3
Creating a Bridge Group 37-3
Preventing the Forwarding of Dynamically Learned Stations 37-5
Configuring the Bridge Table Aging Time 37-5
Filtering Frames by a Specific MAC Address 37-6
Adjusting Spanning-Tree Parameters 37-7
Changing the Switch Priority 37-8
Changing the Interface Priority 37-8
Assigning a Path Cost 37-9
Adjusting BPDU Intervals 37-10
Disabling the Spanning Tree on an Interface 37-12
Monitoring and Maintaining Fallback Bridging 37-12

CHAPTER 38 **Troubleshooting** 38-1

Using Recovery Procedures 38-1
Recovering from a Software Failure 38-2
Recovering from a Lost or Forgotten Password 38-2
Password Recovery with Password Recovery Enabled 38-3
Procedure with Password Recovery Disabled 38-5
Recovering from a Command Switch Failure 38-6
Replacing a Failed Command Switch with a Cluster Member 38-7
Replacing a Failed Command Switch with Another Switch 38-8
Recovering from Lost Member Connectivity 38-10
Preventing Autonegotiation Mismatches 38-10
GBIC Module Security and Identification 38-10
Diagnosing Connectivity Problems 38-11
Using Ping 38-11
Understanding Ping 38-11
Executing Ping 38-11
Using IP Traceroute 38-12
Understanding IP Traceroute 38-13
Executing IP Traceroute 38-13
Using Layer 2 Traceroute 38-14
Understanding Layer 2 Traceroute 38-14
Usage Guidelines 38-15
Displaying the Physical Path 38-15
Troubleshooting Power over Ethernet Switch Ports 38-16
Disabled Port Caused by Power Loss 38-16
Disabled Port Caused by False Link-Up 38-16
Using Debug Commands 38-16
Enabling Debugging on a Specific Feature 38-17
Enabling All-System Diagnostics 38-17
Redirecting Debug and Error Message Output 38-18
Using the debug auto qos Command 38-18
Using the show forward Command 38-19
Using the crashinfo File 38-20
Basic crashinfo Files 38-20
Extended crashinfo Files 38-21

APPENDIX A **Supported MIBs** A-1
MIB List A-1
Using FTP to Access the MIB Files A-3

APPENDIX B **Working with the Cisco IOS File System, Configuration Files, and Software Images** B-1
Working with the Flash File System B-1
Displaying Available File Systems B-2
Setting the Default File System B-3
Displaying Information about Files on a File System B-3
Changing Directories and Displaying the Working Directory B-3
Creating and Removing Directories B-4
Copying Files B-4
Deleting Files B-5
Creating, Displaying, and Extracting tar Files B-5
Creating a tar File B-5
Displaying the Contents of a tar File B-6
Extracting a tar File B-7
Displaying the Contents of a File B-7
Working with Configuration Files B-7
Guidelines for Creating and Using Configuration Files B-8
Configuration File Types and Location B-9
Creating a Configuration File By Using a Text Editor B-9
Copying Configuration Files By Using TFTP B-9
Preparing to Download or Upload a Configuration File By Using TFTP B-10
Downloading the Configuration File By Using TFTP B-10
Uploading the Configuration File By Using TFTP B-11
Copying Configuration Files By Using FTP B-12
Preparing to Download or Upload a Configuration File By Using FTP B-12
Downloading a Configuration File By Using FTP B-13
Uploading a Configuration File By Using FTP B-14
Copying Configuration Files By Using RCP B-15
Preparing to Download or Upload a Configuration File By Using RCP B-15
Downloading a Configuration File By Using RCP B-16
Uploading a Configuration File By Using RCP B-17
Clearing Configuration Information B-18
Clearing the Startup Configuration File B-18
Deleting a Stored Configuration File B-18

Working with Software Images B-18
Image Location on the Switch B-19
tar File Format of Images on a Server or Cisco.com B-19
Copying Image Files By Using TFTP B-20
Preparing to Download or Upload an Image File By Using TFTP B-21
Downloading an Image File By Using TFTP B-21
Uploading an Image File By Using TFTP B-23
Copying Image Files By Using FTP B-23
Preparing to Download or Upload an Image File By Using FTP B-24
Downloading an Image File By Using FTP B-25
Uploading an Image File By Using FTP B-26
Copying Image Files By Using RCP B-27
Preparing to Download or Upload an Image File By Using RCP B-28
Downloading an Image File By Using RCP B-29
Uploading an Image File By Using RCP B-31

APPENDIX C **Unsupported CLI Commands in Cisco IOS Release 12.2(25)SEE** C-1
Access Control Lists C-1
Unsupported Privileged EXEC Commands C-1
ARP Commands C-1
Unsupported Global Configuration Commands C-1
Unsupported Interface Configuration Commands C-1
Boot Loader Commands C-2
Unsupported Global Configuration Commands C-2
FallBack Bridging C-2
Unsupported Privileged EXEC Commands C-2
Unsupported Global Configuration Commands C-2
Unsupported Interface Configuration Commands C-3
HSRP C-3
Unsupported Global Configuration Commands C-3
Unsupported Interface Configuration Commands C-4
Interface Configuration Commands C-4
IP Multicast Routing C-4
Unsupported Privileged EXEC Commands C-4
Unsupported Global Configuration Commands C-5
Unsupported Interface Configuration Commands C-5
IP Unicast Routing C-5
Unsupported Privileged EXEC or User EXEC Commands C-5
Unsupported Global Configuration Commands C-6

Unsupported Interface Configuration Commands C-6
Unsupported BGP Router Configuration Commands C-6
Unsupported VPN Configuration Commands C-7
Unsupported Route Map Commands C-7
Miscellaneous C-7
Privileged EXEC Commands C-7
MSDP C-8
Unsupported Privileged EXEC Commands C-8
Unsupported Global Configuration Commands C-8
NetFlow Commands C-8
Unsupported Global Configuration Commands C-8
Network Address Translation (NAT) commands C-8
Unsupported User EXEC Commands C-8
Unsupported Global Configuration Commands C-8
Unsupported Interface Configuration Commands C-9
QoS C-9
Unsupported Global Configuration Commands C-9
Unsupported Interface Configuration Command C-9
Unsupported Policy-Map Configuration Command C-9
Unsupported Class-Map Configuration Commands C-9
RADIUS C-10
Unsupported Global Configuration Commands C-10
SNMP C-10
Unsupported Global Configuration Commands C-10
Spanning Tree C-10
Unsupported Global Configuration Command C-10
VLAN C-10
Unsupported User EXEC Commands C-10

INDEX

Preface

Audience

This guide is for the networking professional managing the Catalyst 3550 switch, hereafter referred to as *the switch* or *the multilayer switch*. Before using this guide, you should have experience working with the Cisco IOS and be familiar with the concepts and terminology of Ethernet and local area networking.

Purpose

This guide provides the information that you need to configure Layer 2 and Layer 3 software features on your switch. The Catalyst 3550 switch is supported by either the IP base image (formerly known as the standard multilayer image [SMI]), which provides Layer 2+ features and basic Layer 3 routing, or the IP services image (formerly known as the enhanced multilayer image [EMI]), which provides Layer 2+ features, full Layer 3 routing, and advanced services. All Catalyst 3550 Gigabit Ethernet switches are shipped with the IP services image pre-installed. Catalyst 3550 Fast Ethernet switches are shipped with either the IP base image or the IP services image pre-installed. After initial deployment, you can order the software upgrade kit to upgrade Catalyst 3550 Fast Ethernet switches from the IP base image to the IP services image.

Use this guide with other documents for information about these topics:

- Requirements—This guide assumes that you have met the hardware and software requirements and cluster compatibility requirements described in the release notes.
- Start-up information—This guide assumes that you have assigned switch IP information and passwords by using the browser setup program described in the switch hardware installation guide.
- Embedded device manager and Network Assistant graphical user interfaces (GUIs)—This guide does not provide detailed information on the GUIs. However, the concepts in this guide are applicable to the GUI user. For information about the device manager, see the switch online help. For information about Network Assistant, see the *Getting Started with Cisco Network Assistant*, available on Cisco.com.
- Cluster configuration—For information about planning for, creating, and maintaining switch clusters, see the *Getting Started with Cisco Network Assistant*, available on Cisco.com. For information about the clustering-related command-line interface (CLI) commands, see the *command reference for this release*.
- CLI command information—This guide provides an overview for using the CLI. For complete syntax and usage information about the commands that have been specifically created or changed for the switches, see the *command reference for this release*.

This guide provides procedures for using the commands that have been created or changed for use with the switch. It does not provide detailed information about these commands. For detailed information about these commands, see the command reference for this release.

This guide does not repeat the concepts and CLI procedures provided in the standard Cisco IOS Release 12.2 documentation. For information about the standard Cisco IOS Release 12.2 commands, see the Cisco IOS documentation set available from the Cisco.com home page at **Service and Support > Technical Documents**. On the Cisco Product Documentation home page, select Release 12.2 from the Cisco IOS Software drop-down list.

This guide does not describe system messages you might encounter or how to install your switch. For this information, see the system message guide for this release and to the hardware installation guide.

For documentation updates, see the release notes for this release.

Conventions

This publication uses these conventions to convey instructions and information:

Command descriptions use these conventions:

- Commands and keywords are in **boldface** text.
- Arguments for which you supply values are in *italic*.
- Square brackets ([ ]) mean optional elements.
- Braces ({ }) group required choices, and vertical bars (|) separate the alternative elements.
- Braces and vertical bars within square brackets ([{ | }]) mean a required choice within an optional element.

Interactive examples use these conventions:

- Terminal sessions and system displays are in `screen` font.
- Information you enter is in **`boldface screen`** font.
- Nonprinting characters, such as passwords or tabs, are in angle brackets (< >).

Notes, cautions, and timesavers use these conventions and symbols:

Note Means *reader take note*. Notes contain helpful suggestions or references to materials not contained in this manual.

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

Timesaver Means the following *will help you solve a problem*. The tips information might not be troubleshooting or even an action, but could be useful information.

Related Publications

These documents provide complete information about the switch and are available from this Cisco.com site:

http://www.cisco.com/en/US/products/hw/switches/ps646/tsd_products_support_series_home.html

- *Release Notes for the Catalyst 3550 Multilayer Switch*

Note Switch requirements and procedures for initial configurations and software upgrades tend to change and therefore appear only in the release notes. Before installing, configuring, or upgrading the switch, see the release notes on Cisco.com for the latest information.

For information about the switch, see these documents:

- *Catalyst 3750, 3560, 3550, 2970, and 2960 Switch System Message Guide*
- *Catalyst 3550, 2955, 2950, and 2940 Switch System Message Guide*
- *Catalyst 3550 Multilayer Switch Software Configuration Guide*
- *Catalyst 3550 Multilayer Switch Command Reference*
- Device manager online help (available on the switch)
- *Catalyst 3550 Multilayer Switch Hardware Installation Guide*
- *Catalyst 3550 Switch Getting Started Guide*
- *Regulatory Compliance and Safety Information for the* Catalyst 3550 Switch

For information about related products, see these documents:

- *Getting Started with Cisco Network Assistant* (not orderable but available on Cisco.com)
- *Release Notes for Cisco Network Assistant* (not orderable but available on Cisco.com)
- *Catalyst GigaStack Gigabit Interface Converter Hardware Installation Guide*
- *CWDM Passive Optical System Installation Note*
- *1000BASE-T Gigabit Interface Converter Installation Notes*
- *Cisco Small Form-Factor Pluggable Modules Installation Notes*
- *Cisco CWDM GBIC and CWDM SFP Installation Note*
- For information about the NAC features, see the *Network Admission Control Software Configuration Guide*

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, 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

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS version 2.0.

CHAPTER 1

Overview

This chapter provides these topics about the Catalyst 3550 multilayer switch software:

- Features, page 1-1
- Management Options, page 1-9
- Network Configuration Examples, page 1-10
- Where to Go Next, page 1-19

In this document, IP refers to IP version 4 (IPv4). Layer 3 IP version 6 (IPv6) packets are treated as non-IP packets.

Features

The software supports the hardware listed in the release notes. This section describes the features supported in this release:

Note All Catalyst 3550 Gigabit Ethernet switches ship with the IP services image, formerly known as the enhanced multilayer image (EMI), which provides Layer 2+ features, full Layer 3 routing, and advanced services. Catalyst 3550 Fast Ethernet switches can be shipped with either the IP base image, formerly known as the standard multilayer software image (SMI), or the IP services image installed. The IP base image provides Layer 2+ features and basic Layer 3 routing. You can order the IP services Image Upgrade kit to upgrade Catalyst 3550 Fast Ethernet switches from the IP base image to the IP services image.

Ease of Deployment and Ease of Use

The switch ships with these features to make the deployment and the use easier:

- Express Setup for quickly configuring a switch for the first time with basic IP information, contact information, switch and Telnet passwords, and Simple Network Management Protocol (SNMP) information through a browser-based program
- User-defined Smartports macros for creating custom switch configurations for simplified deployment across the network
- An embedded device manager for configuring and monitoring a single switch through a web browser. For information about launching the device manager, see the getting started guide. For more information about the device manager, see the switch online help.

- Network Assistant GUI for
  - Unified configuration, monitoring, authentication, and software upgrade of multiple switches (see the release notes for a list of eligible cluster members).
  - Automatic discovery of candidate switches and creation of clusters of up to 16 switches that can be managed through a single IP address.
  - Extended discovery of cluster candidates that are not directly connected to the command switch.
  - Downloading an image to a switch by using HTTP or TFTP.

Performance

- Autosensing of port speed and autonegotiation of duplex mode on all switch ports for optimizing bandwidth
- IEEE 802.3x flow control on all Ethernet ports
- EtherChannel for enhanced fault tolerance and for providing up to 8 Gbps (Gigabit EtherChannel) or 800 Mbps (Fast EtherChannel) full duplex of bandwidth between switches, routers, and servers
- Port Aggregation Protocol (PAgP) and Link Aggregation Control Protocol (LACP) for automatic creation of EtherChannel links
- Per-port storm control for preventing broadcast, multicast, and unicast storms
- Port blocking on forwarding unknown unicast and multicast traffic
- Cisco Group Management Protocol (CGMP) server support and Internet Group Management Protocol (IGMP) snooping for IGMP versions 1, 2, and 3:
  - (For CGMP devices) CGMP for limiting multicast traffic to specified end stations and reducing overall network traffic
  - (For IGMP devices) IGMP snooping for limiting flooding of multicast traffic
- IGMP report suppression for sending only one IGMP report per multicast router query to the multicast devices (supported only for IGMPv1 or IGMPv2 queries)
- Multicast VLAN registration (MVR) to continuously send multicast streams in a multicast VLAN while isolating the streams from subscriber VLANs for bandwidth and security reasons
- IGMP filtering for controlling the set of multicast groups to which hosts on a switch port can belong
- IGMP throttling for configuring the action when the maximum number of entries is in the IGMP forwarding table
- IGMP leave timer for configuring the leave latency for the network.
- System Database Management (SDM) templates for allocating system resources to maximize support for user-selected features
- Web Cache Communication Protocol (WCCP) for redirecting traffic to local cache engines, for enabling content requests to be fulfilled locally, and for localizing web-traffic patterns in the network (requires the enhanced multilayer software image)

Manageability

- Cisco Intelligence Engine 2100 (IE2100) Series Cisco Networking Services (CNS) embedded agents for automating switch management, configuration storage and delivery.

- DHCP for automating configuration of switch information (such as IP address, default gateway, host name, and Domain Name System [DNS] and TFTP server names)
- DHCP server for automatic assignment of IP addresses and other DHCP options to IP hosts
- DHCP relay agent information (option 82) for subscriber identification and IP address management
- Directed unicast requests to a DNS server for identifying a switch through its IP address and its corresponding host name and to a TFTP server for administering software upgrades from a TFTP server
- Address Resolution Protocol (ARP) for identifying a switch through its IP address and its corresponding Media Access Control (MAC) address
- Unicast MAC address filtering to drop packets with specific source or destination MAC addresses
- Cisco Discovery Protocol (CDP) versions 1 and 2 for network topology discovery and mapping between the switch and other Cisco devices on the network
- Network Time Protocol (NTP) for providing a consistent timestamp to all switches from an external source
- Cisco IOS File System (IFS) for providing a single interface to all file systems that the switch uses
- Configuration logging to log and to view changes to the switch configuration
- Unique device identifier to provide product identification information through a **show inventory** user EXEC command display
- In-band management access through the embedded device manager over a Netscape Navigator or Internet Explorer session or through the Network Assistant application
- In-band management access through up to 16 simultaneous Telnet connections for multiple command-line interface (CLI)-based sessions over the network
- In-band management access for up to five simultaneous, encrypted Secure Shell (SSH) connections for multiple CLI-based sessions over the network
- In-band management access through SNMP versions 1, 2c, and 3 get and set requests
- Out-of-band management access through the switch console port to a directly attached terminal or to a remote terminal through a serial connection or a modem
- Secure Copy Protocol (SCP) feature to provide a secure and authenticated method for copying switch configuration or switch image files (requires the cryptographic version of the software)

Note For additional descriptions of the management interfaces, see the "Management Options" section on page 1-9.

Redundancy

- Hot Standby Router Protocol (HSRP) for command switch and Layer 3 router redundancy
- UniDirectional Link Detection (UDLD) and aggressive UDLD on all Ethernet ports for detecting and disabling unidirectional links on fiber-optic interfaces caused by incorrect fiber-optic wiring or port faults
- IEEE 802.1D Spanning Tree Protocol (STP) for redundant backbone connections and loop-free networks. STP has these features:
  - Per-VLAN spanning-tree plus (PVST+) for load balancing across VLANs

  - Rapid PVST+ for load balancing across VLANs
  - UplinkFast, cross-stack UplinkFast, and BackboneFast for fast convergence after a spanning-tree topology change and for achieving load balancing between redundant uplinks, including Gigabit uplinks and cross-stack Gigabit uplinks
- IEEE 802.1s Multiple Spanning Tree Protocol (MSTP) for grouping VLANs into a spanning-tree instance, and providing for multiple forwarding paths for data traffic and load balancing
- IEEE 802.1w Rapid Spanning Tree Protocol (RSTP) for rapid convergence of the spanning tree by immediately transitioning root and designated ports to the forwarding state
- Optional spanning-tree features available in PVST+, rapid-PVST+, and MSTP mode:
  - Port Fast for eliminating the forwarding delay by enabling a port to immediately transition from the blocking state to the forwarding state
  - BPDU guard for shutting down Port Fast-enabled ports that receive BPDUs
  - BPDU filtering for preventing a Port Fast-enabled port from sending or receiving BPDUs
  - Root guard for preventing switches outside the network core from becoming the spanning-tree root
  - Loop guard for preventing alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link

Note The switch supports up to 128 spanning-tree instances.

- Flex Link Layer 2 interfaces to back up one another as an alternative to STP for basic link redundancy

VLAN Support

- Support for up to 1005 VLANs for assigning users to VLANs associated with appropriate network resources, traffic patterns, and bandwidth
- Support for VLAN IDs in the full 1 to 4094 range as allowed by the IEEE 802.1Q standard
- VLAN Query Protocol (VQP) for dynamic VLAN membership
- Inter-Switch Link (ISL) and IEEE 802.1Q trunking encapsulation on all ports for network moves, adds, and changes; management and control of broadcast and multicast traffic; and network security by establishing VLAN groups for high-security users and network resources
- Dynamic Trunking Protocol (DTP) for negotiating trunking on a link between two devices and for negotiating the type of trunking encapsulation (IEEE 802.1Q or ISL) to be used
- VLAN Trunking Protocol (VTP) and VTP pruning for reducing network traffic by restricting flooded traffic to links destined for stations receiving the traffic
- Voice VLAN for creating subnets for voice traffic from Cisco IP Phones
- VLAN 1 minimization to reduce the risk of spanning-tree loops or storms by allowing VLAN 1 to be disabled on any individual VLAN trunk link. With this feature enabled, no user traffic is sent or received. The switch CPU continues to send and receive control protocol frames.

Security

- Web authentication to allow a supplicant (client) that does not support IEEE 802.1x functionality to be authenticated using a web browser.
- Password-protected access (read-only and read-write access) to management interfaces (device manager, Network Assistant, and CLI) for protection against unauthorized configuration changes
- Multilevel security for a choice of security level, notification, and resulting actions
- Static MAC addressing for ensuring security
- Protected port option for restricting the forwarding of traffic to designated ports on the same switch
- Port security option for limiting and identifying MAC addresses of the stations allowed to access the port
- Port security on trunk ports for limiting and identifying MAC addresses of the stations allowed to access the VLAN
- Port security aging to set the aging time for secure addresses on a port
- IP source guard to restrict traffic on nonrouted interfaces by filtering traffic based on the DHCP snooping database and IP source bindings.
- DHCP snooping to filter untrusted DHCP messages between untrusted hosts and DHCP servers
- Dynamic ARP inspection to prevent malicious attacks on the switch by not relaying invalid ARP requests and responses to other ports in the same VLAN
- Bridge protocol data unit (BPDU) guard for shutting down a Port Fast-configured port when an invalid configuration occurs
- Standard and extended IP access control lists (ACLs) for defining security policies in both directions on routed interfaces (router ACLs) and inbound on Layer 2 interfaces (port ACLs)
- Extended MAC access control lists for defining security policies in the inbound direction on Layer 2 interfaces
- VLAN ACLs (VLAN maps) for providing intra-VLAN security by filtering traffic based on information in the MAC, IP, and TCP/User Datagram Protocol (UDP) headers
- Source and destination MAC-based ACLs for filtering non-IP traffic
- IEEE 802.1x port-based authentication to prevent unauthorized devices (clients) from gaining access to the network
- IEEE 802.1x with per-user access control lists for providing different levels of network access and service to an IEEE 802.1x-authenticated user
- IEEE 802.1x with VLAN assignment for restricting IEEE 802.1x-authenticated users to a specified VLAN
- IEEE 802.1x with port security for controlling access to IEEE 802.1x multiple-host ports
- IEEE 802.1x with voice VLAN to permit an IP phone access to the voice VLAN irrespective of the authorized or unauthorized state of the port
- IEEE 802.1x with guest VLAN to provide limited services to non-IEEE 802.1x compliant users
- IEEE 802.1x with restricted VLAN to provide limited services to users who are IEEE 802.1x compliant, but do not have the credentials to authenticate via the standard IEEE 802.1x processes.
- IEEE 802.1x accounting to track network usage
- IEEE 802.1x with wake-on-LAN to allow dormant PCs to be powered on based on the receipt of a specific Ethernet frame

- Network Admission Control (NAC) features:
  - NAC Layer 2 IEEE 802.1x validation of the antivirus condition or *posture* of endpoint systems or clients before granting the devices network access.

    For information about configuring NAC Layer 2 IEEE 802.1x validation, see the "Configuring NAC Layer 2 IEEE 802.1x Validation" section on page 8-38.
  - NAC Layer 2 IP validation to validate the posture of endpoint systems or clients before granting the devices network access.

    For information about configuring NAC Layer 2 IP validation, see the *Network Admission Control Software Configuration Guide*.
  - IEEE 802.1x inaccessible authentication bypass.

    For information about configuring this feature, see the "Configuring the Inaccessible Authentication Bypass Feature" section on page 8-34.
  - Authentication, authorization, and accounting (AAA) down policy for a NAC Layer 2 IP validation of a host if the AAA server is not available when the posture validation occurs.

    For information about this feature, see the *Configuring Network Admission Control Software Configuration Guide*.
- Network Admission Control (NAC) Layer 2 IEEE 802.1x validation to validate the antivirus condition or *posture* of endpoint systems or clients before granting the devices network access by using IEEE 802.1x port-based authentication on the network edge
- NAC Layer 2 IP validation to validate the posture of endpoint systems or clients before granting the devices network access by using UDP on the network edge
- TACACS +, a proprietary feature for managing network security through a TACACS server
- Kerberos security system to authenticate requests for network resources by using a trusted third party
- RADIUS, which provides detailed accounting information and flexible administrative control over authentication and authorization processes
- Secure Socket Layer (SSL) Version 3.0 support for the HTTP 1.1 server authentication, encryption, and message integrity, and HTTP client authentication to allow secure HTTP communications
- IEEE 802.1Q tunneling to allow customers with users at remote sites across a service provider network to keep VLANs segregated from other customers and Layer 2 protocol tunneling to ensure that the customer's network has complete STP, CDP, and VTP information about all users
- Layer 2 point-to-point tunneling to facilitate the automatic creation of EtherChannels

Quality of Service (QoS) and Class of Service (CoS)

- Automatic QoS (auto-QoS) to simplify the deployment of existing QoS features by classifying traffic and configuring egress queues
- Classification
  - Classification on a physical interface or on a per-port per-VLAN basis
  - IP type-of-service/Differentiated Services Code Point (IP TOS/DSCP) and IEEE 802.1P CoS marking priorities on a per-port basis for protecting the performance of mission-critical applications

  - IP TOS/DSCP and IEEE 802.1P CoS marking based on flow-based packet classification (classification based on information in the MAC, IP, and TCP/UDP headers) for high-performance quality of service at the network edge, allowing for differentiated service levels for different types of network traffic and for prioritizing mission-critical traffic in the network
  - Trusted port states (CoS, DSCP, and IP precedence) within a QoS domain and with a port bordering another QoS domain
  - Trusted boundary for detecting the presence of a Cisco IP phone, trusting the CoS value received, and ensuring port security
- Policing
  - Policing on a physical interface or on a per-port per-VLAN basis
  - Traffic-policing policies on the switch port for managing how much of the port bandwidth should be allocated to a specific traffic flow
  - Aggregate policing for policing traffic flows in aggregate to restrict specific applications or traffic flows to metered, predefined rates
  - Up to 128 policers on ingress Gigabit-capable Ethernet ports
    Up to eight policers on ingress 10/100 ports
    Up to eight policers per egress port (aggregate policers only)
- Out-of-Profile
  - Out-of-profile markdown for packets that exceed bandwidth utilization limits
- Egress Policing and Scheduling of Egress Queues
  - Four egress queues on all switch ports. These queues can either be configured with the Weighted Round Robin (WRR) scheduling algorithm or configured with one queue as a strict priority queue and the other three queues for WRR. The strict priority queue must be empty before the other three queues are serviced. You can use the strict priority queue for mission-critical and time-sensitive traffic.
  - Tail drop and Weight Random Early Detection (WRED) techniques for avoiding congestion on Gigabit Ethernet ports; tail drop for congestion avoidance on Fast Ethernet ports

Layer 3 Support

Some features and protocols require the enhanced multilayer software image.

- Hot Standby Router Protocol (HSRP) for Layer 3 router redundancy
- IP routing protocols for load balancing and for constructing scalable, routed backbones:
  - Routing Information Protocol (RIP) versions 1 and 2
  - Open Shortest Path First (OSPF)
  - Enhanced IGRP (EIGRP)
  - Border Gateway Protocol (BGP) Version 4
- IP routing between VLANs (inter-VLAN routing) for full Layer 3 routing between two or more VLANs, allowing each VLAN to maintain its own autonomous data-link domain
- Multiple VPN routing/forwarding (multi-VRF) instances in customer edge (CE) devices to allow service providers to support multiple virtual private networks (VPNs) and overlap IP addresses between VPNs.

- Policy-based routing (PBR) for configuring defined policies for traffic flows
- Fallback bridging for forwarding non-IP traffic between two or more VLANs
- Static IP routing for manually building a routing table of network path information
- Equal-cost routing for load balancing and redundancy
- Internet Control Message Protocol (ICMP) and ICMP Router Discovery Protocol (IRDP) for using router advertisement and router solicitation messages to discover the addresses of routers on directly attached subnets
- Protocol-Independent Multicast (PIM) for multicast routing within the network, allowing for devices in the network to receive the multicast feed requested and for switches not participating in the multicast to be pruned. Includes support for PIM sparse mode (PIM-SM), PIM dense mode (PIM-DM), and PIM sparse-dense mode.
- Distance Vector Multicast Routing Protocol (DVMRP) tunneling for interconnecting two multicast-enabled networks across non-multicast networks
- DHCP relay for forwarding UDP broadcasts, including IP address requests, from DHCP clients
- Nonstop forwarding (NSF) awareness to enable the Layer 3 switch to continue forwarding packets from an NSF-capable neighboring router during the interval while the primary route processor (RP) is crashing and the backup RP is taking over, or while the primary RP is manually reloaded for a nondisruptive software upgrade (requires the IP services image)

Monitoring

- Switch LEDs that provide port- and switch-level status
- Switched Port Analyzer (SPAN) and Remote SPAN (RSPAN) for traffic monitoring on any port or VLAN
- SPAN and RSPAN support of Intrusion Detection Systems (IDSs) to monitor, repel, and report network security violations
- Four groups (history, statistics, alarms, and events) of embedded remote monitoring (RMON) agents for network monitoring and traffic analysis
- Syslog facility for logging system messages about authentication or authorization errors, resource issues, and time-out events
- MAC address notification for tracking users on a network by storing the MAC addresses that the switch has learned or removed
- Layer 2 traceroute to identify the physical path that a packet takes from a source device to a destination device

Power over Ethernet Support for the Catalyst 3550-24PWR Switch

- Ability to provide power to connected Cisco pre-standard and IEEE 802.3af-compliant powered devices from Power over Ethernet (PoE)-capable ports if the switch detects that there is no power on the circuit.
- Support for CDP with power consumption. The powered device notifies the switch of the amount of power it is consuming.

- Support for Cisco intelligent power management. The powered device and the switch negotiate through power-negotiation CDP messages for an agreed power-consumption level. The negotiation allows a high-power Cisco powered device to operate at its highest power mode.
- Automatic detection and power budgeting; the switch maintains a power budget, monitors and tracks requests for power, and grants power only when it is available.
- Fan-fault and over-temperature detection through the device manager and Network Assistant

Management Options

The switch is designed for plug-and-play operation: you need to configure only basic IP information for the switch and connect it to the other devices in your network. If you have specific network needs, you can configure and monitor the switch—on an individual basis or as part of a switch cluster—through its various management interfaces.

Management Interface Options

You can configure and monitor individual switches and switch clusters by using these interfaces:

- An embedded device manager—The device manger is a GUI that is integrated in the software image. You use it to configure and to monitor a single switch. For more information about the device manager, see the switch online help.
- Network Assistant—Network Assistant is a GUI that can be downloaded from Cisco.com. You use it to manage a single switch or a cluster of switches. For more information about Network Assistant, see the *Getting Started with Cisco Network Assistant*, available on Cisco.com.
- CLI—The switch Cisco IOS software supports desktop- and multilayer-switching features. You can access the CLI either by connecting your management station directly to the switch console port or by using Telnet from a remote management station.

  For more information about the CLI, see Chapter 2, “Using the Command-Line Interface.”
- IE2100—Cisco Intelligence Engine 2100 Series Configuration Registrar is a network management device that works with embedded CNS Agents in the switch software. You can automate initial configurations and configuration updates by generating switch-specific configuration changes, sending them to the switch, executing the configuration change, and logging the results.

  For more information about IE2100, see Chapter 4, “Configuring Cisco IOS CNS Agents.”
- SNMP—SNMP provides a means to monitor and control the switch and switch cluster members. You can manage switch configuration settings, performance, security, and collect statistics by using SNMP management applications such as CiscoWorks2000 LAN Management Suite (LMS) and HP OpenView.

  You can manage the switch from an SNMP-compatible management station that is running platforms such as HP OpenView or SunNet Manager. The switch supports a comprehensive set of MIB extensions and four RMON groups.

  For more information about using SNMP, see Chapter 27, “Configuring SNMP.”

Advantages of Using Network Assistant and Clustering Switches

Using Network Assistant and switch clusters can simplify and minimize your configuration and monitoring tasks. You can use Cisco switch clustering technology to manage up to 16 interconnected, supported Catalyst switches through one IP address. This can conserve IP addresses if you have a limited number of them. Network Assistant is the easiest interface to use and makes switch and switch cluster management accessible to authorized users from any PC on your network.

By using switch clusters and Network Assistant, you can

- Manage and monitor interconnected Catalyst switches (see the release notes for a list of supported switches), regardless of their geographic proximity and interconnection media, including Ethernet, Fast Ethernet, Fast EtherChannel, Cisco GigaStack Gigabit Interface Converter (GBIC), Gigabit Ethernet, and Gigabit EtherChannel connections.
- Accomplish multiple configuration tasks from a single Network Assistant window without needing to remember CLI commands to accomplish specific tasks.
- Apply actions from Network Assistant to multiple ports and multiple switches at the same time. Here are some examples of configuring and managing multiple ports and switches:
  - Port configuration such as speed and duplex settings
  - Port and console port security settings
  - NTP, STP, VLAN, and QoS configurations
  - Inventory and statistic reporting and link- and switch-level monitoring and troubleshooting
  - Group software upgrades
- View a topology of interconnected devices to identify existing switch clusters and eligible switches that can join a cluster. You can also use the topology to quickly identify link information between switches.
- Monitor real-time status of a switch or multiple switches from the LEDs on the front-panel images. The system, redundant power system (RPS), and port LED colors on the images are similar to those used on the physical LEDs.
- Use an interactive mode that takes you step-by-step through configuring complex features such as VLANs, ACLs, and QoS.
- Use a wizard that prompts you to provide only the minimum required information to configure complex features such as QoS priorities for video traffic, priority levels for data applications, and security.

For the Network Assistant software and browser requirements, and for more information about clustering, see *Getting Started with Cisco Network Assistant*, available on Cisco.com. For clustering requirements, including supported Cisco IOS releases, see the release notes for this release.

Network Configuration Examples

This section provides network configuration concepts and includes examples of using the switch to create dedicated network segments and interconnecting the segments through Fast Ethernet and Gigabit Ethernet connections.

- "Design Concepts for Using the Switch" section on page 1-11
- "Small to Medium-Sized Network Using Mixed Switches" section on page 1-14
- "Large Network Using Only Catalyst 3550 Switches" section on page 1-15

- "Multidwelling Network Using Catalyst 3550 Switches" section on page 1-17
- "Long-Distance, High-Bandwidth Transport Configuration" section on page 1-18

Design Concepts for Using the Switch

As your network users compete for network bandwidth, it takes longer to send and receive data. When you configure your network, consider the bandwidth required by your network users and the relative priority of the network applications they use.

Table 1-1 describes what can cause network performance to degrade and how you can configure your network to increase the bandwidth available to your network users.

Table 1-1 Increasing Network Performance

Network Demands	Suggested Design Methods
Too many users on a single network segment and a growing number of users accessing the Internet	• Create smaller network segments so that fewer users share the bandwidth, and use VLANs and IP subnets to place the network resources in the same logical network as the users who access those resources most. • Use full-duplex operation between the switch and its connected workstations.
• Increased power of new PCs, workstations, and servers • High bandwidth demand from networked applications (such as e-mail with large attached files) and from bandwidth-intensive applications (such as multimedia)	• Connect global resources—such as servers and routers to which the network users require equal access—directly to the high-speed switch ports so that they have their own high-speed segment. • Use the EtherChannel feature between the switch and its connected servers and routers.

Bandwidth alone is not the only consideration when designing your network. As your network traffic profiles evolve, consider providing network services that can support applications for voice and data integration, multimedia integration, application prioritization, and security. Table 1-2 describes some network demands and how you can meet those demands.

Table 1-2 Providing Network Services

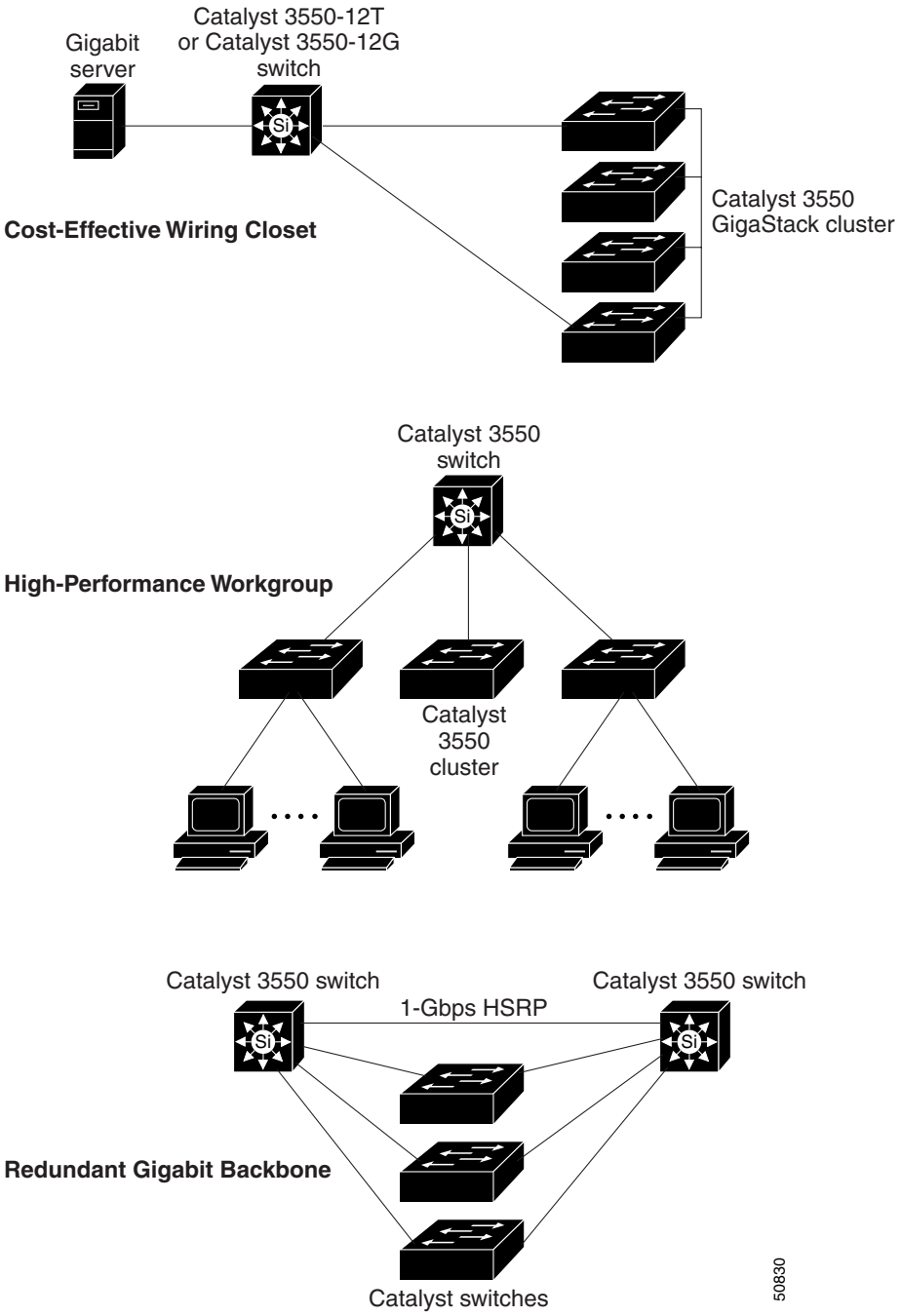
Network Demands	Suggested Design Methods
Efficient bandwidth usage for multimedia applications and guaranteed bandwidth for critical applications	• Use IGMP snooping to efficiently forward multimedia and multicast traffic. • Use other QoS mechanisms such as packet classification, marking, scheduling, and congestion avoidance to classify traffic with the appropriate priority level, thereby providing maximum flexibility and support for mission-critical, unicast, and multicast and multimedia applications. • Use optional IP multicast routing to design networks better suited for multicast traffic. • Use MVR to continuously send multicast streams in a multicast VLAN, but to isolate the streams from subscriber VLANs for bandwidth and security reasons.
High demand on network redundancy to provide *always on* mission-critical applications	• Use HSRP for router redundancy. • Use VLAN trunks, cross-stack UplinkFast, and BackboneFast for traffic-load balancing on the uplink ports so that the uplink port with a lower relative port cost is selected to carry the VLAN traffic.
An evolving demand for IP telephony	• Use QoS to prioritize applications such as IP telephony during congestion and to help control both delay and jitter within the network. • Use switches that support at least two queues per port to prioritize voice and data traffic as either high- or low-priority, based on IEEE 802.1P/Q. • Use voice VLAN IDs (VVIDs) on the Catalyst 2900 XL and 3500 XL switches to provide separate VLANs for voice traffic.
A growing demand for using existing infrastructure to transport data and voice from a home or office to the Internet or an intranet at higher speeds	Use the Catalyst 2900 LRE XL and Catalyst 2950 LRE switches to provide up to 15 Mb of IP connectivity over existing infrastructure, such as existing telephone lines. **Note** Long-Reach Ethernet (LRE) is the technology used in the Catalyst 2900 LRE XL and Catalyst 2950 LRE switches. See the switch documentation sets about these switches and the LRE technology.

Figure 1-1 shows three configuration examples of using Catalyst switches to create the following:

- Cost-effective wiring closet—A cost-effective way to connect many users to the wiring closet is to connect a Catalyst switch cluster of up to nine Catalyst 3550 XL switches (or with a mix of Catalyst 3550, Catalyst 2950, Catalyst 3500 XL, and Catalyst 2900 XL switches) through GigaStack GBIC connections. To preserve switch connectivity if one switch in the stack fails, connect the bottom switch to the top switch to create a GigaStack loopback, and enable cross-stack UplinkFast on the cross-stack Gigabit uplinks.

  You can have redundant uplink connections, using Gigabit GBIC modules, from the GigaStack cluster to a Gigabit backbone switch such as the Catalyst 3550-12T or Catalyst 3550-12G switch. You can also create backup paths by using Fast Ethernet, Gigabit, or EtherChannel links. If one of the redundant connections fails, the other can serve as a backup path. You can configure the Catalyst 3550-12T or Catalyst 3550-12G switch as a switch cluster manager to manage stack members through a single IP address. The Catalyst 3550-12T or Catalyst 3550-12G switch can be connected to a Gigabit server through a 1000BASE-T connection.

Figure 1-1 Example Configurations



- High-performance workgroup—For high-speed access to network resources, you can use Catalyst 3550 switches in the access layer to provide Gigabit Ethernet to the desktop. To prevent congestion, use QoS DSCP marking priorities on these switches. For high-speed IP forwarding at the distribution layer, connect the Catalyst 3550 switches in the access layer to a Gigabit multilayer switch (such as the Catalyst 3550 multilayer switch) in the backbone.

Each switch in this configuration provides users with a dedicated 1-Gbps connection to network resources in the backbone. Compare this with the switches in a GigaStack configuration, where the 1-Gbps connection is shared among the switches in the stack. Using these Gigabit GBIC modules also provides flexibility in media and distance options:

- 1000BASE-T GBIC: copper connections of up to 328 feet (100 m)
- 1000BASE-SX GBIC: fiber-optic connections of up to 1804 feet (550 m)
- 1000BASE-LX/LH GBIC: fiber-optic connections of up to 32,808 feet (6 miles or 10 km)
- 1000BASE-ZX GBIC: fiber-optic connections of up to 328,084 feet (62 miles or 100 km)

- Redundant Gigabit backbone—Using HSRP, you can create backup paths between two Catalyst 3550 multilayer switches to enhance network reliability and load balancing for different VLANs and subnets. Using HSRP also provides faster network convergence if any network failure occurs. You can connect the Catalyst switches, again in a star configuration, to two Catalyst 3550 multilayer backbone switches. If one of the backbone switches fails, the second backbone switch preserves connectivity between the switches and network resources.

Small to Medium-Sized Network Using Mixed Switches

Figure 1-2 shows a configuration for a network of up to 500 employees. This network uses Catalyst 3550 multilayer switches to aggregate up to ten wiring closets through high-speed uplinks. For network reliability and load balancing, this network includes two routers and two Catalyst 3550 multilayer switches, all with HSRP enabled. This ensures connectivity to the Internet, WAN, and mission-critical network resources if one of the routers or Catalyst 3550 multilayer switches fails.

The wiring closets have a mix of switches such as the Catalyst 3550, Catalyst 3500 XL, Catalyst 2950, Catalyst 2900 XL, Catalyst 2820, and Catalyst 1900 switches. These switches are connected to workstations, Cisco IP Phones, and local servers. You can cluster these switches into multiple clusters, as shown, or into a single cluster. You can manage a cluster through the IP address of its primary and secondary command switches, regardless of the geographic location of the cluster members.

This network uses VLANs to segment the network logically into well-defined broadcast groups and for security management. Data and multimedia traffic are configured on the same VLAN. Voice traffic from the Cisco IP Phones are configured on separate VVIDs. You can have up to four VVIDs per wiring closet. If data, multimedia, and voice traffic are assigned to the same VLAN, only one VLAN can be configured per wiring closet. For any switch port connected to Cisco IP Phones, IEEE 802.1P/Q QoS gives voice traffic forwarding-priority over data traffic.

Cisco IP Phones are connected—using standard straight-through, twisted-pair cable with RJ-45 connectors—to the 10/100 PoE ports on the Catalyst 3550-24PWR switches and to the 10/100 ports on the Catalyst 3550 switches. These multiservice switch ports automatically detect any IP phones that are connected. Cisco CallManager controls call processing, routing, and IP phone features and configuration. Users with workstations running Cisco SoftPhone software can place, receive, and control calls from their PCs. Using Cisco IP Phones, Cisco CallManager software, and Cisco SoftPhone software integrates telephony and IP networks, and the IP network supports both voice and data.

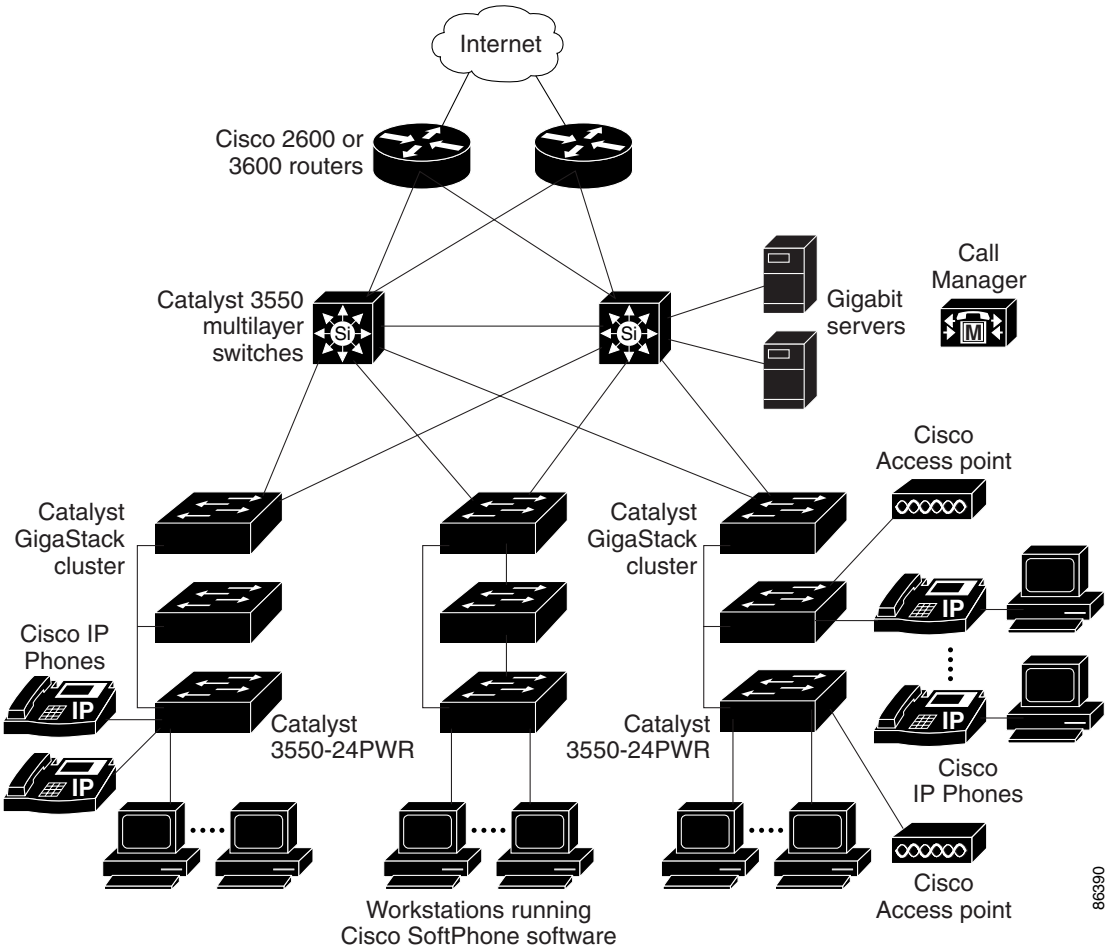
Each 10/100 PoE port on the Catalyst 3550-24PWR switches provides 15.4 W per port. The IP phone can receive redundant power when it is also connected to an AC power source. IP phones not connected to the Catalyst 3550-24PWR switches receive power from an AC power source.

When an end station in one VLAN needs to communicate with an end station in another VLAN, a router or multilayer switch routes the traffic to the appropriate destination VLAN. In this network, the Catalyst 3550 multilayer switches provide inter-VLAN routing. VLAN access control lists (VLAN maps) on the Catalyst 3550 switches provide intra-VLAN security and prevent unauthorized users from accessing critical pieces of the network.

In addition to inter-VLAN routing, the Catalyst 3550 multilayer switches provide QoS mechanisms such as DSCP priorities to prioritize the different types of network traffic and to deliver high-priority traffic in a predictable manner. If congestion occurs, QoS drops low-priority traffic to allow delivery of high-priority traffic.

With the Catalyst 3550 multilayer switches providing inter-VLAN routing and other network services, the routers focus on firewall services, Network Address Translation (NAT) services, voice-over-IP (VoIP) gateway services, and WAN and Internet access.

Figure 1-2 Catalyst 3550 Switches in a Collapsed Backbone Configuration

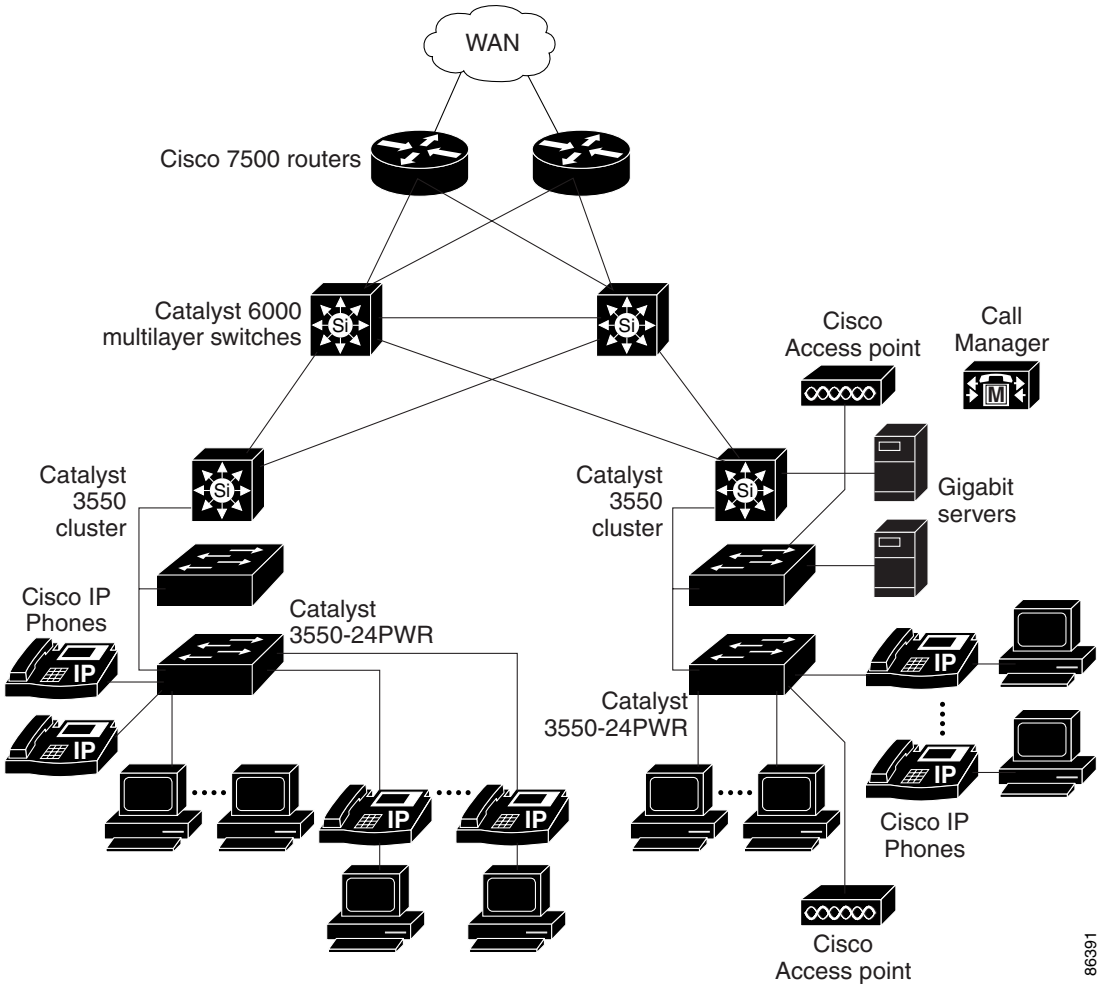


Large Network Using Only Catalyst 3550 Switches

Switches in the wiring closet have traditionally been Layer 2-only devices, but as network traffic profiles evolve, switches in the wiring closet are increasingly employing multilayer services such as multicast management and traffic classification. Figure 1-3 shows a configuration for a network exclusively using Catalyst 3550 multilayer switches in the wiring closets and a Catalyst 6000 switch in the backbone to aggregate up to ten wiring closets.

In the wiring closet, each Catalyst 3550 switch has IGMP snooping enabled to efficiently forward multimedia and multicast traffic. QoS ACLs that either drop or mark nonconforming traffic based on bandwidth limits are also configured on each switch. VLAN maps provide intra-VLAN security and prevent unauthorized users from accessing critical pieces of the network. QoS features can limit bandwidth on a per-port or per-user basis. The switch ports are configured as either trusted or untrusted. You can configure a trusted port to trust the CoS value, the DSCP value, or the IP precedence. If you configure the port as untrusted, you can use an ACL to mark the frame in accordance with the network policy.

Figure 1-3 Catalyst 3550 Switches in Wiring Closets in a Backbone Configuration



Within each wiring closet is a Catalyst 3550 multilayer switch for inter-VLAN routing. These switches provide proxy ARP services to determine IP and MAC address mapping, thereby removing this task from the routers and lessening this type of traffic on the WAN links. These switches also have redundant uplink connections to the backbone switches, with each uplink port configured as a trusted routed uplink to provide faster convergence in case of an uplink failure.

The routers and Catalyst 6000 multilayer backbone switches have HSRP enabled for load balancing and redundant connectivity to guarantee mission-critical traffic.

The Catalyst 6000 switch provides the workgroups with Gigabit access to core resources. The server farm includes a call-processing server running Cisco CallManager software. Cisco CallManager controls call processing, routing, and IP phone features and configuration.

Multidwelling Network Using Catalyst 3550 Switches

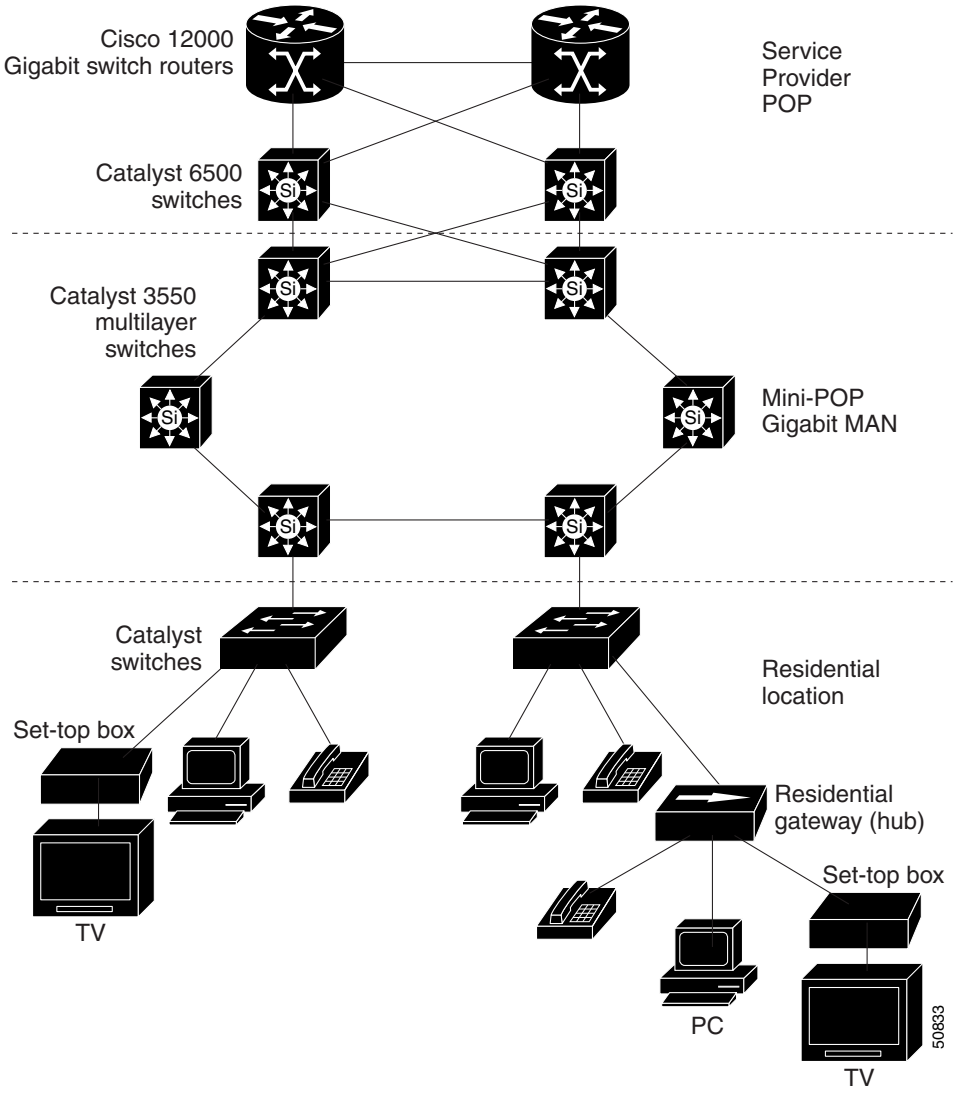
A growing segment of residential and commercial customers are requiring high-speed access to Ethernet metropolitan-area networks (MANs). Figure 1-4 shows a configuration for a Gigabit Ethernet MAN ring using Catalyst 3550 multilayer switches as aggregation switches in the mini-point-of-presence (POP) location. These switches are connected through 1000BASE-X GBIC ports.

The resident switches can be Catalyst 3550 switches, providing customers with high-speed connections to the MAN. Catalyst 2900 LRE XL or 2950 LRE Layer 2-only switches also can be used as residential switches for customers requiring connectivity through existing phone lines. The Catalyst LRE switches can then connect to another residential switch or to an aggregation switch.

All ports on the residential Catalyst 3550 switches (and Catalyst LRE switches if they are included) are configured as IEEE 802.1Q trunks with protected port and STP root guard features enabled. The protected port feature provides security and isolation between ports on the switch, ensuring that subscribers cannot view packets destined for other subscribers. STP root guard prevents unauthorized devices from becoming the STP root switch. All ports have IGMP snooping or CGMP enabled for multicast traffic management. ACLs on the uplink ports to the aggregating Catalyst 3550 multilayer switches provide security and bandwidth management.

The aggregating switches and routers provide services such as those described in the previous examples, "Small to Medium-Sized Network Using Mixed Switches" section on page 1-14 and "Large Network Using Only Catalyst 3550 Switches" section on page 1-15.

Figure 1-4 Catalyst 3550 Switches in a MAN Configuration



Long-Distance, High-Bandwidth Transport Configuration

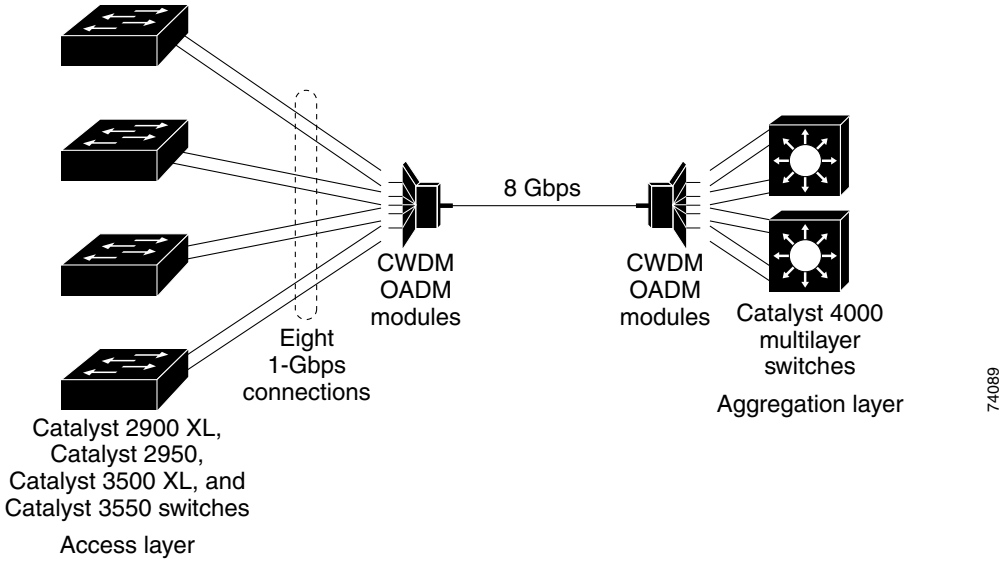
Figure 1-5 shows a configuration for transporting 8 Gigabits of data over a single fiber-optic cable. The Catalyst switches have Coarse Wave Division Multiplexer (CWDM) fiber-optic GBIC modules installed. Depending on the CWDM GBIC module, data is sent at wavelengths from 1470 nm to 1610 nm. The higher the wavelength, the farther the transmission can travel. A common wavelength used for long-distance transmissions is 1550 nm.

The CWDM GBIC modules connect to CWDM optical add/drop multiplexer (OADM) modules over distances of up to 393,701 feet (74.5 miles or 120 km). The CWDM OADM modules combine (or *multiplex*) the different CWDM wavelengths, allowing them to travel simultaneously on the same fiber-optic cable. The CWDM OADM modules on the receiving end separate (or *demultiplex*) the different wavelengths.

Using CWDM technology with the switches translates to farther data transmission and an increased bandwidth capacity (up to 8 Gbps) on a single fiber-optic cable.

For more information about the CWDM GBIC modules and CWDM OADM modules, see the *Installation Note for the CWDM Passive Optical System*.

Figure 1-5 Long-Distance, High-Bandwidth Transport Configuration



Where to Go Next

Before configuring the switch, review these sections for start up information:

- Chapter 2, "Using the Command-Line Interface"
- Chapter 3, "Assigning the Switch IP Address and Default Gateway"
- Chapter 4, "Configuring Cisco IOS CNS Agents"

CHAPTER 2

Using the Command-Line Interface

This chapter describes the Cisco IOS command-line interface (CLI) that you can use to configure your Catalyst 3550 switches. It contains these sections:

- Cisco IOS Command Modes, page 2-1
- Getting Help, page 2-3
- Abbreviating Commands, page 2-4
- Using no and default Forms of Commands, page 2-4
- Understanding CLI Messages, page 2-5
- Using Configuration Logging, page 2-5
- Using Command History, page 2-6
- Using Editing Features, page 2-7
- Searching and Filtering Output of show and more Commands, page 2-10
- Accessing the CLI, page 2-10

Cisco IOS Command Modes

The user interface is divided into many different modes. The commands available to you depend on which mode you are currently in. Enter a question mark (?) at the system prompt to obtain a list of commands available for each command mode.

When you start a session on the switch, you begin in user mode, often called user EXEC mode. Only a limited subset of the commands are available in user EXEC mode. For example, most of the user EXEC commands are one-time commands, such as **show** commands, which show the current configuration status, and **clear** commands, which clear counters or interfaces. The user EXEC commands are not saved when the switch reboots.

To have access to all commands, you must enter privileged EXEC mode. Normally, you must enter a password to enter privileged EXEC mode. From this mode, you can enter any privileged EXEC command or enter global configuration mode.

Using the configuration modes (global, interface, and line), you can make changes to the running configuration. If you save the configuration, these commands are stored and used when the switch reboots. To access the various configuration modes, you must start at global configuration mode. From global configuration mode, you can enter interface configuration mode and line configuration mode.

For information on accessing the CLI through the switch console port or through a Telnet session, see the hardware installation guide or the getting started guide.

Table 2-1 describes the main command modes, how to access each one, the prompt you see in that mode, and how to exit the mode. The examples in the table use the host name *Switch*.

Table 2-1 Command Mode Summary

Mode	Access Method	Prompt	Exit Method	About This Mode
User EXEC	Begin a session with your switch.	`Switch>`	Enter **logout** or **quit**.	Use this mode to • Change terminal settings. • Perform basic tests. • Display system information.
Privileged EXEC	While in user EXEC mode, enter the **enable** command.	`Switch#`	Enter **disable** to exit.	Use this mode to verify commands that you have entered. Use a password to protect access to this mode.
Global configuration	While in privileged EXEC mode, enter the **configure** command.	`Switch(config)#`	To exit to privileged EXEC mode, enter **exit** or **end**, or press **Ctrl-Z**.	Use this mode to configure parameters that apply to the entire switch.
Config-vlan	While in global configuration mode, enter the **vlan** *vlan-id* command.	`Switch(config-vlan)#`	To exit to global configuration mode, enter the **exit** command. To return to privileged EXEC mode, press **Ctrl-Z** or enter **end**.	Use this mode to configure VLAN parameters. When VTP mode is transparent, you can create extended-range VLANs (VLAN IDs greater than 1005) and save configurations in the switch startup configuration file.
VLAN configuration	While in privileged EXEC mode, enter the **vlan database** command.	`Switch(vlan)#`	To exit to privileged EXEC mode, enter **exit**.	Use this mode to configure VLAN parameters for VLANs 1 to 1005 in the VLAN database.

Table 2-1 Command Mode Summary (continued)

Mode	Access Method	Prompt	Exit Method	About This Mode
Interface configuration	While in global configuration mode, enter the **interface** command (with a specific interface).	`Switch(config-if)#`	To exit to global configuration mode, enter **exit**. To return to privileged EXEC mode, press **Ctrl-Z** or enter **end**.	Use this mode to configure parameters for the interfaces. To configure multiple interfaces with the same parameters, see the "Configuring a Range of Interfaces" section on page 9-10.
Line configuration	While in global configuration mode, specify a line with the **line vty** or **line console** command.	`Switch(config-line)#`	To exit to global configuration mode, enter **exit**. To return to privileged EXEC mode, press **Ctrl-Z** or enter **end**.	Use this mode to configure parameters for the terminal line.

Getting Help

You can enter a question mark (?) at the system prompt to display a list of commands available for each command mode. You can also obtain a list of associated keywords and arguments for any command, as shown in Table 2-2.

Table 2-2 Help Summary

Command	Purpose
help	Obtain a brief description of the help system in any command mode.

*abbreviated-command-entry***?**	Obtain a list of commands that begin with a particular character string.
	For example:
	`Switch# di?`
	`dir disable disconnect`

*abbreviated-command-entry***<Tab>**	Complete a partial command name.
	For example:
	`Switch# sh conf<tab>`
	`Switch# show configuration`

?	List all commands available for a particular command mode. For example: `Switch> ?`

Table 2-2 Help Summary (continued)

Command	Purpose
command **?**	List the associated keywords for a command. For example: `Switch> show ?`

command keyword **?**	List the associated arguments for a keyword.
	For example:
	`Switch(config)# cdp holdtime ?`
	`  <10-255>  Length of time (in sec) that receiver must keep this packet`

Abbreviating Commands

You have to enter only enough characters for the switch to recognize the command as unique. This example shows how to enter the **show configuration** privileged EXEC command:

```
Switch# show conf
```

Using no and default Forms of Commands

Almost every configuration command also has a **no** form. In general, use the **no** form to disable a feature or function or reverse the action of a command. For example, the **no shutdown** interface configuration command reverses the shutdown of an interface. Use the command without the keyword **no** to re-enable a disabled feature or to enable a feature that is disabled by default.

Configuration commands can also have a **default** form. The **default** form of a command returns the command setting to its default. Most commands are disabled by default, so the **default** form is the same as the **no** form. However, some commands are enabled by default and have variables set to certain default values. In these cases, the **default** command enables the command and sets variables to their default values.

Understanding CLI Messages

Table 2-3 lists some error messages that you might encounter while using the CLI to configure your switch.

Table 2-3 Common CLI Error Messages

Error Message	Meaning	How to Get Help
`% Ambiguous command: "show con"`	You did not enter enough characters for your switch to recognize the command.	Re-enter the command followed by a question mark (?) with a space between the command and the question mark. The possible keywords that you can enter with the command appear.
`% Incomplete command.`	You did not enter all the keywords or values required by this command.	Re-enter the command followed by a question mark (?) with a space between the command and the question mark. The possible keywords that you can enter with the command appear.
`% Invalid input detected at '^' marker.`	You entered the command incorrectly. The caret (^) marks the point of the error.	Enter a question mark (?) to display all the commands that are available in this command mode. The possible keywords that you can enter with the command appear.

Using Configuration Logging

Beginning with Cisco IOS Release 12.2(25)SEC, you can log and view changes to the switch configuration. You can use the Configuration Change Logging and Notification feature to track changes on a per-session and per-user basis. The logger tracks each configuration command that is applied, along with the user who entered the command, the time that the command was entered, and the parser return code for the command. This feature includes a mechanism for asynchronous notification to registered applications whenever the configuration changes. You can choose to have the notifications sent to the syslog.

For more information, see the *Configuration Change Notification and Logging* feature module at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123newft/123t/123t_4/gtconlog.htm

Note Only CLI or HTTP changes are logged.

Using Command History

The software provides a history or record of commands that you have entered. This feature is particularly useful for recalling long or complex commands or entries, including access lists. You can customize the command history feature to suit your needs as described in these sections:

- Changing the Command History Buffer Size, page 2-6
- Recalling Commands, page 2-6
- Disabling the Command History Feature, page 2-7

Changing the Command History Buffer Size

By default, the switch records ten command lines in its history buffer. Beginning in privileged EXEC mode, enter this command to change the number of command lines that the switch records during the current terminal session:

```
Switch# terminal history [size number-of-lines]
```

The range is from 0 to 256.

Beginning in line configuration mode, enter this command to configure the number of command lines the switch records for all sessions on a particular line:

```
Switch(config-line)# history [size number-of-lines]
```

The range is from 0 to 256.

Recalling Commands

To recall commands from the history buffer, perform one of the actions listed in Table 2-4:

Table 2-4 Recalling Commands

Action[1]	Result
Press **Ctrl-P** or the up arrow key.	Recall commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Press **Ctrl-N** or the down arrow key.	Return to more recent commands in the history buffer after recalling commands with **Ctrl-P** or the up arrow key. Repeat the key sequence to recall successively more recent commands.
show history	While in privileged EXEC mode, list the last several commands that you just entered. The number of commands that appear is determined by the setting of the **terminal history** global configuration command and **history** line configuration command.

1. The arrow keys function only on ANSI-compatible terminals such as VT100s.

Disabling the Command History Feature

The command history feature is automatically enabled.

To disable the feature during the current terminal session, enter the **terminal no history** privileged EXEC command.

To disable command history for the line, enter the **no history** line configuration command.

Using Editing Features

This section describes the editing features that can help you manipulate the command line. It contains these sections:

- Enabling and Disabling Editing Features, page 2-7
- Editing Commands through Keystrokes, page 2-7
- Editing Command Lines that Wrap, page 2-9

Enabling and Disabling Editing Features

Although enhanced editing mode is automatically enabled, you can disable it.

To re-enable the enhanced editing mode for the current terminal session, enter this command in privileged EXEC mode:

```
Switch# terminal editing
```

To reconfigure a specific line to have enhanced editing mode, enter this command in line configuration mode:

```
Switch(config-line)# editing
```

To globally disable enhanced editing mode, enter this command in line configuration mode:

```
Switch(config-line)# no editing
```

Editing Commands through Keystrokes

Table 2-5 shows the keystrokes that you need to edit command lines.

Table 2-5 Editing Commands through Keystrokes

Capability	Keystroke[1]	Purpose
Move around the command line to make changes or corrections.	Press **Ctrl-B**, or press the left arrow key.	Move the cursor back one character.
	Press **Ctrl-F**, or press the right arrow key.	Move the cursor forward one character.
	Press **Ctrl-A**.	Move the cursor to the beginning of the command line.
	Press **Ctrl-E**.	Move the cursor to the end of the command line.
	Press **Esc B**.	Move the cursor back one word.
	Press **Esc F**.	Move the cursor forward one word.
	Press **Ctrl-T**.	Transpose the character to the left of the cursor with the character located at the cursor.
Recall commands from the buffer and paste them in the command line. The switch provides a buffer with the last ten items that you deleted.	Press **Ctrl-Y.**	Recall the most recent entry in the buffer.
	Press **Esc Y**.	Recall the next buffer entry. The buffer contains only the last 10 items that you have deleted or cut. If you press **Esc Y** more than ten times, you cycle to the first buffer entry.
Delete entries if you make a mistake or change your mind.	Press the **Delete** or **Backspace** key.	Erase the character to the left of the cursor.
	Press **Ctrl-D**.	Delete the character at the cursor.
	Press **Ctrl-K**.	Delete all characters from the cursor to the end of the command line.
	Press **Ctrl-U** or **Ctrl-X**.	Delete all characters from the cursor to the beginning of the command line.
	Press **Ctrl-W**.	Delete the word to the left of the cursor.
	Press **Esc D**.	Delete from the cursor to the end of the word.
Capitalize or lowercase words or capitalize a set of letters.	Press **Esc C**.	Capitalize at the cursor.
	Press **Esc L**.	Change the word at the cursor to lowercase.
	Press **Esc U**.	Capitalize letters from the cursor to the end of the word.
Designate a particular keystroke as an executable command, perhaps as a shortcut.	Press **Ctrl-V** or **Esc Q**.	

Table 2-5 Editing Commands through Keystrokes (continued)

Capability	Keystroke[1]	Purpose
Scroll down a line or screen on displays that are longer than the terminal screen can display. **Note** The More prompt is used for any output that has more lines than can be displayed on the terminal screen, including **show** command output. You can use the **Return** and **Space** bar keystrokes whenever you see the More prompt.	Press the **Return** key.	Scroll down one line.
	Press the **Space** bar.	Scroll down one screen.
Redisplay the current command line if the switch suddenly sends a message to your screen.	Press **Ctrl-L** or **Ctrl-R.**	Redisplay the current command line.

1. The arrow keys function only on ANSI-compatible terminals such as VT100s.

Editing Command Lines that Wrap

You can use a wraparound feature for commands that extend beyond a single line on the screen. When the cursor reaches the right margin, the command line shifts ten spaces to the left. You cannot see the first ten characters of the line, but you can scroll back and check the syntax at the beginning of the command.

To scroll back to the beginning of the command entry, press **Ctrl-B** or the left arrow key repeatedly. You can also press **Ctrl-A** to immediately move to the beginning of the line.

Note The arrow keys function only on ANSI-compatible terminals such as VT100s.

In this example, the **access-list** global configuration command entry extends beyond one line. When the cursor first reaches the end of the line, the line is shifted ten spaces to the left and redisplayed. The dollar sign ($) shows that the line has been scrolled to the left. Each time the cursor reaches the end of the line, the line is again shifted ten spaces to the left.

```
Switch(config)# access-list 101 permit tcp 131.108.2.5 255.255.255.0 131.108.1
Switch(config)# $ 101 permit tcp 131.108.2.5 255.255.255.0 131.108.1.20 255.25
Switch(config)# $t tcp 131.108.2.5 255.255.255.0 131.108.1.20 255.255.255.0 eq
Switch(config)# $108.2.5 255.255.255.0 131.108.1.20 255.255.255.0 eq 45
```

After you complete the entry, press **Ctrl-A** to check the complete syntax before pressing the **Return** key to execute the command. The dollar sign ($) appears at the end of the line to show that the line has been scrolled to the right:

```
Switch(config)# access-list 101 permit tcp 131.108.2.5 255.255.255.0 131.108.1$
```

The software assumes you have a terminal screen that is 80 columns wide. If you have a width other than that, use the **terminal width** privileged EXEC command to set the width of your terminal.

Use line wrapping with the command history feature to recall and modify previous complex command entries. For information about recalling previous command entries, see the "Editing Commands through Keystrokes" section on page 2-7.

Searching and Filtering Output of show and more Commands

You can search and filter the output for **show** and **more** commands. This is useful when you need to sort through large amounts of output or if you want to exclude output that you do not need to see.

To use this functionality, enter a **show** or **more** command followed by the *pipe* character (|), one of the keywords **begin**, **include**, or **exclude**, and an expression that you want to search for or filter out:

command **|** {**begin** | **include** | **exclude**} *regular-expression*

Expressions are case sensitive. For example, if you enter **| exclude output**, the lines that contain *output* are not displayed, but the lines that contain *Output* appear.

This example shows how to include in the output display only lines where the expression *protocol* appears:

```
Switch# show interfaces | include protocol
Vlan1 is up, line protocol is up
Vlan10 is up, line protocol is down
GigabitEthernet0/1 is up, line protocol is down
GigabitEthernet0/2 is up, line protocol is up
```

Accessing the CLI

Before you can access the CLI, you need to connect a terminal or PC to the switch console port and power on the switch as described in the hardware installation guide that shipped with your switch. Then, to understand the boot process and the options available for assigning IP information, see Chapter 3, "Assigning the Switch IP Address and Default Gateway."

If your switch is already configured, you can access the CLI through a local console connection or through a remote Telnet session, but your switch must first be configured for this type of access. For more information, see the "Setting a Telnet Password for a Terminal Line" section on page 7-6.

You can establish a connection with the switch by either

- Connecting the switch console port to a management station or dial-up modem. For information about connecting to the console port, see the switch hardware installation guide.
- Using any Telnet TCP/IP or encrypted Secure Shell (SSH) package from a remote management station. The switch must have network connectivity with the Telnet or SSH client, and the switch must have an enable secret password configured.

  For information about configuring the switch for Telnet access, see the "Setting a Telnet Password for a Terminal Line" section on page 7-6. The switch supports up to 16 simultaneous Telnet sessions. Changes made by one Telnet user are reflected in all other Telnet sessions.

  For information about configuring the switch for SSH, see the "Configuring the Switch for Secure Shell" section on page 7-37. The switch supports up to five simultaneous secure SSH sessions.

After you connect through the console port, or through a Telnet session, or through an SSH session, the user EXEC prompt appears on the management station.

CHAPTER 3

Assigning the Switch IP Address and Default Gateway

This chapter describes how to create the initial switch configuration (for example, assign the switch IP address and default gateway information) for the Catalyst 3550 switch by using a variety of automatic and manual methods. It also describes how to modify the switch startup configuration.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release and the *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2*.

This chapter consists of these sections:

- Understanding the Boot Process, page 3-1
- Assigning Switch Information, page 3-2
- Checking and Saving the Running Configuration, page 3-10
- Modifying the Startup Configuration, page 3-11
- Scheduling a Reload of the Software Image, page 3-16

Understanding the Boot Process

To start your switch, you need to follow the procedures in the hardware installation guide about installing and powering on the switch, and setting up the initial configuration (IP address, subnet mask, default gateway, secret and Telnet passwords, and so forth) of the switch.

The normal boot process involves the operation of the boot loader software, which performs these activities:

- Performs low-level CPU initialization. It initializes the CPU registers, which control where physical memory is mapped, its quantity, its speed, and so forth.
- Performs power-on self-test (POST) for the CPU subsystem. It tests the CPU DRAM and the portion of the flash device that makes up the flash file system.
- Initializes the flash file system on the system board.
- Loads a default operating system software image into memory and boots the switch.

The boot loader provides access to the flash file system before the operating system is loaded. Normally, the boot loader is used only to load, uncompress, and launch the operating system. After the boot loader gives the operating system control of the CPU, the boot loader is not active until the next system reset or power on.

The boot loader also provides trap-door access into the system if the operating system has problems so serious that it cannot be used. The trap-door mechanism provides enough access to the system so that if it is necessary, you can format the flash file system, re-install the operating system software image by using the XMODEM Protocol, recover from a lost or forgotten password, and finally restart the operating system. For more information, see the "Recovering from a Software Failure" section on page 6-2 and the "Recovering from a Lost or Forgotten Password" section on page 6-2.

Note On Catalyst 3550 Fast Ethernet switches only, you can disable password recovery. For more information, see the "Disabling Password Recovery" section on page 7-5.

Before you can assign switch information, make sure you have connected a PC or terminal to the console port, and configured the PC or terminal-emulation software baud rate and character format to match these of the switch console port:

- Baud rate default is 9600.
- Data bits default is 8.
- Stop bits default is 1.
- Parity settings default is none.

Note If you are using Express Setup, do not connect any devices to the switch before starting Express Setup.

See your switch hardware installation guide for more information.

Assigning Switch Information

You can assign IP information through the switch Express Setup program, through the command-line-interface (CLI)-based setup program, through a DHCP server, or manually by using the CLI. If you are an experienced user familiar with the switch configuration steps, manually configure the switch. Otherwise, use one of the setup programs.

Note Your switch must be running Cisco IOS Release 12.1(14)EA1 or later to use the Express Setup program.

Use the switch Express Setup or CLI-based setup program if you want to be prompted for specific IP information. With these programs, you can also configure a default gateway, a host name, and a switch (enable secret) password. You also have the option of assigning a Telnet password (to provide security during remote management) and enabling Simple Network Management Protocol (SNMP). The CLI-based setup program also allows you to configure your switch as a command or member switch of a cluster or as a standalone switch. For more information about the Express Setup and CLI-based setup programs, see the hardware installation guide for your switch.

Use a DHCP server for centralized control and automatic assignment of IP information after the server is configured.

Note If you are using DHCP, do not respond to any of the questions in the setup program until the switch receives the dynamically assigned IP address and reads the configuration file.

This section has this configuration information:

- Default Switch Information, page 3-3
- Understanding DHCP-Based Autoconfiguration, page 3-3
- Configuring DHCP-Based Autoconfiguration, page 3-5
- Manually Assigning IP Information, page 3-10

Default Switch Information

Table 3-1 shows the default switch information.

Table 3-1 Default Switch Information

Feature	Default Setting
IP address and subnet mask	No IP address or subnet mask are defined.
Default gateway	No default gateway is defined.
Enable secret password	No password is defined.
Hostname	The factory-assigned default host name is *Switch*.
Telnet password	No password is defined.
Cluster command switch functionality	Disabled.
Cluster name	No cluster name is defined.

Understanding DHCP-Based Autoconfiguration

DHCP provides configuration information to Internet hosts and internetworking devices. This protocol consists of two components: one for delivering configuration parameters from a DHCP server to a device and a mechanism for allocating network addresses to devices. DHCP is built on a client-server model, in which designated DHCP servers allocate network addresses and deliver configuration parameters to dynamically configured devices. The switch can act as both a DHCP client and a DHCP server.

During DHCP-based autoconfiguration, your switch (DHCP client) is automatically configured at startup with IP address information and a configuration file.

Note The DHCP-base autoconfiguration only occurs when you place a switch with no configuration or a new switch on the network.

With DHCP-based autoconfiguration, no DHCP client-side configuration is needed on your switch. However, you need to configure the DHCP server for various lease options associated with IP addresses. If you are using DHCP to relay the configuration file location on the network, you might also need to configure a TFTP server and a Domain Name System (DNS) server.

The DHCP server for your switch can be on the same LAN or on a different LAN than the switch. If the DHCP server is running on a different LAN, you should configure a DHCP relay device between your switch and the DHCP server. A relay device forwards broadcast traffic between two directly connected LANs. A router does not forward broadcast packets, but it forwards packets based on the destination IP address in the received packet.

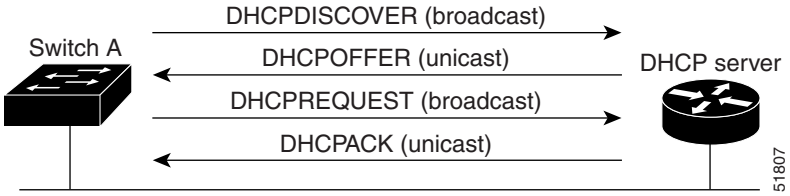
DHCP-based autoconfiguration replaces the BOOTP client functionality on your switch.

DHCP Client Request Process

When you boot your switch, the DHCP client is invoked and requests configuration information from a DHCP server when the configuration file is not present on the switch. If the configuration file is present and the configuration includes the **ip address dhcp** interface configuration command on specific routed interfaces, the DHCP client is invoked and requests the IP address information for those interfaces.

Figure 3-1 shows the sequence of messages that are exchanged between the DHCP client and the DHCP server.

Figure 3-1 DHCP Client and Server Message Exchange



The client, Switch A, broadcasts a DHCPDISCOVER message to locate a DHCP server. The DHCP server offers configuration parameters (such as an IP address, subnet mask, gateway IP address, DNS IP address, a lease for the IP address, and so forth) to the client in a DHCPOFFER unicast message.

In a DHCPREQUEST broadcast message, the client returns a formal request for the offered configuration information to the DHCP server. The formal request is broadcast so that all other DHCP servers that received the DHCPDISCOVER broadcast message from the client can reclaim the IP addresses that they offered to the client.

The DHCP server confirms that the IP address has been allocated to the client by returning a DHCPACK unicast message to the client. With this message, the client and server are bound, and the client uses configuration information received from the server. The amount of information the switch receives depends on how you configure the DHCP server. For more information, see the "DHCP Server Configuration Guidelines" section on page 3-5.

If the configuration parameters sent to the client in the DHCPOFFER unicast message are invalid (a configuration error exists), the client returns a DHCPDECLINE broadcast message to the DHCP server.

The DHCP server sends the client a DHCPNAK denial broadcast message, which means that the offered configuration parameters have not been assigned, that an error has occurred during the negotiation of the parameters, or that the client has been slow in responding to the DHCPOFFER message (the DHCP server assigned the parameters to another client).

A DHCP client might receive offers from multiple DHCP or BOOTP servers and can accept any of the offers; however, the client usually accepts the first offer it receives. The offer from the DHCP server is not a guarantee that the IP address is allocated to the client; however, the server usually reserves the address until the client has had a chance to formally request the address. If the switch accepts replies from a BOOTP server and configures itself, the switch broadcasts, instead of unicasts, TFTP requests to obtain the switch configuration file.

Configuring DHCP-Based Autoconfiguration

These sections describe how to configure DHCP-based autoconfiguration.

- DHCP Server Configuration Guidelines, page 3-5
- Configuring the TFTP Server, page 3-6
- Configuring the DNS, page 3-6
- Configuring the Relay Device, page 3-6
- Obtaining Configuration Files, page 3-7
- Example Configuration, page 3-8

If your DHCP server is a Cisco device, see the "Configuring DHCP" section of the "IP Addressing and Services" section of the *Cisco IOS IP Configuration Guide, Release 12.2* for additional information about configuring DHCP.

DHCP Server Configuration Guidelines

Follow these guidelines if you are configuring a device as a DHCP server:

The switch can act as both the DHCP client and the DHCP server. By default, the Cisco IOS DHCP server and relay agent features are enabled on your switch.

You should configure the DHCP server with reserved leases that are bound to each switch by the switch hardware address.

If you want the switch to receive IP address information, you must configure the DHCP server with these lease options:

- IP address of the client (required)
- Subnet mask of the client (required)
- DNS server IP address (optional)
- Router IP address (default gateway address to be used by the switch) (required)

If you want the switch to receive the configuration file from a TFTP server, you must configure the DHCP server with these lease options:

- TFTP server name (required)
- Boot filename (the name of the configuration file that the client needs) (recommended)
- Host name (optional)

Depending on the settings of the DHCP server, the switch can receive IP address information, the configuration file, or both.

If you do not configure the DHCP server with the lease options described previously, it replies to client requests with only those parameters that are configured. If the IP address and subnet mask are not in the reply, the switch is not configured. If the router IP address or TFTP server name are not found, the switch might send broadcast, instead of unicast, TFTP requests. Unavailability of other lease options does not affect autoconfiguration.

Configuring the TFTP Server

Based on the DHCP server configuration, the switch attempts to download one or more configuration files from the TFTP server. If you configured the DHCP server to respond to the switch with all the options required for IP connectivity to the TFTP server, and if you configured the DHCP server with a TFTP server name, address, and configuration filename, the switch attempts to download the specified configuration file from the specified TFTP server.

If you did not specify the configuration filename, the TFTP server, or if the configuration file could not be downloaded, the switch attempts to download a configuration file by using various combinations of filenames and TFTP server addresses. The files include the specified configuration filename (if any) and these files: network-config, cisconet.cfg, *hostname*.config, or *hostname*.cfg, where *hostname* is the switch's current hostname. The TFTP server addresses used include the specified TFTP server address (if any) and the broadcast address (255.255.255.255).

For the switch to successfully download a configuration file, the TFTP server must contain one or more configuration files in its base directory. The files can include these files:

- The configuration file named in the DHCP reply (the actual switch configuration file).
- The network-confg or the cisconet.cfg file (known as the default configuration files).
- The router-confg or the ciscortr.cfg file (These files contain commands common to all switches. Normally, if the DHCP and TFTP servers are properly configured, these files are not accessed.)

If you specify the TFTP server name in the DHCP server-lease database, you must also configure the TFTP server name-to-IP-address mapping in the DNS-server database.

If the TFTP server to be used is on a different LAN from the switch, or if it is to be accessed by the switch through the broadcast address (which occurs if the DHCP server response does not contain all the required information described previously), a relay must be configured to forward the TFTP packets to the TFTP server. For more information, see the "Configuring the Relay Device" section on page 3-6. The preferred solution is to configure the DHCP server with all the required information.

Configuring the DNS

The DHCP server uses the DNS server to resolve the TFTP server name to an IP address. You must configure the TFTP server name-to-IP address map on the DNS server. The TFTP server contains the configuration files for the switch.

You can configure the IP addresses of the DNS servers in the lease database of the DHCP server from where the DHCP replies will retrieve them. You can enter up to two DNS server IP addresses in the lease database.

The DNS server can be on the same or on a different LAN as the switch. If it is on a different LAN, the switch must be able to access it through a router.

Configuring the Relay Device

You must configure a relay device, also referred to an a relay agent, when a switch sends broadcast packets that require a response from a host on a different LAN. Examples of broadcast packets that the switch might send are DHCP, DNS, and in some cases, TFTP packets. You must configure this relay device to forward received broadcast packets on an interface to the destination host.

If the relay device is a Cisco router, enable IP routing (**ip routing** global configuration command), and configure helper addresses by using the **ip helper-address** interface configuration command.

For example, in Figure 3-2, configure the router interfaces as follows:

On interface 10.0.0.2:

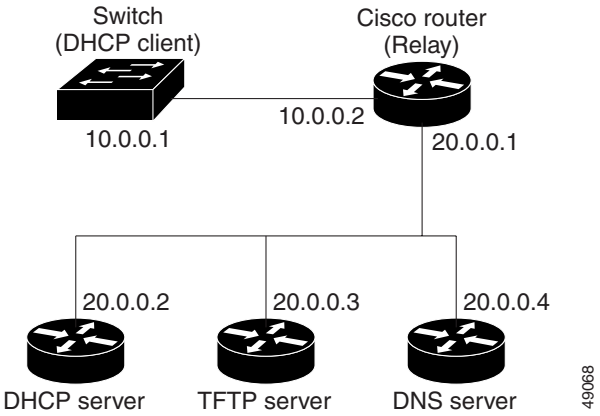
```
router(config-if)# ip helper-address 20.0.0.2
router(config-if)# ip helper-address 20.0.0.3
router(config-if)# ip helper-address 20.0.0.4
```

On interface 20.0.0.1

```
router(config-if)# ip helper-address 10.0.0.1
```

Note If the Catalyst 3550 multilayer switch is acting as the relay device, configure the interface as a routed port. For more information, see the "Routed Ports" section on page 9-4 and the "Configuring Layer 3 Interfaces" section on page 9-19.

Figure 3-2 Relay Device Used in Autoconfiguration



Obtaining Configuration Files

Depending on the availability of the IP address and the configuration filename in the DHCP reserved lease, the switch obtains its configuration information in these ways:

- The IP address and the configuration filename is reserved for the switch and provided in the DHCP reply (one-file read method).

  The switch receives its IP address, subnet mask, TFTP server address, and the configuration filename from the DHCP server. The switch sends a unicast message to the TFTP server to retrieve the named configuration file from the base directory of the server, and upon receipt, it completes its boot-up process.

- The IP address and the configuration filename is reserved for the switch, but the TFTP server address is not provided in the DHCP reply (one-file read method).

  The switch receives its IP address, subnet mask, and the configuration filename from the DHCP server. The switch sends a broadcast message to a TFTP server to retrieve the named configuration file from the base directory of the server, and upon receipt, it completes its boot-up process.

- Only the IP address is reserved for the switch and provided in the DHCP reply. The configuration filename is not provided (two-file read method).

The switch receives its IP address, subnet mask, and the TFTP server address from the DHCP server. The switch sends a unicast message to the TFTP server to retrieve the network-confg or cisconet.cfg default configuration file. (If the network-confg file cannot be read, the switch reads the cisconet.cfg file.)

The default configuration file contains the host names-to-IP-address mapping for the switch. The switch fills its host table with the information in the file and obtains its host name. If the host name is not found in the file, the switch uses the host name in the DHCP reply. If the host name is not specified in the DHCP reply, the switch uses the default *Switch* as its host name.

After obtaining its host name from the default configuration file or the DHCP reply, the switch reads the configuration file that has the same name as its host name (*hostname*-confg or *hostname*.cfg, depending on whether network-confg or cisconet.cfg was read earlier) from the TFTP server. If the cisconet.cfg file is read, the filename of the host is truncated to eight characters.

If the switch cannot read the network-confg, cisconet.cfg, or the hostname file, it reads the router-confg file. If the switch cannot read the router-confg file, it reads the ciscortr.cfg file.

Note The switch broadcasts TFTP server requests if the TFTP server is not obtained from the DHCP replies, if all attempts to read the configuration file through unicast transmissions fail, or if the TFTP server name cannot be resolved to an IP address.

Example Configuration

Figure 3-3 shows a sample network for retrieving IP information by using DHCP-based autoconfiguration.

Figure 3-3 DHCP-Based Autoconfiguration Network Example

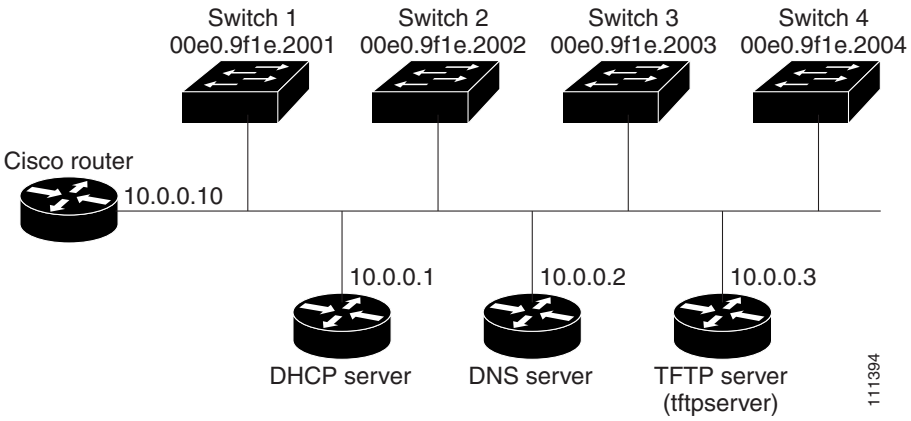


Table 3-2 shows the configuration of the reserved leases on the DHCP server.

Table 3-2 DHCP Server Configuration

	Switch-1	Switch-2	Switch-3	Switch-4
Binding key (hardware address)	00e0.9f1e.2001	00e0.9f1e.2002	00e0.9f1e.2003	00e0.9f1e.2004
IP address	10.0.0.21	10.0.0.22	10.0.0.23	10.0.0.24
Subnet mask	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0

Table 3-2 DHCP Server Configuration (continued)

	Switch-1	Switch-2	Switch-3	Switch-4
Router address	10.0.0.10	10.0.0.10	10.0.0.10	10.0.0.10
DNS server address	10.0.0.2	10.0.0.2	10.0.0.2	10.0.0.2
TFTP server name	*tftpserver* or *10.0.0.3*	*tftpserver* or *10.0.0.3*	*tftpserver* or *10.0.0.3*	*tftpserver* or *10.0.0.3*
Boot filename (configuration file) (optional)	switcha-confg	switchb-confg	switchc-confg	switchd-confg
Host name (optional)	switcha	switchb	switchc	switchd

DNS Server Configuration

The DNS server maps the TFTP server name *tftpserver* to IP address 10.0.0.3.

TFTP Server Configuration (on UNIX)

The TFTP server base directory is set to /tftpserver/work/. This directory contains the network-confg file used in the two-file read method. This file contains the host name to be assigned to the switch based on its IP address. The base directory also contains a configuration file for each switch (*switcha-confg*, *switchb-confg*, and so forth) as shown in this display:

```
prompt> cd /tftpserver/work/
prompt> ls
network-confg
switcha-confg
switchb-confg
switchc-confg
switchd-confg
prompt> cat network-confg
ip host switch1 10.0.0.21
ip host switch2 10.0.0.22
ip host switch3 10.0.0.23
ip host switch4 10.0.0.24
```

DHCP Client Configuration

No configuration file is present on Switch A through Switch D.

Configuration Explanation

In Figure 3-3, Switch A reads its configuration file as follows:

- It obtains its IP address 10.0.0.21 from the DHCP server.
- If no configuration filename is given in the DHCP server reply, Switch A reads the network-confg file from the base directory of the TFTP server.
- It adds the contents of the network-confg file to its host table.
- It reads its host table by indexing its IP address 10.0.0.21 to its host name (switcha).
- It reads the configuration file that corresponds to its host name; for example, it reads *switch1-confg* from the TFTP server.

Switches B through D retrieve their configuration files and IP addresses in the same way.

Manually Assigning IP Information

Beginning in privileged EXEC mode, follow these steps to manually assign IP information to a VLAN interface and to then designate that VLAN interface as the *management* VLAN:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface vlan** *vlan-id*	Enter interface configuration mode, and enter the VLAN to which the IP information is assigned. The range is 1 to 4094.
Step 3	**ip address** *ip-address subnet-mask*	Enter the IP address and subnet mask.
Step 4	**exit**	Return to global configuration mode.
Step 5	**ip default-gateway** *ip-address*	Enter the IP address of the next-hop router interface that is directly connected to the switch where a default gateway is being configured. The default gateway receives IP packets with unresolved destination IP addresses from the switch. Once the default gateway is configured, the switch has connectivity to the remote networks with which a host needs to communicate. **Note** When your switch is configured to route with IP, it does not need to have a default gateway set.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces vlan** *vlan-id*	Verify the configured IP address.
Step 8	**show ip redirects**	Verify the configured default gateway.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the switch IP address, use the **no ip address** interface configuration command. If you are removing the address through a Telnet session, your connection to the switch will be lost. To remove the default gateway address, use the **no ip default-gateway** global configuration command.

management For information on setting the switch system name, protecting access to privileged EXEC commands, and setting time and calendar services, see Chapter 6, "Administering the Switch."

Checking and Saving the Running Configuration

You can check the configuration settings you entered or changes that you made by entering the **show running-config** privileged EXEC command: For information about the output of this command, see the *Cisco IOS Configuration Fundamental Command Reference for Release 12*.1.

To store the configuration or changes you have made to your startup configuration in flash memory, enter the **copy running-config startup-config** privileged EXEC command. This command saves the configuration settings that you made. If you fail to do this, your configuration will be lost the next time you reload the system. To display information stored in the NVRAM section of flash memory, use the **show startup-config** or **more startup-config** privileged EXEC command.

For more information about alternative locations from which to copy the configuration file, see Appendix B, "Working with the Cisco IOS File System, Configuration Files, and Software Images."

Modifying the Startup Configuration

This section describes how to modify the switch startup configuration. It contains this configuration information:

- Default Boot Configuration, page 3-11
- Automatically Downloading a Configuration File, page 3-11
- Specifying the Filename to Read and Write the System Configuration, page 3-11
- Booting Manually, page 3-12
- Booting a Specific Software Image, page 3-13
- Controlling Environment Variables, page 3-13

Default Boot Configuration

Table 3-3 shows the default boot configuration.

Table 3-3 Default Boot Configuration

Feature	Default Setting
Operating system software image	The switch attempts to automatically boot the system using information in the BOOT environment variable. If the variable is not set, the switch attempts to load and execute the first executable image it can by performing a recursive, depth-first search throughout the flash file system. The software image is stored in a directory that has the same name as the image file (excluding the .bin extension). In a depth-first search of a directory, each encountered subdirectory is completely searched before continuing the search in the original directory.
Configuration file	Configured switches use the *config.text* file stored on the system board in flash memory. A new switch has no configuration file.

Automatically Downloading a Configuration File

You can automatically download a configuration file to your switch by using the DHCP-based autoconfiguration feature. For more information, see the "Understanding DHCP-Based Autoconfiguration" section on page 3-3.

Specifying the Filename to Read and Write the System Configuration

By default, the Cisco IOS software uses the file *config.text* to read and write a nonvolatile copy of the system configuration. However, you can specify a different filename that is loaded during the next boot cycle.

Beginning in privileged EXEC mode, follow these steps to specify a different configuration filename:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**boot config-file flash:/***file-url*	Specify the configuration file to load during the next boot cycle. For *file-url*, specify the path (directory) and the configuration filename. Filenames and directory names are case sensitive.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show boot**	Verify your entries. The **boot config-file** global configuration command changes the setting of the CONFIG_FILE environment variable.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no boot config-file** global configuration command.

Booting Manually

By default, the switch automatically boots; however, you can configure it to manually boot.

Beginning in privileged EXEC mode, follow these steps to configure the switch to manually boot during the next boot cycle:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**boot manual**	Enable the switch to manually boot during the next boot cycle.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show boot**	Verify your entries. The **boot manual** global command changes the setting of the MANUAL_BOOT environment variable. The next time you reboot the system, the switch is in boot loader mode, shown by the *switch:* prompt. To boot the system, use the **boot** *filesystem***:/***file-url* boot loader command. • For *filesystem***:**, use **flash:** for the system board flash device. • For *file-url*, specify the path (directory) and the name of the bootable image. Filenames and directory names are case sensitive.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable manual booting, use the **no boot manual** global configuration command.

Booting a Specific Software Image

By default, the switch attempts to automatically boot the system using information in the BOOT environment variable. If this variable is not set, the switch attempts to load and execute the first executable image it can by performing a recursive, depth-first search throughout the flash file system. In a depth-first search of a directory, each encountered subdirectory is completely searched before continuing the search in the original directory. However, you can specify a specific image to boot.

Beginning in privileged EXEC mode, follow these steps to configure the switch to boot a specific image during the next boot cycle:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**boot system** *filesystem***:/***file-url*	Configure the switch to boot a specific image in flash memory during the next boot cycle. • For *filesystem***:**, use **flash:** for the system board flash device. • For *file-url*, specify the path (directory) and the name of the bootable image. Filenames and directory names are case sensitive.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show boot**	Verify your entries. The **boot system** global command changes the setting of the BOOT environment variable. During the next boot cycle, the switch attempts to automatically boot the system using information in the BOOT environment variable.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no boot system** global configuration command.

Controlling Environment Variables

You enter the boot loader mode only through a switch console connection configured for 9600 bps. Unplug the switch power cord, and press the switch **Mode** button while reconnecting the power cord. Release the **Mode** button a second or two after the LED above port 1X turns off. Then the boot loader *switch:* prompt appears.

The switch boot loader software provides support for nonvolatile environment variables, which can be used to control how the boot loader, or any other software running on the system, behaves. Boot loader environment variables are similar to environment variables that can be set on UNIX or DOS systems.

Environment variables that have values are stored in the flash file system in various files as shown in Table 3-4.

Table 3-4 Environment Variables Storage Location

Environment Variable	Location (file system:filename)
BAUD, ENABLE_BREAK, CONFIG_BUFSIZE, CONFIG_FILE, MANUAL_BOOT, PS1	flash:env_vars
BOOT, BOOTHLPR, HELPER, HELPER_CONFIG_FILE	flash:system_env_vars

Each line in these files contains an environment variable name and an equal sign followed by the value of the variable. A variable has no value if it is not listed in this file; it has a value if it is listed in the file even if the value is a null string. A variable that is set to a null string (for example, “ ”) is a variable with a value. Many environment variables are predefined and have default values.

Environment variables store two kinds of data:

- Data that controls code, which does not read the Cisco IOS configuration file. For example, the name of a boot loader helper file, which extends or patches the functionality of the boot loader can be stored as an environment variable.
- Data that controls code, which is responsible for reading the Cisco IOS configuration file. For example, the name of the Cisco IOS configuration file can be stored as an environment variable.

You can change the settings of the environment variables by accessing the boot loader or by using Cisco IOS commands. It is not necessary to alter the setting of the environment variables.

Note For complete syntax and usage information for the boot loader commands and environment variables, see the command reference for this release.

Table 3-5 describes the function of the most common environment variables.

Table 3-5 Environment Variables

Variable	Boot Loader Command	Cisco IOS Global Configuration Command
MANUAL_BOOT	**set MANUAL_BOOT yes** Decides whether the switch automatically or manually boots. Valid values are 1, yes, 0, and no. If it is set to no or 0, the boot loader attempts to automatically boot the system. If it is set to anything else, you must manually boot the switch from the boot loader mode.	**boot manual** Enables manually booting the switch during the next boot cycle and changes the setting of the MANUAL_BOOT environment variable. The next time you reboot the system, the switch is in boot loader mode. To boot the system, use the **boot flash:***filesystem***:/***file-url* boot loader command, and specify the name of the bootable image.
BOOT	**set BOOT** *filesystem***:/***file-url* ... A semicolon-separated list of executable files to try to load and execute when automatically booting. If the BOOT environment variable is not set, the system attempts to load and execute the first executable image it can find by using a recursive, depth-first search through the flash file system. If the BOOT variable is set but the specified images cannot be loaded, the system attempts to boot the first bootable file that it can find in the flash file system.	**boot system** *filesystem***:/***file-url* Specifies the software image to load during the next boot cycle. This command changes the setting of the BOOT environment variable.
CONFIG_FILE	**set CONFIG_FILE flash:/***file-url* Changes the filename that the software uses to read and write a nonvolatile copy of the system configuration.	**boot config-file flash:/***file-url* Specifies the filename that the software uses to read and write a nonvolatile copy of the system configuration. This command changes the CONFIG_FILE environment variable.
CONFIG_BUFSIZE	**set CONFIG_BUFSIZE** *size* Changes the buffer size that the software uses to hold a copy of the configuration file in memory. The configuration file cannot be larger than the buffer size allocation. The range is from 4096 to 524288 bytes.	**boot buffersize** *size* Specifies the size of the file system-simulated NVRAM in flash memory. The buffer holds a copy of the configuration file in memory. This command changes the setting of the CONFIG_BUFSIZE environment variable. You must reload the switch by using the **reload** privileged EXEC command for this command to take effect.

Scheduling a Reload of the Software Image

You can schedule a reload of the software image to occur on the switch at a later time (for example, late at night or during the weekend when the switch is used less), or you can synchronize a reload network-wide (for example, to perform a software upgrade on all switches in the network).

Note A scheduled reload must take place within approximately 24 days.

Configuring a Scheduled Reload

To configure your switch to reload the software image at a later time, use one of these commands in privileged EXEC mode:

- **reload in** [*hh*:]*mm* [*text*]

  This command schedules a reload of the software to take affect in the specified minutes or hours and minutes. The reload must take place within approximately 24 days. You can specify the reason for the reload in a string up to 255 characters in length.

- **reload at** *hh*:*mm* [*month day* | *day month*] [*text*]

  This command schedules a reload of the software to take place at the specified time (using a 24-hour clock). If you specify the month and day, the reload is scheduled to take place at the specified time and date. If you do not specify the month and day, the reload takes place at the specified time on the current day (if the specified time is later than the current time) or on the next day (if the specified time is earlier than the current time). Specifying 00:00 schedules the reload for midnight.

  Note Use the **at** keyword only if the switch system clock has been set (through Network Time Protocol (NTP), the hardware calendar, or manually). The time is relative to the configured time zone on the switch. To schedule reloads across several switches to occur simultaneously, the time on each switch must be synchronized with NTP.

The **reload** command halts the system. If the system is not set to manually boot, it reboots itself. Use the **reload** command after you save the switch configuration information to the startup configuration (**copy running-config startup-config**).

If your switch is configured for manual booting, do not reload it from a virtual terminal. This restriction prevents the switch from entering the boot loader mode and thereby taking it from the remote user's control.

If you modify your configuration file, the switch prompts you to save the configuration before reloading. During the save operation, the system requests whether you want to proceed with the save if the CONFIG_FILE environment variable points to a startup configuration file that no longer exists. If you proceed in this situation, the system enters setup mode upon reload.

This example shows how to reload the software on the switch on the current day at 7:30 p.m:

```
Switch# reload at 19:30
Reload scheduled for 19:30:00 UTC Wed Jun 5 1996 (in 2 hours and 25 minutes)
Proceed with reload? [confirm]
```

This example shows how to reload the software on the switch at a future time:

```
Switch# reload at 02:00 jun 20
Reload scheduled for 02:00:00 UTC Thu Jun 20 1996 (in 344 hours and 53 minutes)
Proceed with reload? [confirm]
```

To cancel a previously scheduled reload, use the **reload cancel** privileged EXEC command.

Displaying Scheduled Reload Information

To display information about a previously scheduled reload or to determine if a reload has been scheduled on the switch, use the **show reload** privileged EXEC command.

It displays reload information including the time the reload is scheduled to occur and the reason for the reload (if it was specified when the reload was scheduled).

CHAPTER 4

Configuring Cisco IOS CNS Agents

This chapter describes how to configure the Cisco IOS CNS agents on the Catalyst 3550 switch.

Note For complete configuration information for the Cisco Configuration Engine, see this URL on Cisco.com http://www.cisco.com/en/US/products/sw/netmgtsw/ps4617/tsd_products_support_series_home.html

For complete syntax and usage information for the commands used in this chapter, see the *Cisco IOS Network Management Command Reference, Release 12.4* at this URL on Cisco.com: http://www.cisco.com/en/US/products/ps6350/products_command_reference_book09186a008042df72.html

This chapter consists of these sections:

- Understanding Cisco Configuration Engine Software, page 4-1
- Understanding Cisco IOS Agents, page 4-5
- Configuring Cisco IOS Agents, page 4-6
- Displaying CNS Configuration, page 4-15

Understanding Cisco Configuration Engine Software

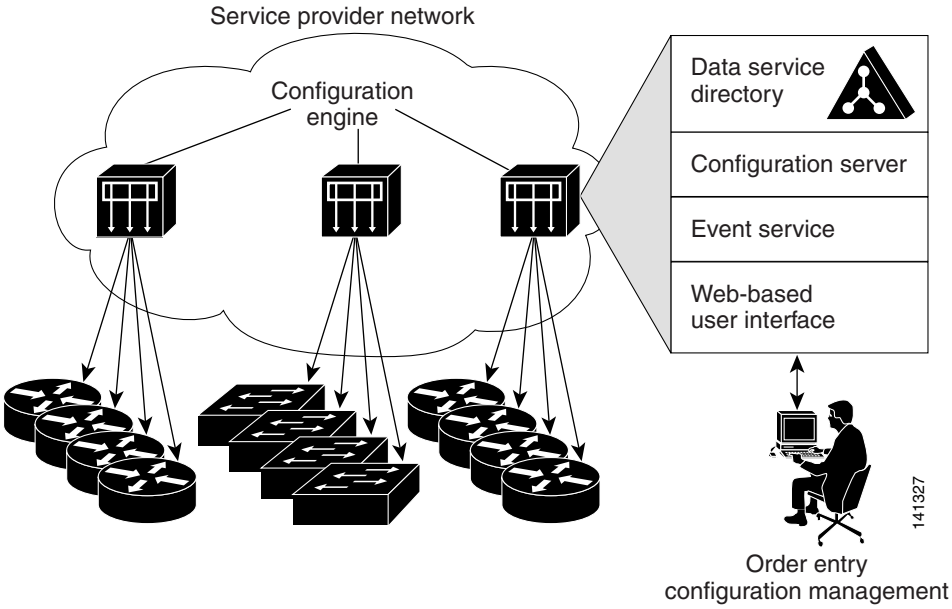
The Cisco Configuration Engine is network management software that acts as a configuration service for automating the deployment, management, and upgrading of network devices and services (see Figure 4-1). Each Configuration Engine manages a group of Cisco devices (switches and routers) and the services that they deliver, storing their configurations and delivering them as needed. The Configuration Engine automates initial configurations and configuration updates by generating device-specific configuration changes, sending them to the device, executing the configuration change, and logging the results.

The Configuration Engine supports standalone and server modes and has these CNS components:

- Configuration service (web server, file manager, and namespace mapping server)
- Event service (event gateway)
- Data service directory (data models and schema)

In standalone mode, the Configuration Engine supports an embedded Directory Service. In this mode, no external directory or other data store is required. In server mode, the Configuration Engine supports the use of a user-defined external directory.

Figure 4-1 Configuration Engine Architectural Overview



These sections contain this conceptual information:

- Configuration Service, page 4-2
- Event Service, page 4-3
- What You Should Know About the CNS IDs and Device Hostnames, page 4-3

Configuration Service

The Configuration Service is the core component of the Cisco Configuration Engine. It consists of a configuration server that works with Cisco IOS CNS agents on the switch. The Configuration Service delivers device and service configurations to the switch for initial configuration and mass reconfiguration by logical groups. Switches receive their initial configuration from the Configuration Service when they start up on the network for the first time.

The Configuration Service uses the CNS Event Service to send and receive configuration change events and to send success and failure notifications.

The configuration server is a web server that uses configuration templates and the device-specific configuration information stored in the embedded (standalone mode) or remote (server mode) directory.

Configuration templates are text files containing static configuration information in the form of CLI commands. In the templates, variables are specified using Lightweight Directory Access Protocol (LDAP) URLs that reference the device-specific configuration information stored in a directory.

The Cisco IOS agent can perform a syntax check on received configuration files and publish events to show the success or failure of the syntax check. The configuration agent can either apply configurations immediately or delay the application until receipt of a synchronization event from the configuration server.

Event Service

The Cisco Configuration Engine uses the Event Service for receipt and generation of configuration events. The event agent is on the switch and facilitates the communication between the switch and the event gateway on the Configuration Engine.

The Event Service is a highly capable publish-and-subscribe communication method. The Event Service uses subject-based addressing to send messages to their destinations. Subject-based addressing conventions define a simple, uniform namespace for messages and their destinations.

NameSpace Mapper

The Configuration Engine includes the NameSpace Mapper (NSM) that provides a lookup service for managing logical groups of devices based on application, device or group ID, and event.

Cisco IOS devices recognize only event subject-names that match those configured in Cisco IOS software; for example, cisco.cns.config.load. You can use the namespace mapping service to designate events by using any desired naming convention. When you have populated your data store with your subject names, NSM changes your event subject-name strings to those known by Cisco IOS.

For a subscriber, when given a unique device ID and event, the namespace mapping service returns a set of events to which to subscribe. Similarly, for a publisher, when given a unique group ID, device ID, and event, the mapping service returns a set of events on which to publish.

What You Should Know About the CNS IDs and Device Hostnames

The Configuration Engine assumes that a unique identifier is associated with each configured switch. This unique identifier can take on multiple synonyms, where each synonym is unique within a particular namespace. The event service uses namespace content for subject-based addressing of messages.

The Configuration Engine intersects two namespaces, one for the event bus and the other for the configuration server. Within the scope of the configuration server namespace, the term *ConfigID* is the unique identifier for a device. Within the scope of the event bus namespace, the term *DeviceID* is the CNS unique identifier for a device.

Because the Configuration Engine uses both the event bus and the configuration server to provide configurations to devices, you must define both ConfigID and Device ID for each configured switch.

Within the scope of a single instance of the configuration server, no two configured switches can share the same value for ConfigID. Within the scope of a single instance of the event bus, no two configured switches can share the same value for DeviceID.

ConfigID

Each configured switch has a unique ConfigID, which serves as the key into the Configuration Engine directory for the corresponding set of switch CLI attributes. The ConfigID defined on the switch must match the ConfigID for the corresponding switch definition on the Configuration Engine.

The ConfigID is fixed at startup time and cannot be changed until the device restarts, even if the switch hostname is reconfigured.

DeviceID

Each configured switch participating on the event bus has a unique DeviceID, which is analogous to the switch source address so that the switch can be targeted as a specific destination on the bus. All switches configured with the **cns config partial** global configuration command must access the event bus. Therefore, the DeviceID, as originated on the switch, must match the DeviceID of the corresponding switch definition in the Configuration Engine.

The origin of the DeviceID is defined by the Cisco IOS hostname of the switch. However, the DeviceID variable and its usage reside within the event gateway adjacent to the switch.

The logical Cisco IOS termination point on the event bus is embedded in the event gateway, which in turn functions as a proxy on behalf of the switch. The event gateway represents the switch and its corresponding DeviceID to the event bus.

The switch declares its hostname to the event gateway immediately after the successful connection to the event gateway. The event gateway couples the DeviceID value to the Cisco IOS hostname each time this connection is established. The event gateway caches this DeviceID value for the duration of its connection to the switch.

Hostname and DeviceID

The DeviceID is fixed at the time of the connection to the event gateway and does not change even when the switch hostname is reconfigured.

When changing the switch hostname on the switch, the only way to refresh the DeviceID is to break the connection between the switch and the event gateway. Enter the **no cns event** global configuration command followed by the **cns event** global configuration command.

When the connection is re-established, the switch sends its modified hostname to the event gateway. The event gateway redefines the DeviceID to the new value.

Caution When using the Configuration Engine user interface, you must first set the DeviceID field to the hostname value that the switch acquires *after*–not *before*–you use the **cns config initial** global configuration command at the switch. Otherwise, subsequent **cns config partial** global configuration command operations malfunction.

Using Hostname, DeviceID, and ConfigID

In standalone mode, when a hostname value is set for a switch, the configuration server uses the hostname as the DeviceID when an event is sent on hostname. If the hostname has not been set, the event is sent on the cn=<*value*> of the device.

In server mode, the hostname is not used. In this mode, the unique DeviceID attribute is always used for sending an event on the bus. If this attribute is not set, you cannot update the switch.

These and other associated attributes (tag value pairs) are set when you run **Setup** on the Configuration Engine.

Note For more information about running the setup program on the Configuration Engine, see the Configuration Engine setup and configuration guide at this URL on Cisco.com:
http://www.cisco.com/en/US/products/sw/netmgtsw/ps4617/prod_installation_guides_list.html

Understanding Cisco IOS Agents

The CNS event agent feature allows the switch to publish and subscribe to events on the event bus and works with the Cisco IOS agent. The Cisco IOS agent feature supports the switch by providing these features:

- Initial Configuration, page 4-5
- Incremental (Partial) Configuration, page 4-6
- Synchronized Configuration, page 4-6

Initial Configuration

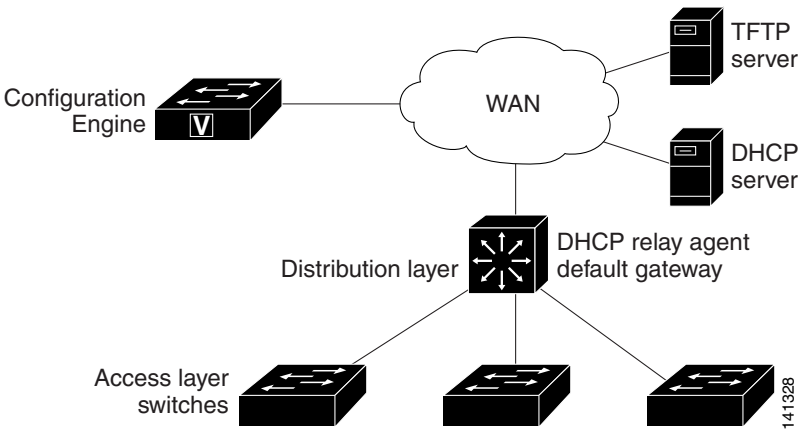
When the switch first comes up, it attempts to get an IP address by broadcasting a DHCP request on the network. Assuming there is no DHCP server on the subnet, the distribution switch acts as a DHCP relay agent and forwards the request to the DHCP server. Upon receiving the request, the DHCP server assigns an IP address to the new switch and includes the TFTP server IP address, the path to the bootstrap configuration file, and the default gateway IP address in a unicast reply to the DHCP relay agent. The DHCP relay agent forwards the reply to the switch.

The switch automatically configures the assigned IP address on interface VLAN 1 (the default) and downloads the bootstrap configuration file from the TFTP server. Upon successful download of the bootstrap configuration file, the switch loads the file in its running configuration.

The Cisco IOS agents initiate communication with the Configuration Engine by using the appropriate ConfigID and EventID. The Configuration Engine maps the Config ID to a template and downloads the full configuration file to the switch.

Figure 4-2 shows a sample network configuration for retrieving the initial bootstrap configuration file by using DHCP-based autoconfiguration.

Figure 4-2 Initial Configuration Overview



Incremental (Partial) Configuration

After the network is running, new services can be added by using the Cisco IOS agent. Incremental (partial) configurations can be sent to the switch. The actual configuration can be sent as an event payload by way of the event gateway (push operation) or as a signal event that triggers the switch to initiate a pull operation.

The switch can check the syntax of the configuration before applying it. If the syntax is correct, the switch applies the incremental configuration and publishes an event that signals success to the configuration server. If the switch does not apply the incremental configuration, it publishes an event showing an error status. When the switch has applied the incremental configuration, it can write it to NVRAM or wait until signaled to do so.

Synchronized Configuration

When the switch receives a configuration, it can defer application of the configuration upon receipt of a write-signal event. The write-signal event tells the switch not to save the updated configuration into its NVRAM. The switch uses the updated configuration as its running configuration. This ensures that the switch configuration is synchronized with other network activities before saving the configuration in NVRAM for use at the next reboot.

Configuring Cisco IOS Agents

The Cisco IOS agents embedded in the switch Cisco IOS software allow the switch to be connected and automatically configured as described in the "Enabling Automated CNS Configuration" section on page 4-6. If you want to change the configuration or install a custom configuration, see these sections for instructions:

- Enabling the CNS Event Agent, page 4-7
- Enabling the Cisco IOS CNS Agent, page 4-9

Enabling Automated CNS Configuration

To enable automated CNS configuration of the switch, you must first complete the prerequisites in Table 4-1. When you complete them, power on the switch. At the **setup** prompt, do nothing: The switch begins the initial configuration as described in the "Initial Configuration" section on page 4-5. When the full configuration file is loaded on your switch, you need to do nothing else.

Table 4-1 Prerequisites for Enabling Automatic Configuration

Device	Required Configuration
Access switch	Factory default (no configuration file)
Distribution switch	• IP helper address • Enable DHCP relay agent • IP routing (if used as default gateway)

Table 4-1 Prerequisites for Enabling Automatic Configuration (continued)

Device	Required Configuration
DHCP server	• IP address assignment • TFTP server IP address • Path to bootstrap configuration file on the TFTP server • Default gateway IP address
TFTP server	• A bootstrap configuration file that includes the CNS configuration commands that enable the switch to communicate with the Configuration Engine • The switch configured to use either the switch MAC address or the serial number (instead of the default hostname) to generate the ConfigID and EventID • The CNS event agent configured to push the configuration file to the switch
CNS Configuration Engine	One or more templates for each type of device, with the ConfigID of the device mapped to the template.

Note For more information about running the setup program and creating templates on the Configuration Engine, see the *Cisco Configuration Engine Installation and Setup Guide, 1.5 for Linux* at this URL: http://www.cisco.com/en/US/products/sw/netmgtsw/ps4617/products_installation_and_configuration_guide_book09186a00803b59db.html

Enabling the CNS Event Agent

Note You must enable the CNS event agent on the switch before you enable the CNS configuration agent.

Beginning in privileged EXEC mode, follow these steps to enable the CNS event agent on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**cns event** {*hostname* | *ip-address*} [*port-number*] [**backup**] [**failover-time** *seconds*] [**keepalive** *seconds retry-count*] [**reconnect** *time*] [**source** *ip-address*]	Enable the event agent, and enter the gateway parameters. • For {*hostname* | *ip-address*}, enter either the hostname or the IP address of the event gateway. • (Optional) For *port number*, enter the port number for the event gateway. The default port number is 11011. • (Optional) Enter **backup** to show that this is the backup gateway. (If omitted, this is the primary gateway.) • (Optional) For **failover-time** *seconds*, enter how long the switch waits for the primary gateway route after the route to the backup gateway is established. • (Optional) For **keepalive** *seconds*, enter how often the switch sends keepalive messages. For *retry-count*, enter the number of unanswered keepalive messages that the switch sends before the connection is terminated. The default for each is 0. • (Optional) For **reconnect** *time*, enter the maximum time interval that the switch waits before trying to reconnect to the event gateway. • (Optional) For **source** *ip-address*, enter the source IP address of this device. **Note** Though visible in the command-line help string, the **encrypt** and the **clock-timeout** *time* keywords are not supported.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show cns event connections**	Verify information about the event agent.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the CNS event agent, use the **no cns event** {*ip-address* | *hostname*} global configuration command.

This example shows how to enable the CNS event agent, set the IP address gateway to 10.180.1.27, set 120 seconds as the keepalive interval, and set 10 as the retry count.

```
Switch(config)# cns event 10.180.1.27 keepalive 120 10
```

Enabling the Cisco IOS CNS Agent

After enabling the CNS event agent, start the Cisco IOS CNS agent on the switch. You can enable the Cisco IOS agent with these commands:

- The **cns config initial** global configuration command enables the Cisco IOS agent and initiates an initial configuration on the switch.
- The **cns config partial** global configuration command enables the Cisco IOS agent and initiates a partial configuration on the switch. You can then use the Configuration Engine to remotely send incremental configurations to the switch.

Enabling an Initial Configuration

Beginning in privileged EXEC mode, follow these steps to enable the CNS configuration agent and initiate an initial configuration on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**cns template connect** *name*	Enter CNS template connect configuration mode, and specify the name of the CNS connect template.
Step 3	**cli** *config-text*	Enter a command line for the CNS connect template. Repeat this step for each command line in the template.
Step 4		Repeat Steps 2 to 3 to configure another CNS connect template.
Step 5	**exit**	Return to global configuration mode.
Step 6	**cns connect** *name* [**retries** *number*] [**retry-interval** *seconds*] [**sleep** *seconds*] [**timeout** *seconds*]	Enter CNS connect configuration mode, specify the name of the CNS connect profile, and define the profile parameters. The switch uses the CNS connect profile to connect to the Configuration Engine. • Enter the name of the CNS connect profile. • (Optional) For **retries** *number*, enter the number of connection retries. The range is 1 to 30. The default is 3. • (Optional) For **retry-interva**l *seconds*, enter the interval between successive connection attempts to the Configuration Engine. The range is 1 to 40 seconds. The default is 10 seconds. • (Optional) For **sleep** *seconds*, enter the amount of time before which the first connection attempt occurs. The range is 0 to 250 seconds. The default is 0. • (Optional) For **timeout** *seconds*, enter the amount of time after which the connection attempts end. The range is 0 to *tbd* seconds. The default is 120.

	Command	Purpose
Step 7	**discover** {**controller** *controller-type* | **dlci** [**subinterface** *subinterface-number*] | **interface** [*interface-type*] | **line** *line-type*}	Specify the interface parameters in the CNS connect profile. • For **controller** *controller-type*, enter the controller type. • For **dlci**, enter the active data-link connection identifiers (DLCIs). (Optional) For **subinterface** *subinterface-number*, specify the point-to-point subinterface number that is used to search for active DLCIs. • For **interface** [*interface-type*], enter the type of interface. • For **line** *line-type*, enter the line type.
Step 8	**template** *name* [... *name*]	Specify the list of CNS connect templates in the CNS connect profile to be applied to the switch configuration. You can specify more than one template.
Step 9		Repeat Steps 7 to 8 to specify more interface parameters and CNS connect templates in the CNS connect profile.
Step 10	**exit**	Return to global configuration mode.
Step 11	**hostname** *name*	Enter the hostname for the switch.
Step 12	**ip route** *network-number*	(Optional) Establish a static route to the Configuration Engine whose IP address is *network-number*.

	Command	Purpose
Step 13	**cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**] or **cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**]	(Optional) Set the unique EventID or ConfigID used by the Configuration Engine. • For *interface num*, enter the type of interface–for example, ethernet, group-async, loopback, or virtual-template. This setting specifies from which interface the IP or MAC address should be retrieved to define the unique ID. • For {**dns-reverse** | **ipaddress** | **mac-address**}, enter **dns-reverse** to retrieve the hostname and assign it as the unique ID, enter **ipaddress** to use the IP address, or enter **mac-address** to use the MAC address as the unique ID. • (Optional) Enter **event** to set the ID to be the event-id value used to identify the switch. • (Optional) Enter **image** to set the ID to be the image-id value used to identify the switch. **Note** If both the **event** and **image** keywords are omitted, the image-id value is used to identify the switch. • For {**hardware-serial** | **hostname**| **string** *string* | **udi**}, enter **hardware-serial** to set the switch serial number as the unique ID, enter **hostname** (the default) to select the switch hostname as the unique ID, enter an arbitrary text string for **string** *string* as the unique ID, or enter **udi** to set the unique device identifier (UDI) as the unique ID.

	Command	Purpose
Step 14	**cns config initial** {*hostname* | *ip-address*} [*port-number*] [**event**] [**no-persist**] [**page** *page*] [**source** *ip-address*] [**syntax-check**]	Enable the Cisco IOS agent, and initiate an initial configuration. • For {*hostname* | *ip-address*}, enter the hostname or the IP address of the configuration server. • (Optional) For *port-number*, enter the port number of the configuration server. The default port number is 80. • (Optional) Enable **event** for configuration success, failure, or warning messages when the configuration is finished. • (Optional) Enable **no-persist** to suppress the automatic writing to NVRAM of the configuration pulled as a result of entering the **cns config initial** global configuration command. If the **no-persist** keyword is not entered, using the **cns config initial** command causes the resultant configuration to be automatically written to NVRAM. • (Optional) For **page** *page*, enter the web page of the initial configuration. The default is /Config/config/asp. • (Optional) Enter **source** *ip-address* to use for source IP address. • (Optional) Enable **syntax-check** to check the syntax when this parameter is entered. **Note** Though visible in the command-line help string, the **encrypt**, **status** *url*, and **inventory** keywords are not supported.
Step 15	**end**	Return to privileged EXEC mode.
Step 16	**show cns config connections**	Verify information about the configuration agent.
Step 17	**show running-config**	Verify your entries.

To disable the CNS Cisco IOS agent, use the **no cns config initial** {*ip-address* | *hostname*} global configuration command.

This example shows how to configure an initial configuration on a remote switch when the switch configuration is unknown (the CNS Zero Touch feature).

```
Switch(config)# cns template connect template-dhcp
Switch(config-tmpl-conn)# cli ip address dhcp
Switch(config-tmpl-conn)# exit
Switch(config)# cns template connect ip-route
Switch(config-tmpl-conn)# cli ip route 0.0.0.0 0.0.0.0 ${next-hop}
Switch(config-tmpl-conn)# exit
Switch(config)# cns connect dhcp
Switch(config-cns-conn)# discover interface gigabitethernet
Switch(config-cns-conn)# template template-dhcp
Switch(config-cns-conn)# template ip-route
Switch(config-cns-conn)# exit
Switch(config)# hostname RemoteSwitch
RemoteSwitch(config)# cns config initial 10.1.1.1 no-persist
```

This example shows how to configure an initial configuration on a remote switch when the switch IP address is known. The Configuration Engine IP address is 172.28.129.22.

```
Switch(config)# cns template connect template-dhcp
Switch(config-tmpl-conn)# cli ip address dhcp
Switch(config-tmpl-conn)# exit
Switch(config)# cns template connect ip-route
Switch(config-tmpl-conn)# cli ip route 0.0.0.0 0.0.0.0 ${next-hop}
Switch(config-tmpl-conn)# exit
Switch(config)# cns connect dhcp
Switch(config-cns-conn)# discover interface gigabitethernet
Switch(config-cns-conn)# template template-dhcp
Switch(config-cns-conn)# template ip-route
Switch(config-cns-conn)# exit
Switch(config)# hostname RemoteSwitch
RemoteSwitch(config)# ip route 172.28.129.22 255.255.255.255 11.11.11.1
RemoteSwitch(config)# cns id ethernet 0 ipaddress
RemoteSwitch(config)# cns config initial 172.28.129.22 no-persist
```

Enabling a Partial Configuration

Beginning in privileged EXEC mode, follow these steps to enable the Cisco IOS agent and to initiate a partial configuration on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**cns config partial** {*ip-address* | *hostname*} [*port-number*] [**source** *ip-address*]	Enable the configuration agent, and initiate a partial configuration. • For {*ip-address* | *hostname*}, enter the IP address or the hostname of the configuration server. • (Optional) For *port-number*, enter the port number of the configuration server. The default port number is 80. • (Optional) Enter **source** *ip-address* to use for the source IP address. **Note** Though visible in the command-line help string, the **encrypt** keyword is not supported.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show cns config stats** or **show cns config outstanding**	Verify information about the configuration agent.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the Cisco IOS agent, use the **no cns config partial** {*ip-address* | *hostname*} global configuration command. To cancel a partial configuration, use the **cns config cancel** privileged EXEC command.

Upgrading Devices with Cisco IOS Image Agent

Administrators maintaining large networks of Cisco IOS devices need an automated mechanism to load image files onto large numbers of remote devices. Existing network management applications are useful to determine which images to run and how to manage images received from the Cisco online software center. Other image distribution solutions do not scale to cover thousands of devices and cannot distribute images to devices behind a firewall or using Network Address Translation (NAT). The CNS image agent enables the managed device to initiate a network connection and request an image download allowing devices using NAT, or behind firewalls, to access the image server.

You can use image agent to download one or more devices. The switches must have the image agent running on them.

Prerequisites for the CNS Image Agent

Confirm these prerequisites before upgrading one or more devices with image agent:

- Determine where to store the Cisco IOS images on a file server to make the image available to the other networking devices. If the CNS Event Bus is to be used to store and distribute the images, the CNS event agent must be configured.
- Set up a file server to enable the networking devices to download the new images using the HTTPS protocol.
- Determine how to handle error messages generated by image agent operations. Error messages can be sent to the CNS Event Bus or an HTTP or HTTPS URL.

Restrictions for the CNS Image Agent

During automated image loading operations you must try to prevent the Cisco IOS device from losing connectivity with the file server that is providing the image. Image reloading is subject to memory issues and connection issues. Boot options must also be configured to allow the Cisco IOS device to boot another image if the first image reload fails.

These other restrictions apply to the image agent running on a Catalyst 3550 switch:

- You can only download the tar image file. Downloading the bin image file is not supported.
- Only the immediate download option is supported. You cannot schedule a download to occur at a specified date and time.
- The Destination field in the Associate Image with Device window is not supported.

For more details, see your CNS IE2100 documentation and see the “File Management” section of the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2*.

Beginning in privileged EXEC mode, follow these steps to initiate the image agent to check for a new image and upgrade a device:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**ip host** {*ip-address*} {*hostname*}	Enter the IP address and the hostname of the event gateway.
Step 3	**cns trusted-server all-agents** {*hostname*}	Specify a trusted server for CNS agent.
Step 4	**no cns aaa enable cns event** {*ip-address*} {*port number*}	Disable AAA authentication on the event gateway.

	Command	Purpose
Step 5	**cns image retry** {*number*}	Specify the number of times to retry and download the image.
Step 6	**cns image server** {*ip-address*} **status** {*ip-address*}	Download the image from the server to the switch.
Step 7	**end**	Return to privileged EXEC mode.

Note This example shows how to upgrade a switch from a server with the address of **172.20.249.20:**

```
Switch(config)> configure terminal
Switch(config)# ip host cns-dsbu.cisco.com 172.20.249.20
Switch(config)# cns trusted-server all-agents cns-dsbu.cisco.com
Switch(config)# no cns aaa enable cns event 172.20.249.20 22022
Switch(config)# cns image retry 1
Switch(config)# cns image server http://172.20.249.20:80/cns/HttpMsgDispatcher status
http://172.20.249.20:80/cns/HttpMsgDispatcher
Switch(config)#end
```

You can check the status of the image download by using the **show cns image** status user EXEC command.

Displaying CNS Configuration

You can use the privileged EXEC commands in Table 4-2 to display CNS configuration information.

Table 4-2 Displaying CNS Configuration

Command	Purpose
show cns config connections	Displays the status of the CNS Cisco IOS agent connections.
show cns config outstanding	Displays information about incremental (partial) CNS configurations that have started but are not yet completed.
show cns config stats	Displays statistics about the Cisco IOS agent.
show cns event connections	Displays the status of the CNS event agent connections.
show cns event stats	Displays statistics about the CNS event agent.
show cns event subject	Displays a list of event agent subjects that are subscribed to by applications.

CHAPTER 5

Clustering Switches

This chapter provides the concepts and procedures to create and manage Catalyst 3550 switch clusters.

You can create and manage switch clusters by using Cisco Network Assistant (hereafter known as Network Assistant), the command-line interface (CLI), or SNMP. For complete procedures, see the online help. For the CLI cluster commands, see the switch command reference.

Note Network Assistant supports switch clusters, but we recommend that you instead group switches into *communities*. Network Assistant has a Cluster Conversion Wizard to help you convert the cluster to a *community*. For more information about Network Assistant, including introductory information on managing switch clusters and converting a switch cluster to a community, see *Getting Started with Cisco Network Assistant*, available on Cisco.com.

This chapter focuses on Catalyst 3550 switch clusters. It also includes guidelines and limitations for clusters mixed with other cluster-capable Catalyst switches, but it does not provide complete descriptions of the cluster features for these other switches. For complete cluster information for a specific Catalyst platform, refer to the software configuration guide for that switch.

This chapter consists of these sections:

- Understanding Switch Clusters, page 5-1
- Planning a Switch Cluster, page 5-4
- Using SNMP to Manage Switch Clusters, page 5-15

Note We do not recommend using the **ip http access-class** global configuration command to limit access to specific hosts or networks. Access should be controlled through the cluster command switch or by applying access control lists (ACLs) on interfaces that are configured with IP address. For more information on ACLs, see Chapter 29, “Configuring Network Security with ACLs.”.

Understanding Switch Clusters

A *switch cluster* is a set of up to 16 connected, cluster-capable Catalyst switches that are managed as a single entity. The switches in the cluster use the switch clustering technology so that you can configure and troubleshoot a group of different Catalyst desktop switch platforms through a single IP address.

In a switch cluster, 1 switch must be the *cluster command switch* and up to 15 other switches can be *cluster member switches*. The total number of switches in a cluster cannot exceed 16 switches. The cluster command switch is the single point of access used to configure, manage, and monitor the cluster member switches. Cluster members can belong to only one cluster at a time.

The benefits of clustering switches include:

- Management of Catalyst switches regardless of their interconnection media and their physical locations. The switches can be in the same location, or they can be distributed across a network.

  Cluster members are connected to the cluster command switch according to the connectivity guidelines described in the "Automatic Discovery of Cluster Candidates and Members" section on page 5-4. This section includes management VLAN considerations for the Catalyst 1900, Catalyst 2820, Catalyst 2900 XL, Catalyst 2950, and Catalyst 3500 XL switches. For complete information about these switches in a switch-cluster environment, refer to the software configuration guide for that specific switch.
- Command-switch redundancy if a cluster command switch fails. One or more switches can be designated as *standby cluster command switches* to avoid loss of contact with cluster members. A *cluster standby group* is a group of standby cluster command switches.
- Management of a variety of Catalyst switches through a single IP address. This conserves on IP addresses, especially if you have a limited number of them. All communication with the switch cluster is through the cluster command switch IP address.

Table 5-1 lists the Catalyst switches eligible for switch clustering, including which ones can be cluster command switches and which ones can only be cluster member switches, and the required software versions.

Table 5-1 Switch Software and Cluster Capability

Switch	Cisco IOS Release	Cluster Capability
Catalyst 3750	12.1(11)AX or later	Member or command switch
Catalyst 3560	12.1(19)EA1b or later	Member or command switch
Catalyst 3550	12.1(4)EA1 or later	Member or command switch
Catalyst 2970	12.1(11)AX or later	Member or command switch
Catalyst 2960	12.2(25)FX or later	Member or command switch
Catalyst 2955	12.1(12c)EA1 or later	Member or command switch
Catalyst 2950	12.0(5.2)WC(1) or later	Member or command switch
Catalyst 2950 LRE	12.1(11)JY or later	Member or command switch
Catalyst 2940	12.1(13)AY or later	Member or command switch
Catalyst 3500 XL	12.0(5.1)XU or later	Member or command switch
Catalyst 2900 XL (8-MB switches)	12.0(5.1)XU or later	Member or command switch
Catalyst 2900 XL (4-MB switches)	11.2(8.5)SA6 (recommended)	Member switch only
Catalyst 1900 and 2820	9.00(-A or -EN) or later	Member switch only

Cluster Command Switch Characteristics

A Catalyst 3550 cluster command switch must meet these requirements:

- It is running Cisco IOS Release 12.1(4)EA1 or later.
- It has an IP address.
- It has Cisco Discovery Protocol (CDP) version 2 enabled (the default).
- It is not a command or cluster member switch of another cluster.
- It is connected to the standby cluster command switches through the management VLAN and to the cluster member switches through a common VLAN.

We strongly recommend that the highest-end, command-capable switch in the cluster be the command switch:

- If your switch cluster has a Catalyst 3550 switch, that switch should be the command switch.
- If your switch cluster has Catalyst 2900 XL, Catalyst 2940, Catalyst 2950, Catalyst 2955, and Catalyst 3500 XL switches, the Catalyst 2950 or the Catalyst 2955 should be the command switch

Standby Cluster Command Switch Characteristics

A standby cluster command switch must meet these requirements:

- It is running Cisco IOS 12.1(4)EA1 or later.
- It has an IP address.
- It has CDP version 2 enabled.
- It is connected to other standby switches through its management VLAN and to all member switches through a common VLAN.
- It is redundantly connected to the cluster so that connectivity to member switches is maintained.
- It is not a command or member switch of another cluster.

Note Standby cluster command switches must be the same type of switches as the cluster command switch. For example, if the cluster command switch is a Catalyst 3550 switch, the standby cluster command switches must also be Catalyst 3550 switches.

Candidate Switch and Member Switch Characteristics

Candidate switches are cluster-capable switches that have not yet been added to a cluster. Cluster member switches are switches that have actually been added to a switch cluster. Although not required, a candidate or cluster member switch can have its own IP address and password.

To join a cluster, a candidate switch must meet these requirements:

- It is running cluster-capable software.
- It has CDP Version 2 enabled.
- It is not a command or member switch of another cluster.

- It is connected to the command switch through at least one common VLAN.
- If a cluster standby group exists, it is connected to every standby command switch through at least one common VLAN. The VLAN to each standby command switch can be different.

Note These candidate and member switches must be connected through their management VLAN to the command switch and standby command switches: Catalyst 1900 switches, Catalyst 2820 switches, Catalyst 2900 XL switches, non-LRE Catalyst 2950 switches running a release earlier than Cisco IOS Release 12.1(9)EA1, and Catalyst 3500 XL switches.

This requirement does not apply if you have a non-LRE Catalyst 2950 command switch running Cisco IOS Release 12.1(9)EA1 or later, a Catalyst 2950 LRE command switch, Catalyst 2940 command switch, a Catalyst 2955 command switch, or a Catalyst 3550 command switch. Candidate and member switches can connect through any VLAN in common with the command switch.

Planning a Switch Cluster

Anticipating conflicts and compatibility issues is a high priority when you manage several switches through a cluster. This section describes these guidelines, requirements, and caveats that you should understand before you create the cluster:

- Automatic Discovery of Cluster Candidates and Members, page 5-4
- HSRP and Standby Cluster Command Switches, page 5-10
- IP Addresses, page 5-13
- Hostnames, page 5-13
- Passwords, page 5-13
- SNMP Community Strings, page 5-14
- TACACS+ and RADIUS, page 5-14

Automatic Discovery of Cluster Candidates and Members

The cluster command switch uses Cisco Discovery Protocol (CDP) to discover cluster member switches, candidate switches, neighboring switch clusters, and edge devices across multiple VLANs and in star or cascaded topologies.

Note Do not disable CDP on the cluster command switch, on cluster members, or on any cluster-capable switches that you might want a cluster command switch to discover. For more information about CDP, see Chapter 22, “Configuring CDP.”

Following these connectivity guidelines ensures automatic discovery of the switch cluster, cluster candidates, connected switch clusters, and neighboring edge devices:

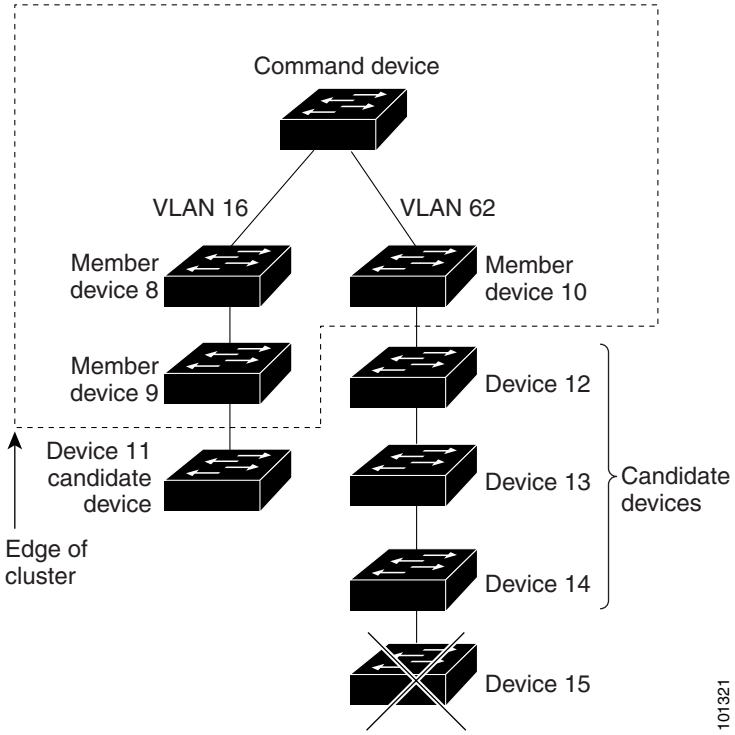
- Discovery Through CDP Hops, page 5-5
- Discovery Through Non-CDP-Capable and Noncluster-Capable Devices, page 5-6
- Discovery Through Different VLANs, page 5-6
- Discovery Through Different Management VLANs, page 5-7
- Discovery Through Routed Ports, page 5-8
- Discovery of Newly Installed Switches, page 5-9

Discovery Through CDP Hops

By using CDP, a cluster command switch can discover switches up to seven CDP hops away (the default is three hops) from the edge of the cluster. The edge of the cluster is where the last cluster member switches are connected to the cluster and to candidate switches. For example, cluster member switches 9 and 10 in Figure 5-1 are at the edge of the cluster.

In Figure 5-1, the cluster command switch has ports assigned to VLANs 16 and 62. The CDP hop count is three. The cluster command switch discovers switches 11, 12, 13, and 14 because they are within three hops from the edge of the cluster. It does not discover switch 15 because it is four hops from the edge of the cluster.

Figure 5-1 Discovery Through CDP Hops

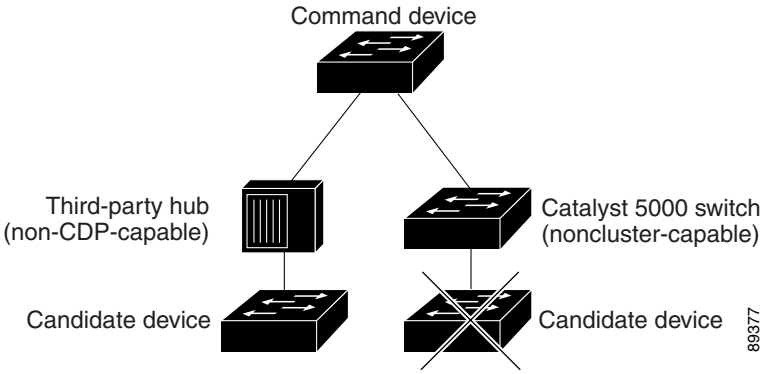


Discovery Through Non-CDP-Capable and Noncluster-Capable Devices

If a cluster command switch is connected to a *non-CDP-capable third-party hub* (such as a non-Cisco hub), it can discover cluster-enabled devices connected to that third-party hub. However, if the cluster command switch is connected to a *noncluster-capable Cisco device*, it cannot discover a cluster-enabled device connected beyond the noncluster-capable Cisco device.

Figure 5-2 shows that the cluster command switch discovers the switch that is connected to a third-party hub. However, the cluster command switch does not discover the switch that is connected to a Catalyst 5000 switch.

Figure 5-2 Discovery Through Non-CDP-Capable and Noncluster-Capable Devices

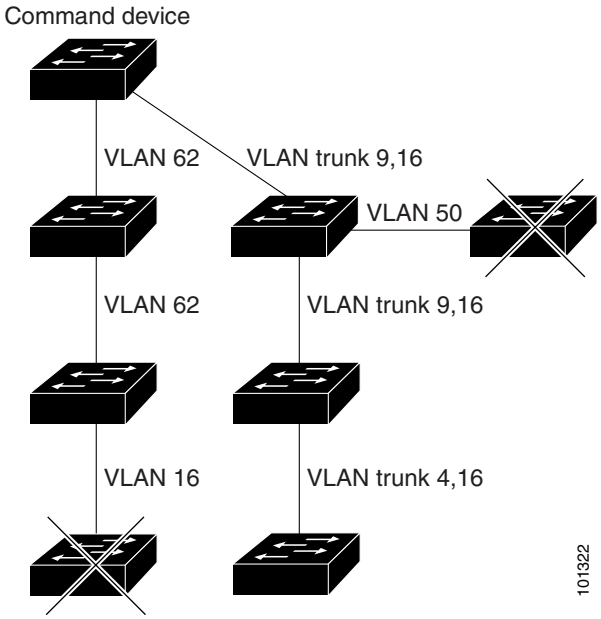


Discovery Through Different VLANs

If the cluster command switch is a Catalyst 2940, Catalyst 2950, Catalyst 2955, or Catalyst 3550 switch, the cluster can have cluster member switches in different VLANs. As cluster member switches, they must be connected through at least one VLAN in common with the cluster command switch. The cluster command switch in Figure 5-3 has ports assigned to VLANs 9, 16, and 62 and therefore discovers the switches in those VLANs. It does not discover the switch in VLAN 50. It also does not discover the switch in VLAN 16 in the first column because the cluster command switch has no VLAN connectivity to it.

Catalyst 2900 XL, Catalyst 2950, and Catalyst 3500 XL cluster member switches must be connected to the cluster command switch through their management VLAN. For information about discovery through management VLANs, see the "Discovery Through Different Management VLANs" section on page 5-7. For more information about VLANs, see Chapter 11, "Configuring VLANs."

Figure 5-3 Discovery Through Different VLANs



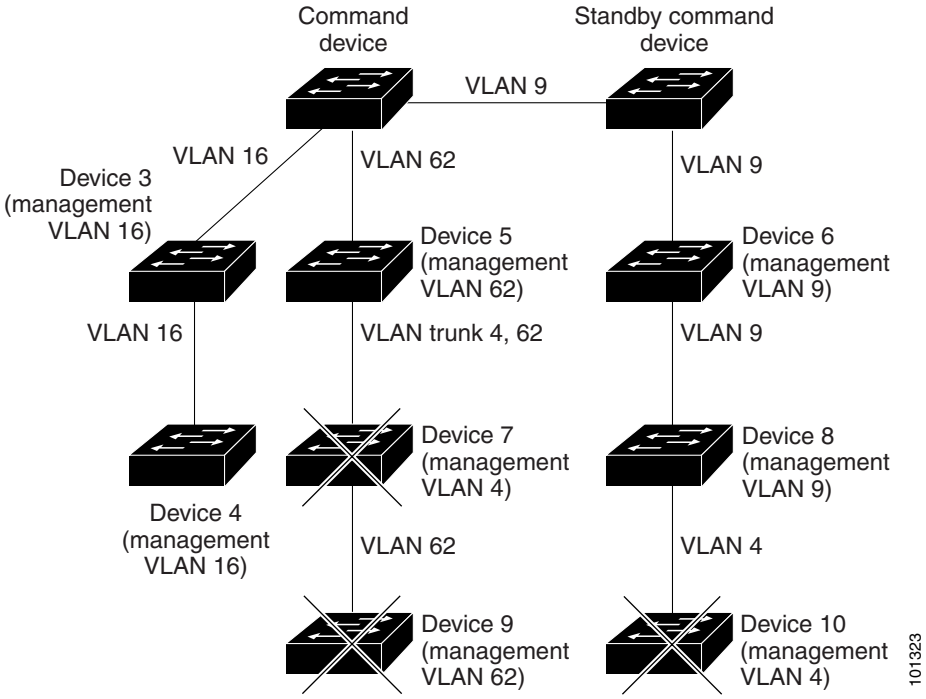
Discovery Through Different Management VLANs

Catalyst 2940, Catalyst 2950, Catalyst 2955, or Catalyst 3550 cluster command switches can discover and manage cluster member switches in different VLANs and different management VLANs. As cluster member switches, they must be connected through at least one VLAN in common with the cluster command switch. They do not need to be connected to the cluster command switch through their management VLAN. The default management VLAN is VLAN 1.

The cluster command switch and standby command switch in Figure 5-4 (assuming they are Catalyst 2940, Catalyst 2950, Catalyst 2955, or Catalyst 3550 cluster command switches) have ports assigned to VLANs 9, 16, and 62. The management VLAN on the cluster command switch is VLAN 9. Each cluster command switch discovers the switches in the different management VLANs except these:

- Switches 7 and 10 (switches in management VLAN 4) because they are not connected through a common VLAN (meaning VLANs 62 and 9) with the cluster command switch
- Switch 9 because automatic discovery does not extend beyond a noncandidate device, which is switch 7

Figure 5-4 Discovery Through Different Management VLANs with a Layer 3 Cluster Command Switch

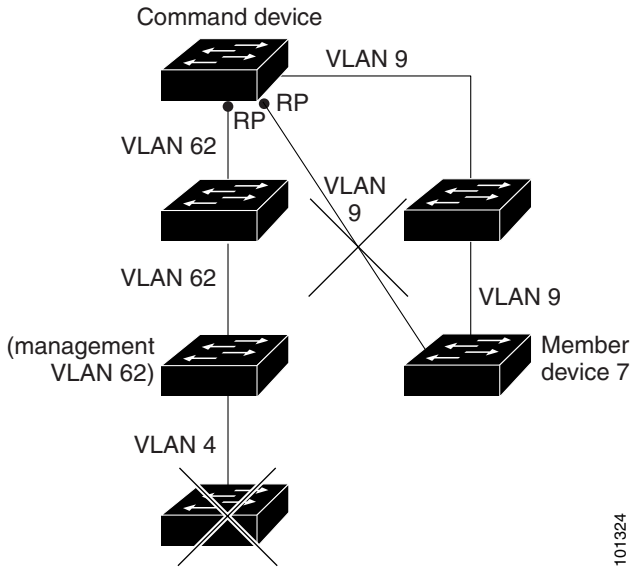


Discovery Through Routed Ports

If the cluster command switch has a routed port (RP) configured, it discovers only candidate and cluster member switches in the *same* VLAN as the routed port. For more information about routed ports, see the "Routed Ports" section on page 9-4.

The Layer 3 cluster command switch in Figure 5-5 can discover the switches in VLANs 9 and 62 but not the switch in VLAN 4. If the routed port path between the cluster command switch and cluster member switch 7 is lost, connectivity with cluster member switch 7 is maintained because of the redundant path through VLAN 9.

Figure 5-5 Discovery Through Routed Ports



Discovery of Newly Installed Switches

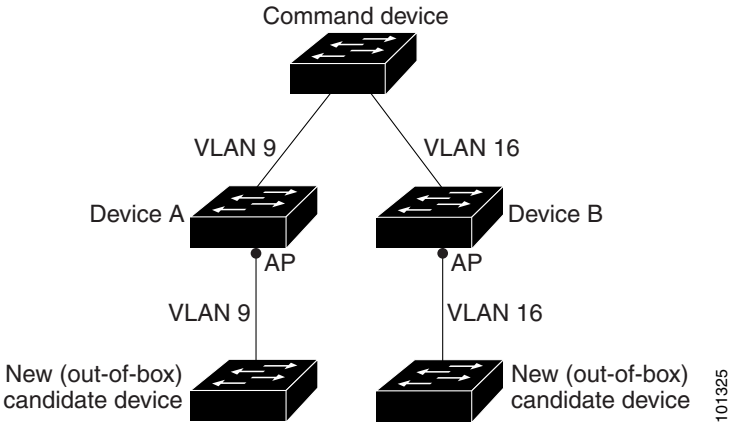
To join a cluster, the new, out-of-the-box switch must be connected to the cluster through one of its access ports. An access port (AP) carries the traffic of and belongs to only one VLAN. By default, the new switch and its access ports are assigned to VLAN 1.

When the new switch joins a cluster, its default VLAN changes to the VLAN of the immediately upstream neighbor. The new switch also configures its access port to belong to the VLAN of the immediately upstream neighbor.

The cluster command switch in Figure 5-6 belongs to VLANs 9 and 16. When new cluster-capable switches join the cluster:

- One cluster-capable switch and its access port are assigned to VLAN 9.
- The other cluster-capable switch and its access port are assigned to management VLAN 16.

Figure 5-6 Discovery of Newly Installed Switches



HSRP and Standby Cluster Command Switches

The switch supports Hot Standby Router Protocol (HSRP) so that you can configure a group of standby cluster command switches. Because a cluster command switch manages the forwarding of all communication and configuration information to all the cluster member switches, we strongly recommend the following:

- For a cluster command switch stack, a standby cluster command switch is necessary if the entire switch stack fails. However, if only the stack master in the command switch stack fails, the switch stack elects a new stack master and resumes its role as the cluster command switch stack.
- For a cluster command switch that is a standalone switch, configure a standby cluster command switch to take over if the primary cluster command switch fails.

A *cluster standby group* is a group of command-capable switches that meet the requirements described in the "Standby Cluster Command Switch Characteristics" section on page 5-3. Only one cluster standby group can be assigned per cluster.

Note The cluster standby group is an HSRP group. Disabling HSRP disables the cluster standby group.

The switches in the cluster standby group are ranked according to HSRP priorities. The switch with the highest priority in the group is the *active cluster command switch* (AC). The switch with the next highest priority is the *standby cluster command switch* (SC). The other switches in the cluster standby group are the *passive cluster command switches* (PC). If the active cluster command switch and the standby cluster command switch become disabled *at the same time*, the passive cluster command switch with the highest priority becomes the active cluster command switch. For the limitations to automatic discovery, see the "Automatic Recovery of Cluster Configuration" section on page 5-12. For information about changing HSRP priority values, see the "Configuring HSRP Priority" section on page 32-6. The HSRP **standby priority** interface configuration commands are the same for changing the priority of cluster standby group members and router-redundancy group members.

Note The HSRP standby hold time interval should be greater than or equal to three times the hello time interval. The default HSRP standby hold time interval is 10 seconds. The default HSRP standby hello time interval is 3 seconds. For more information about the standby hold time and standby hello time intervals, see the "Configuring HSRP Authentication and Timers" section on page 32-8.

These connectivity guidelines ensure automatic discovery of the switch cluster, cluster candidates, connected switch clusters, and neighboring edge devices. These topics also provide more detail about standby cluster command switches:

- Virtual IP Addresses, page 5-10
- Other Considerations for Cluster Standby Groups, page 5-11
- Automatic Recovery of Cluster Configuration, page 5-12

Virtual IP Addresses

You need to assign a unique virtual IP address and group number and name to the cluster standby group. This information must be configured on a specific VLAN or routed port on the active cluster command switch. The active cluster command switch receives traffic destined for the virtual IP address. To

manage the cluster, you must access the active cluster command switch through the virtual IP address, not through the command-switch IP address. This is in case the IP address of the active cluster command switch is different from the virtual IP address of the cluster standby group.

If the active cluster command switch fails, the standby cluster command switch assumes ownership of the virtual IP address and becomes the active cluster command switch. The passive switches in the cluster standby group compare their assigned priorities to decide the new standby cluster command switch. The passive standby switch with the highest priority then becomes the standby cluster command switch. When the previously active cluster command switch becomes active again, it resumes its role as the active cluster command switch, and the current active cluster command switch becomes the standby cluster command switch again. For more information about IP address in switch clusters, see the "IP Addresses" section on page 5-13.

Other Considerations for Cluster Standby Groups

These requirements also apply:

- Standby cluster command switches must be the same type of switches as the cluster command switch. For example, if the cluster command switch is a Catalyst 3550 switch, the standby cluster command switches must also be Catalyst 3550 switches. Refer to the switch configuration guide of other cluster-capable switches for their requirements on standby cluster command switches.
- Only one cluster standby group can be assigned to a cluster. You can have more than one router-redundancy standby group.

  An HSRP group can be both a cluster standby group and a router-redundancy group. However, if a router-redundancy group becomes a cluster standby group, router redundancy becomes disabled on that group. You can re-enable it by using the CLI. For more information about HSRP and router redundancy, see Chapter 32, "Configuring HSRP."
- All standby-group members must be members of the cluster.

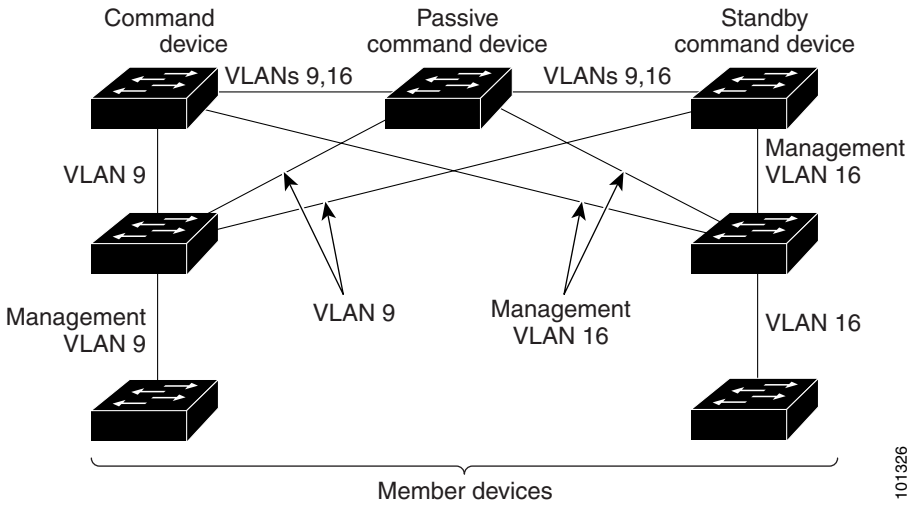
  Note There is no limit to the number of switches that you can assign as standby cluster command switches. However, the total number of switches in the cluster—which would include the active cluster command switch, standby-group members, and cluster member switches—cannot be more than 16.

- Each standby-group member (Figure 5-7) must be connected to the cluster command switch through the same VLAN. Each standby-group member must also be redundantly connected to each other through at least one VLAN in common with the switch cluster.

  Catalyst 1900, Catalyst 2820, Catalyst 2900 XL, Catalyst 2950, and Catalyst 3500 XL cluster member switches must be connected to the cluster standby group through their management VLANs. For more information about VLANs in switch clusters, see these sections:

  – "Discovery Through Different VLANs" section on page 5-6
  – "Discovery Through Different Management VLANs" section on page 5-7

Figure 5-7 VLAN Connectivity between Standby-Group Members and Cluster Members



Automatic Recovery of Cluster Configuration

The active cluster command switch continually forwards cluster-configuration information (but not device-configuration information) to the standby cluster command switch. This ensures that the standby cluster command switch can take over the cluster immediately after the active cluster command switch fails.

Automatic discovery has these limitations:

- This limitation applies only to clusters that have Catalyst 2950, Catalyst 3550, Catalyst 3560, and Catalyst 3750 command and standby cluster command switches: If the active cluster command switch and standby cluster command switch become disabled *at the same time*, the passive cluster command switch with the highest priority becomes the active cluster command switch. However, because it was a passive standby cluster command switch, the previous cluster command switch *did not* forward cluster-configuration information to it. The active cluster command switch only forwards cluster-configuration information to the standby cluster command switch. You must therefore rebuild the cluster.
- This limitation applies to all clusters: If the active cluster command switch fails and there are more than two switches in the cluster standby group, the new cluster command switch does not discover any Catalyst 1900, Catalyst 2820, and Catalyst 2916M XL cluster member switches. You must re-add these cluster member switches to the cluster.
- This limitation applies to all clusters: If the active cluster command switch fails and becomes active again, it does not discover any Catalyst 1900, Catalyst 2820, and Catalyst 2916M XL cluster member switches. You must again add these cluster member switches to the cluster.

When the previously active cluster command switch resumes its active role, it receives a copy of the latest cluster configuration from the active cluster command switch, including members that were added while it was down. The active cluster command switch sends a copy of the cluster configuration to the cluster standby group.

IP Addresses

You must assign IP information to a cluster command switch. You can assign more than one IP address to the cluster command switch, and you can access the cluster through any of the command-switch IP addresses. If you configure a cluster standby group, you must use the standby-group virtual IP address to manage the cluster from the active cluster command switch. Using the virtual IP address ensures that you retain connectivity to the cluster if the active cluster command switch fails and that a standby cluster command switch becomes the active cluster command switch.

If the active cluster command switch fails and the standby cluster command switch takes over, you must either use the standby-group virtual IP address or any of the IP addresses available on the new active cluster command switch to access the cluster.

You can assign an IP address to a cluster-capable switch, but it is not necessary. A cluster member switch is managed and communicates with other cluster member switches through the command-switch IP address. If the cluster member switch leaves the cluster and it does not have its own IP address, you must assign an IP address to manage it as a standalone switch.

For more information about IP addresses, see Chapter 3, "Assigning the Switch IP Address and Default Gateway."

Hostnames

You do not need to assign a host name to either a cluster command switch or an eligible cluster member. However, a hostname assigned to the cluster command switch can help to identify the switch cluster. The default hostname for the switch is *Switch*.

If a switch joins a cluster and it does not have a hostname, the cluster command switch appends a unique member number to its own hostname and assigns it sequentially as each switch joins the cluster. The number means the order in which the switch was added to the cluster. For example, a cluster command switch named *eng-cluster* could name the fifth cluster member *eng-cluster-5*.

If a switch has a hostname, it retains that name when it joins a cluster and when it leaves the cluster.

If a switch received its hostname from the cluster command switch, was removed from a cluster, was then added to a new cluster, and kept the same member number (such as *5*), the switch overwrites the old hostname (such as *eng-cluster-5*) with the hostname of the cluster command switch in the new cluster (such as *mkg-cluster-5*). If the switch member number changes in the new cluster (such as *3*), the switch retains the previous name (*eng-cluster-5*).

Passwords

You do not need to assign passwords to an individual switch if it will be a cluster member. When a switch joins a cluster, it inherits the command-switch password and retains it when it leaves the cluster. If no command-switch password is configured, the cluster member switch inherits a null password. Cluster member switches only inherit the command-switch password.

If you change the member-switch password to be different from the command-switch password and save the change, the switch is not manageable by the cluster command switch until you change the member-switch password to match the command-switch password. Rebooting the member switch does not revert the password back to the command-switch password. We recommend that you do not change the member-switch password after it joins a cluster.

For more information about passwords, see the "Preventing Unauthorized Access to Your Switch" section on page 7-1.

For password considerations specific to the Catalyst 1900 and Catalyst 2820 switches, refer to the installation and configuration guides for those switches.

SNMP Community Strings

A cluster member switch inherits the command-switch first read-only (RO) and read-write (RW) community strings with *@esN* appended to the community strings:

- *command-switch-readonly-community-string@esN*, where *N* is the member-switch number.
- *command-switch-readwrite-community-string@esN*, where *N* is the member-switch number.

If the cluster command switch has multiple read-only or read-write community strings, only the first read-only and read-write strings are propagated to the cluster member switch.

The switches support an unlimited number of community strings and string lengths. For more information about SNMP and community strings, see Chapter 27, "Configuring SNMP."

For SNMP considerations specific to the Catalyst 1900 and Catalyst 2820 switches, refer to the installation and configuration guides specific to those switches.

TACACS+ and RADIUS

If Terminal Access Controller Access Control System Plus (TACACS+) is configured on a cluster member, it must be configured on all cluster members. Similarly, if RADIUS is configured on a cluster member, it must be configured on all cluster members. Further, the same switch cluster cannot have some members configured with TACACS+ and other members configured with RADIUS.

For more information about TACACS+, see the "Controlling Switch Access with TACACS+" section on page 7-9. For more information about RADIUS, see the "Controlling Switch Access with RADIUS" section on page 7-17.

Catalyst 1900 and Catalyst 2820 CLI Considerations

If your switch cluster has Catalyst 1900 and Catalyst 2820 switches running standard edition software, the Telnet session accesses the management console (a menu-driven interface) if the command switch is at privilege level 15. If the command switch is at privilege level 1 to 14, you are prompted for the password to access the menu console.

Note Catalyst 1900, 2900 XL (4 MB), and 2820 switches are not supported in Network Assistant. The switches appear as *unknown members* in the Network Assistant Front Panel and Topology views.

Command-switch privilege levels map to the Catalyst 1900 and Catalyst 2820 member switches running standard and Enterprise Edition Software as follows:

- If the command-switch privilege level is 1 to 14, the member switch is accessed at privilege level 1.
- If the command-switch privilege level is 15, the member switch is accessed at privilege level 15.

Note The Catalyst 1900 and Catalyst 2820 CLI is available only on switches running Enterprise Edition Software.

For more information about the Catalyst 1900 and Catalyst 2820 switches, see the installation and configuration guides for those switches.

Using SNMP to Manage Switch Clusters

When you first power on the switch, SNMP is enabled if you enter the IP information by using the setup program and accept its proposed configuration. If you did not use the setup program to enter the IP information and SNMP was not enabled, you can enable it as described in the "Configuring SNMP" section on page 27-6. On Catalyst 1900 and Catalyst 2820 switches, SNMP is enabled by default.

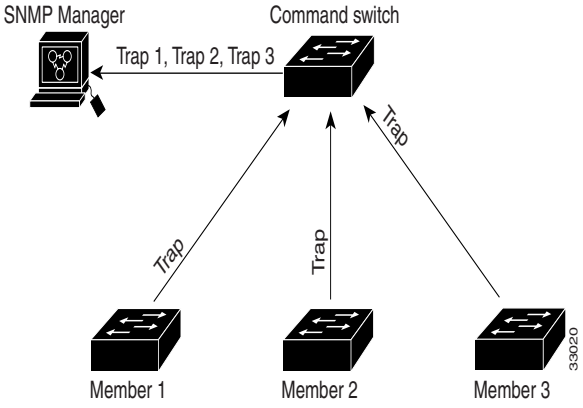
When you create a cluster, the command switch manages the exchange of messages between member switches and an SNMP application. The cluster software on the command switch appends the member switch number (*@esN*, where *N* is the switch number) to the first configured read-write and read-only community strings on the command switch and propagates them to the member switch. The command switch uses this community string to control the forwarding of gets, sets, and get-next messages between the SNMP management station and the member switches.

Note When a cluster standby group is configured, the command switch can change without your knowledge. Use the first read-write and read-only community strings to communicate with the command switch if there is a cluster standby group configured for the cluster.

If the member switch does not have an IP address, the command switch redirects traps from the member switch to the management station, as shown in Figure 5-8. If a member switch has its own IP address and community strings, the member switch can send traps directly to the management station, without going through the command switch.

If a member switch has its own IP address and community strings, they can be used in addition to the access provided by the command switch. For more information about SNMP and community strings, see Chapter 27, "Configuring SNMP."

Figure 5-8 SNMP Management for a Cluster



CHAPTER 6

Administering the Switch

This chapter describes how to perform one-time operations to administer your Catalyst 3550 switch. This chapter consists of these sections:

- Managing the System Time and Date, page 6-1
- Configuring a System Name and Prompt, page 6-14
- Creating a Banner, page 6-17
- Managing the MAC Address Table, page 6-19
- Optimizing System Resources for User-Selected Features, page 6-26
- Managing the ARP Table, page 6-29

Managing the System Time and Date

You can manage the system time and date on your switch using automatic configuration, such as the Network Time Protocol (NTP), or manual configuration methods.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference for Cisco IOS, Release 12.2*.

This section contains this configuration information:

- Understanding the System Clock, page 6-1
- Understanding Network Time Protocol, page 6-2
- Configuring NTP, page 6-3
- Configuring Time and Date Manually, page 6-11

Understanding the System Clock

The heart of the time service is the system clock. This clock runs from the moment the system starts up and keeps track of the date and time.

The system clock can then be set from these sources:

- Network Time Protocol
- Manual configuration

The system clock can provide time to these services:

- User **show** commands
- Logging and debugging messages

The system clock keeps track of time internally based on Universal Time Coordinated (UTC), also known as Greenwich Mean Time (GMT). You can configure information about the local time zone and summer time (daylight saving time) so that the time appears correctly for the local time zone.

The system clock keeps track of whether the time is *authoritative* or not (that is, whether it has been set by a time source considered to be authoritative). If it is not authoritative, the time is available only for display purposes and is not redistributed. For configuration information, see the "Configuring Time and Date Manually" section on page 6-11.

Understanding Network Time Protocol

The NTP is designed to time-synchronize a network of devices. NTP runs over User Datagram Protocol (UDP), which runs over IP. NTP is documented in RFC 1305.

An NTP network usually gets its time from an authoritative time source, such as a radio clock or an atomic clock attached to a time server. NTP then distributes this time across the network. NTP is extremely efficient; no more than one packet per minute is necessary to synchronize two devices to within a millisecond of one another.

NTP uses the concept of a *stratum* to describe how many NTP hops away a device is from an authoritative time source. A stratum 1 time server has a radio or atomic clock directly attached, a stratum 2 time server receives its time through NTP from a stratum 1 time server, and so on. A device running NTP automatically chooses as its time source the device with the lowest stratum number with which it communicates through NTP. This strategy effectively builds a self-organizing tree of NTP speakers.

NTP avoids synchronizing to a device whose time might not be accurate by never synchronizing to a device that is not synchronized. NTP also compares the time reported by several devices and does not synchronize to a device whose time is significantly different than the others, even if its stratum is lower.

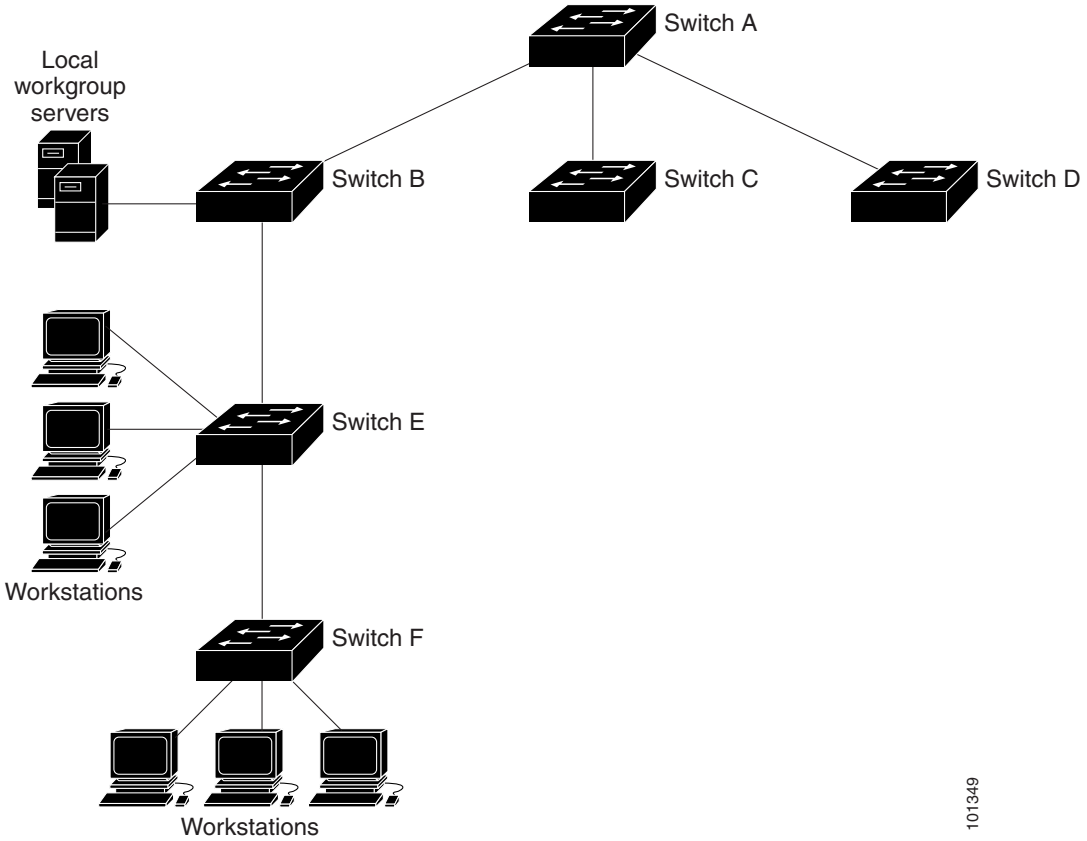
The communications between devices running NTP (known as *associations*) are usually statically configured; each device is given the IP address of all devices with which it should form associations. Accurate timekeeping is possible by exchanging NTP messages between each pair of devices with an association. However, in a LAN environment, NTP can be configured to use IP broadcast messages instead. This alternative reduces configuration complexity because each device can simply be configured to send or receive broadcast messages. However, in that case, information flow is one-way only.

The time kept on a device is a critical resource; you should use the security features of NTP to avoid the accidental or malicious setting of an incorrect time. Two mechanisms are available: an access list-based restriction scheme and an encrypted authentication mechanism.

Cisco's implementation of NTP does not support stratum 1 service; it is not possible to connect to a radio or atomic clock. We recommend that the time service for your network be derived from the public NTP servers available on the IP Internet.

Figure 6-1 show a typical network example using NTP. Switch A is the NTP master, with Switches B, C, and D configured in NTP server mode, in server association with Switch A. Switch E is configured as an NTP peer to the upstream and downstream switches, Switch B and Switch F.

Figure 6-1 Typical NTP Network Configuration



If the network is isolated from the Internet, Cisco's implementation of NTP allows a device to act as though it is synchronized through NTP, when in fact it has determined the time by using other means. Other devices then synchronize to that device through NTP.

When multiple sources of time are available, NTP is always considered to be more authoritative. NTP time overrides the time set by any other method.

Several manufacturers include NTP software for their host systems, and a publicly available version for systems running UNIX and its various derivatives is also available. This software allows host systems to be time-synchronized as well.

Configuring NTP

The switch does not have a hardware-supported clock, and it cannot function as an NTP master clock to which peers synchronize themselves when an external NTP source is not available. The switch also has no hardware support for a calendar. As a result, the **ntp update-calendar** and the **ntp master** global configuration commands are not available.

This section contains this configuration information:

- Default NTP Configuration, page 6-4
- Configuring NTP Authentication, page 6-4
- Configuring NTP Associations, page 6-5

- Configuring NTP Broadcast Service, page 6-6
- Configuring NTP Access Restrictions, page 6-8
- Configuring the Source IP Address for NTP Packets, page 6-10
- Displaying the NTP Configuration, page 6-11

Default NTP Configuration

Table 6-1 shows the default NTP configuration.

Table 6-1 Default NTP Configuration

Feature	Default Setting
NTP authentication	Disabled. No authentication key is specified.
NTP peer or server associations	None configured.
NTP broadcast service	Disabled; no interface sends or receives NTP broadcast packets.
NTP access restrictions	No access control is specified.
NTP packet source IP address	The source address is determined by the outgoing interface.

NTP is enabled on all interfaces by default. All interfaces receive NTP packets.

Configuring NTP Authentication

This procedure must be coordinated with the administrator of the NTP server; the information you configure in this procedure must be matched by the servers used by the switch to synchronize its time to the NTP server.

Beginning in privileged EXEC mode, follow these steps to authenticate the associations (communications between devices running NTP that provide for accurate timekeeping) with other devices for security purposes:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ntp authenticate**	Enable the NTP authentication feature, which is disabled by default.
Step 3	**ntp authentication-key** *number* **md5** *value*	Define the authentication keys. By default, none are defined. • For *number*, specify a key number. The range is 1 to 4294967295. • **md5** specifies that message authentication support is provided by using the message digest algorithm 5 (MD5). • For *value*, enter an arbitrary string of up to eight characters for the key. The switch does not synchronize to a device unless both have one of these authentication keys, and the key number is specified by the **ntp trusted-key** *key-number* command.

	Command	Purpose
Step 4	**ntp trusted-key** *key-number*	Specify one or more key numbers (defined in Step 3) that a peer NTP device must provide in its NTP packets for this switch to synchronize to it. By default, no trusted keys are defined. For *key-number*, specify the key defined in Step 3. This command provides protection against accidentally synchronizing the switch to a device that is not trusted.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable NTP authentication, use the **no ntp authenticate** global configuration command. To remove an authentication key, use the **no ntp authentication-key** *number* global configuration command. To disable authentication of the identity of a device, use the **no ntp trusted-key** *key-number* global configuration command.

This example shows how to configure the switch to synchronize only to devices providing authentication key 42 in the device's NTP packets:

```
Switch(config)# ntp authenticate
Switch(config)# ntp authentication-key 42 md5 aNiceKey
Switch(config)# ntp trusted-key 42
```

Configuring NTP Associations

An NTP association can be a peer association (this switch can either synchronize to the other device or allow the other device to synchronize to it), or it can be a server association (meaning that only this switch synchronizes to the other device, and not the other way around).

Beginning in privileged EXEC mode, follow these steps to form an NTP association with another device:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ntp peer** *ip-address* [**version** *number*] [**key** *keyid*] [**source** *interface*] [**prefer**] or **ntp server** *ip-address* [**version** *number*] [**key** *keyid*] [**source** *interface*] [**prefer**]	Configure the switch system clock to synchronize a peer or to be synchronized by a peer (peer association). or Configure the switch system clock to be synchronized by a time server (server association). No peer or server associations are defined by default. • For *ip-address* in a peer association, specify either the IP address of the peer providing, or being provided, the clock synchronization. For a server association, specify the IP address of the time server providing the clock synchronization. • (Optional) For *number*, specify the NTP version number. The range is 1 to 3. By default, version 3 is selected. • (Optional) For *keyid*, enter the authentication key defined with the **ntp authentication-key** global configuration command. • (Optional) For *interface*, specify the interface from which to pick the IP source address. By default, the source IP address is taken from the outgoing interface. • (Optional) Enter the **prefer** keyword to make this peer or server the preferred one that provides synchronization. This keyword reduces switching back and forth between peers and servers.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

You need to configure only one end of an association; the other device can automatically establish the association. If you are using the default NTP version (version 3) and NTP synchronization does not occur, try using NTP version 2. Many NTP servers on the Internet run version 2.

To remove a peer or server association, use the **no ntp peer** *ip-address* or the **no ntp server** *ip-address* global configuration command.

This example shows how to configure the switch to synchronize its system clock with the clock of the peer at IP address 172.16.22.44 using NTP version 2:

```
Switch(config)# ntp server 172.16.22.44 version 2
```

Configuring NTP Broadcast Service

The communications between devices running NTP (known as *associations*) are usually statically configured; each device is given the IP addresses of all devices with which it should form associations. Accurate timekeeping is possible by exchanging NTP messages between each pair of devices with an association. However, in a LAN environment, NTP can be configured to use IP broadcast messages instead. This alternative reduces configuration complexity because each device can simply be configured to send or receive broadcast messages. However, the information flow is one-way only.

The switch can send or receive NTP broadcast packets on an interface-by-interface basis if there is an NTP broadcast server, such as a router, broadcasting time information on the network. The switch can send NTP broadcast packets to a peer so that the peer can synchronize to it. The switch can also receive NTP broadcast packets to synchronize its own clock. This section has procedures for both sending and receiving NTP broadcast packets.

Beginning in privileged EXEC mode, follow these steps to configure the switch to send NTP broadcast packets to peers so that they can synchronize their clock to the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to send NTP broadcast packets, and enter interface configuration mode.
Step 3	**ntp broadcast** [**version** *number*] [**key** *keyid*] [*destination-address*]	Enable the interface to send NTP broadcast packets to a peer. By default, this feature is disabled on all interfaces. • (Optional) For *number*, specify the NTP version number. The range is 1 to 3. If you do not specify a version, version 3 is used. • (Optional) For *keyid*, specify the authentication key to use when sending packets to the peer. • (Optional) For *destination-address*, specify the IP address of the peer that is synchronizing its clock to this switch.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.
Step 7		Configure the connected peers to receive NTP broadcast packets as described in the next procedure.

To disable the interface from sending NTP broadcast packets, use the **no ntp broadcast** interface configuration command.

This example shows how to configure a port to send NTP version 2 packets:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ntp broadcast version 2
```

Beginning in privileged EXEC mode, follow these steps to configure the switch to receive NTP broadcast packets from connected peers:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to receive NTP broadcast packets, and enter interface configuration mode.
Step 3	**ntp broadcast client**	Enable the interface to receive NTP broadcast packets. By default, no interfaces receive NTP broadcast packets.
Step 4	**exit**	Return to global configuration mode.

	Command	Purpose
Step 5	**ntp broadcastdelay** *microseconds*	(Optional) Change the estimated round-trip delay between the switch and the NTP broadcast server. The default is 3000 microseconds; the range is 1 to 999999.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable an interface from receiving NTP broadcast packets, use the **no ntp broadcast client** interface configuration command. To change the estimated round-trip delay to the default, use the **no ntp broadcastdelay** global configuration command.

This example shows how to configure a port to receive NTP broadcast packets:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ntp broadcast client
```

Configuring NTP Access Restrictions

You can control NTP access on two levels as described in these sections:

- Creating an Access Group and Assigning a Basic IP Access List, page 6-8
- Disabling NTP Services on a Specific Interface, page 6-10

Creating an Access Group and Assigning a Basic IP Access List

Beginning in privileged EXEC mode, follow these steps to control access to NTP services by using access lists:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ntp access-group** {**query-only** | **serve-only** | **serve** | **peer**} *access-list-number*	Create an access group, and apply a basic IP access list. The keywords have these meanings: • **query-only**—Allows only NTP control queries. • **serve-only**—Allows only time requests. • **serve**—Allows time requests and NTP control queries, but does not allow the switch to synchronize to the remote device. • **peer**—Allows time requests and NTP control queries and allows the switch to synchronize to the remote device. For *access-list-number*, enter a standard IP access list number from 1 to 99.

	Command	Purpose
Step 3	**access-list** *access-list-number* **permit** *source* [*source-wildcard*]	Create the access list. • For *access-list-number*, enter the number specified in Step 2. • Enter the **permit** keyword to permit access if the conditions are matched. • For *source*, enter the IP address of the device that is permitted access to the switch. • (Optional) For *source-wildcard*, enter the wildcard bits to be applied to the source. **Note** When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The access group keywords are scanned in this order, from least restrictive to most restrictive:

1. **peer**—Allows time requests and NTP control queries and allows the switch to synchronize itself to a device whose address passes the access list criteria.
2. **serve**—Allows time requests and NTP control queries, but does not allow the switch to synchronize itself to a device whose address passes the access list criteria.
3. **serve-only**—Allows only time requests from a device whose address passes the access list criteria.
4. **query-only**—Allows only NTP control queries from a device whose address passes the access list criteria.

If the source IP address matches the access lists for more than one access type, the first type is granted. If no access groups are specified, all access types are granted to all devices. If any access groups are specified, only the specified access types are granted.

To remove access control to the switch NTP services, use the **no ntp access-group** {**query-only** | **serve-only** | **serve** | **peer**} global configuration command.

This example shows how to configure the switch to allow itself to synchronize to a peer from access list 99. However, the switch restricts access to allow only time requests from access list 42:

```
Switch# configure terminal
Switch(config)# ntp access-group peer 99
Switch(config)# ntp access-group serve-only 42
Switch(config)# access-list 99 permit 172.20.130.5
Switch(config)# access list 42 permit 172.20.130.6
```

Disabling NTP Services on a Specific Interface

NTP services are enabled on all interfaces by default.

Beginning in privileged EXEC mode, follow these steps to disable NTP packets from being received on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to disable.
Step 3	**ntp disable**	Disable NTP packets from being received on the interface. By default, all interfaces receive NTP packets.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To re-enable receipt of NTP packets on an interface, use the **no ntp disable** interface configuration command.

Configuring the Source IP Address for NTP Packets

When the switch sends an NTP packet, the source IP address is normally set to the address of the interface through which the NTP packet is sent. Use the **ntp source** global configuration command when you want to use a particular source IP address for all NTP packets. The address is taken from the specified interface. This command is useful if the address on an interface cannot be used as the destination for reply packets.

Beginning in privileged EXEC mode, follow these steps to configure a specific interface from which the IP source address is to be taken:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ntp source** *type number*	Specify the interface type and number from which the IP source address is taken. By default, the source address is determined by the outgoing interface.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The specified interface is used for the source address for all packets sent to all destinations. If a source address is to be used for a specific association, use the **source** keyword in the **ntp peer** or **ntp server** global configuration command as described in the "Configuring NTP Associations" section on page 6-5.

Displaying the NTP Configuration

You can use two privileged EXEC commands to display NTP information:

- **show ntp associations** [**detail**]
- **show ntp status**

For detailed information about the fields in these displays, see the *Cisco IOS Configuration Fundamentals Command Reference for Cisco IOS, Release 12.2*.

Configuring Time and Date Manually

If no other source of time is available, you can manually configure the time and date after the system is restarted. The time remains accurate until the next system restart. We recommend that you use manual configuration only as a last resort. If you have an outside source to which the switch can synchronize, you do not need to manually set the system clock.

This section contains this configuration information:

- Setting the System Clock, page 6-11
- Displaying the Time and Date Configuration, page 6-12
- Configuring the Time Zone, page 6-12
- Configuring Summer Time (Daylight Saving Time), page 6-13

Setting the System Clock

If you have an outside source on the network that provides time services, such as an NTP server, you do not need to manually set the system clock.

Beginning in privileged EXEC mode, follow these steps to set the system clock:

	Command	Purpose
Step 1	**clock set** *hh*:*mm*:*ss day month year* or **clock set** *hh*:*mm*:*ss month day year*	Manually set the system clock using one of these formats. • For *hh*:*mm*:*ss*, specify the time in hours (24-hour format), minutes, and seconds. The time specified is relative to the configured time zone. • For *day*, specify the day by date in the month. • For *month*, specify the month by name. • For *year*, specify the year (no abbreviation).
Step 2	**show running-config**	Verify your entries.
Step 3	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to manually set the system clock to 1:32 p.m. on July 23, 2001:

```
Switch# clock set 13:32:00 23 July 2001
```

Displaying the Time and Date Configuration

To display the time and date configuration, use the **show clock** [**detail**] privileged EXEC command.

The system clock keeps an *authoritative* flag that shows whether the time is authoritative (believed to be accurate). If the system clock has been set by a timing source such as NTP, the flag is set. If the time is not authoritative, it is used only for display purposes. Until the clock is authoritative and the *authoritative* flag is set, the flag prevents peers from synchronizing to the clock when the peers' time is invalid.

The symbol that precedes the **show clock** display has this meaning:

- *—Time is not authoritative.
- (blank)—Time is authoritative.
- .—Time is authoritative, but NTP is not synchronized.

Configuring the Time Zone

Beginning in privileged EXEC mode, follow these steps to manually configure the time zone:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**clock timezone** *zone hours-offset* [*minutes-offset*]	Set the time zone. The switch keeps internal time in universal time coordinated (UTC), so this command is used only for display purposes and when the time is manually set. • For *zone*, enter the name of the time zone to be displayed when standard time is in effect. The default is UTC. • For *hours-offset*, enter the hours offset from UTC. • (Optional) For *minutes-offset*, enter the minutes offset from UTC.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The *minutes-offset* variable in the **clock timezone** global configuration command is available for those cases where a local time zone is a percentage of an hour different from UTC. For example, the time zone for some sections of Atlantic Canada (AST) is UTC-3.5, where the 3 means 3 hours and .5 means 50 percent. In this case, the necessary command is **clock timezone AST -3 30**.

To set the time to UTC, use the **no clock timezone** global configuration command.

Configuring Summer Time (Daylight Saving Time)

Beginning in privileged EXEC mode, follow these steps to configure summer time (daylight saving time) in areas where it starts and ends on a particular day of the week each year:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**clock summer-time** *zone* **recurring** [*week day month hh*:*mm week day month hh*:*mm* [*offset*]]	Configure summer time to start and end on the specified days every year. Summer time is disabled by default. If you specify **clock summer-time** *zone* **recurring** without parameters, the summer time rules default to the United States rules. • For *zone*, specify the name of the time zone (for example, PDT) to be displayed when summer time is in effect. • (Optional) For *week*, specify the week of the month (1 to 5 or **last**). • (Optional) For *day*, specify the day of the week (Sunday, Monday...). • (Optional) For *month*, specify the month (January, February...). • (Optional) For *hh*:*mm*, specify the time (24-hour format) in hours and minutes. • (Optional) For *offset*, specify the number of minutes to add during summer time. The default is 60.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The first part of the **clock summer-time** global configuration command specifies when summer time begins, and the second part specifies when it ends. All times are relative to the local time zone. The start time is relative to standard time. The end time is relative to summer time. If the starting month is after the ending month, the system assumes that you are in the southern hemisphere.

This example shows how to specify that summer time starts on the first Sunday in April at 02:00 and ends on the last Sunday in October at 02:00:

```
Switch(config)# clock summer-time PDT recurring 1 Sunday April 2:00 last Sunday October
2:00
```

Beginning in privileged EXEC mode, follow these steps if summer time in your area does not follow a recurring pattern (configure the exact date and time of the next summer time events):

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**clock summer-time** *zone* **date** [*month date year hh*:*mm month date year hh*:*mm* [*offset*]] or **clock summer-time** *zone* **date** [*date month year hh*:*mm date month year hh*:*mm* [*offset*]]	Configure summer time to start on the first date and end on the second date. Summer time is disabled by default. • For *zone*, specify the name of the time zone (for example, PDT) to be displayed when summer time is in effect. • (Optional) For *week*, specify the week of the month (1 to 5 or **last**). • (Optional) For *day*, specify the day of the week (Sunday, Monday...). • (Optional) For *month*, specify the month (January, February...). • (Optional) For *hh*:*mm*, specify the time (24-hour format) in hours and minutes. • (Optional) For *offset*, specify the number of minutes to add during summer time. The default is 60.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The first part of the **clock summer-time** global configuration command specifies when summer time begins, and the second part specifies when it ends. All times are relative to the local time zone. The start time is relative to standard time. The end time is relative to summer time. If the starting month is after the ending month, the system assumes that you are in the southern hemisphere.

To disable summer time, use the **no clock summer-time** global configuration command.

This example shows how to set summer time to start on October 12, 2000, at 02:00, and end on April 26, 2001, at 02:00:

```
Switch(config)# clock summer-time pdt date 12 October 2000 2:00 26 April 2001 2:00
```

Configuring a System Name and Prompt

You configure the system name on the switch to identify it. By default, the system name and prompt are *Switch*.

If you have not configured a system prompt, the first 20 characters of the system name are used as the system prompt. A greater-than symbol [>] is appended. The prompt is updated whenever the system name changes.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2* and the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*.

This section contains this configuration information:

- Default System Name and Prompt Configuration, page 6-15
- Configuring a System Name, page 6-15
- Understanding DNS, page 6-15

Default System Name and Prompt Configuration

The default switch system name and prompt is *Switch*.

Configuring a System Name

Beginning in privileged EXEC mode, follow these steps to manually configure a system name:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**hostname** *name*	Manually configure a system name. The default setting is *switch*. The name must follow the rules for ARPANET host names. They must start with a letter, end with a letter or digit, and have as interior characters only letters, digits, and hyphens. Names can be up to 63 characters.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

When you set the system name, it is also used as the system prompt.

To return to the default hostname, use the **no hostname** global configuration command.

Understanding DNS

The DNS protocol controls the Domain Name System (DNS), a distributed database with which you can map host names to IP addresses. When you configure DNS on your switch, you can substitute the host name for the IP address with all IP commands, such as **ping**, **telnet**, **connect**, and related Telnet support operations.

IP defines a hierarchical naming scheme that allows a device to be identified by its location or domain. Domain names are pieced together with periods (.) as the delimiting characters. For example, Cisco Systems is a commercial organization that IP identifies by a *com* domain name, so its domain name is *cisco.com*. A specific device in this domain, for example, the File Transfer Protocol (FTP) system is identified as *ftp.cisco.com*.

To keep track of domain names, IP has defined the concept of a domain name server, which holds a cache (or database) of names mapped to IP addresses. To map domain names to IP addresses, you must first identify the host names, specify the name server that is present on your network, and enable the DNS.

This section contains this configuration information:

- Default DNS Configuration, page 6-16
- Setting Up DNS, page 6-16
- Displaying the DNS Configuration, page 6-17

Default DNS Configuration

Table 6-2 shows the default DNS configuration.

Table 6-2 Default DNS Configuration

Feature	Default Setting
DNS enable state	Enabled.
DNS default domain name	None configured.
DNS servers	No name server addresses are configured.

Setting Up DNS

Beginning in privileged EXEC mode, follow these steps to set up your switch to use the DNS:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip domain-name** *name*	Define a default domain name that the software uses to complete unqualified host names (names without a dotted-decimal domain name). Do not include the initial period that separates an unqualified name from the domain name. At boot time, no domain name is configured; however, if the switch configuration comes from a BOOTP or DHCP server, then the default domain name might be set by the BOOTP or DHCP server (if the servers were configured with this information).
Step 3	**ip name-server** *server-address1* [*server-address2* ... *server-address6*]	Specify the address of one or more name servers to use for name and address resolution. You can specify up to six name servers. Separate each server address with a space. The first server specified is the primary server. The switch sends DNS queries to the primary server first. If that query fails, the backup servers are queried.
Step 4	**ip domain-lookup**	(Optional) Enable DNS-based host name-to-address translation on your switch. This feature is enabled by default. If your network devices require connectivity with devices in networks for which you do not control name assignment, you can dynamically assign device names that uniquely identify your devices by using the global Internet naming scheme (DNS).
Step 5	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

If you use the switch IP address as its hostname, the IP address is used and no DNS query occurs. If you configure a hostname that contains no periods (.), a period followed by the default domain name is appended to the hostname before the DNS query is made to map the name to an IP address. The default domain name is the value set by the **ip domain-name** global configuration command. If there is a period (.) in the hostname, the software looks up the IP address without appending any default domain name to the hostname.

To remove a domain name, use the **no ip domain-name** *name* global configuration command. To remove a name server address, use the **no ip name-server** *server-address* global configuration command. To disable DNS on the switch, use the **no ip domain-lookup** global configuration command.

Displaying the DNS Configuration

To display the DNS configuration information, use the **show running-config** privileged EXEC command.

Creating a Banner

You can configure a message-of-the-day (MOTD) and a login banner. The MOTD banner displays on all connected terminals at login and is useful for sending messages that affect all network users (such as impending system shutdowns).

The login banner also displays on all connected terminals. It appears after the MOTD banner and before the login prompts.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference for Cisco IOS, Release12.2*.

This section contains this configuration information:

- Default Banner Configuration, page 6-17
- Configuring a Message-of-the-Day Login Banner, page 6-18
- Configuring a Login Banner, page 6-18

Default Banner Configuration

The MOTD and login banners are not configured.

Configuring a Message-of-the-Day Login Banner

You can create a single or multiline message banner that appears on the screen when someone logs in to the switch.

Beginning in privileged EXEC mode, follow these steps to configure a MOTD login banner:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**banner motd** *c message c*	Specify the message of the day. For *c*, enter the delimiting character of your choice, for example, a pound sign (#), and press the **Return** key. The delimiting character signifies the beginning and end of the banner text. Characters after the ending delimiter are discarded. For *message*, enter a banner message up to 255 characters. You cannot use the delimiting character in the message.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete the MOTD banner, use the **no banner motd** global configuration command.

This example shows how to configure a MOTD banner for the switch by using the pound sign (#) symbol as the beginning and ending delimiter:

```
Switch(config)# banner motd #
This is a secure site. Only authorized users are allowed.
For access, contact technical support.
#
Switch(config)#
```

This example shows the banner displayed from the previous configuration:

```
Unix> telnet 172.2.5.4
Trying 172.2.5.4...
Connected to 172.2.5.4.
Escape character is '^]'.

This is a secure site. Only authorized users are allowed.
For access, contact technical support.

User Access Verification

Password:
```

Configuring a Login Banner

You can configure a login banner to be displayed on all connected terminals. This banner appears after the MOTD banner and before the login prompt.

Beginning in privileged EXEC mode, follow these steps to configure a login banner:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**banner login** *c message c*	Specify the login message. For *c*, enter the delimiting character of your choice, for example, a pound sign (#), and press the **Return** key. The delimiting character signifies the beginning and end of the banner text. Characters after the ending delimiter are discarded. For *message*, enter a login message up to 255 characters. You cannot use the delimiting character in the message.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete the login banner, use the **no banner login** global configuration command.

This example shows how to configure a login banner for the switch by using the dollar sign ($) symbol as the beginning and ending delimiter:

```
Switch(config)# banner login $
Access for authorized users only. Please enter your username and password.
$
Switch(config)#
```

Managing the MAC Address Table

The MAC address table contains address information that the switch uses to forward traffic between ports. All MAC addresses in the address table are associated with one or more ports. The address table includes these types of addresses:

- Dynamic address: a source MAC address that the switch learns and then ages when it is not in use.
- Static address: a manually entered unicast or multicast address that does not age and that is not lost when the switch resets.

The address table lists the destination MAC address, the associated VLAN ID, and port number associated with the address.

Note For complete syntax and usage information for the commands used in this section, see the command reference for this release.

This section contains this configuration information:

- Building the Address Table, page 6-20
- MAC Addresses and VLANs, page 6-20
- Default MAC Address Table Configuration, page 6-20
- Changing the Address Aging Time, page 6-21
- Removing Dynamic Address Entries, page 6-21

- Configuring MAC Address Notification Traps, page 6-21
- Adding and Removing Static Address Entries, page 6-23
- Configuring Unicast MAC Address Filtering, page 6-24
- Displaying Address Table Entries, page 6-26

Building the Address Table

With multiple MAC addresses supported on all ports, you can connect any port on the switch to individual workstations, repeaters, switches, routers, or other network devices. The switch provides dynamic addressing by learning the source address of packets it receives on each port and adding the address and its associated port number to the address table. As stations are added or removed from the network, the switch updates the address table, adding new dynamic addresses and aging out those that are not in use.

The aging interval is configured on a per-switch basis. However, the switch maintains an address table for each VLAN, and STP can accelerate the aging interval on a per-VLAN basis.

The switch sends packets between any combination of ports, based on the destination address of the received packet. Using the MAC address table, the switch forwards the packet only to the port or ports associated with the destination address. If the destination address is on the port that sent the packet, the packet is filtered and not forwarded. The switch always uses the store-and-forward method: complete packets are stored and checked for errors before transmission.

MAC Addresses and VLANs

All addresses are associated with a VLAN. An address can exist in more than one VLAN and have different destinations in each. Multicast addresses, for example, could be forwarded to port 1 in VLAN 1 and ports 9, 10, and 11 in VLAN 5.

Each VLAN maintains its own logical address table. A known address in one VLAN is unknown in another until it is learned or statically associated with a port in the other VLAN. Addresses that are statically entered in one VLAN must be configured as static addresses in all other VLANs or remain unlearned in the other VLANs.

Default MAC Address Table Configuration

Table 6-3 shows the default MAC address table configuration.

Table 6-3 Default MAC Address Table Configuration

Feature	Default Setting
Aging time	300 seconds
Dynamic addresses	Automatically learned
Static addresses	None configured

Changing the Address Aging Time

Dynamic addresses are source MAC addresses that the switch learns and then ages when they are not in use. You can change the aging time setting for all VLANs or for a specified VLAN.

Setting too short an aging time can cause addresses to be prematurely removed from the table. Then when the switch receives a packet for an unknown destination, it floods the packet to all ports in the same VLAN as the receiving port. This unnecessary flooding can impact performance. Setting too long an aging time can cause the address table to be filled with unused addresses, which prevents new addresses from being learned.

Beginning in privileged EXEC mode, follow these steps to configure the dynamic address table aging time:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mac address-table aging-time** [**0** | *10-1000000*] [**vlan** *vlan-id*]	Set the length of time that a dynamic entry remains in the MAC address table after the entry is used or updated. The range is 10 to 1000000 seconds. The default is 300. You can also enter 0, which disables aging. Static address entries are never aged or removed from the table. For *vlan-id*, valid IDs are 1 to 4094.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mac address-table aging-time**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default value, use the **no mac address-table aging-time** global configuration command.

Removing Dynamic Address Entries

To remove all dynamic entries, use the **clear mac address-table dynamic** command in privileged EXEC mode. You can also remove a specific MAC address (**clear mac address-table dynamic address** *mac-address*), remove all addresses on the specified physical port or port channel (**clear mac address-table dynamic interface** *interface-id*), or remove all addresses on a specified VLAN (**clear mac address-table dynamic vlan** *vlan-id*).

To verify that dynamic entries have been removed, use the **show mac address-table dynamic** privileged EXEC command.

Configuring MAC Address Notification Traps

MAC address notification enables you to track users on a network by storing the MAC address activity on the switch. Whenever the switch learns or removes a MAC address, an SNMP notification can be generated and sent to the NMS. If you have many users coming and going from the network, you can set a trap interval time to bundle the notification traps and reduce network traffic. The MAC notification history table stores the MAC address activity for each hardware port for which the trap is enabled. MAC address notifications are generated for dynamic and secure MAC addresses; events are not generated for self addresses, multicast addresses, or other static addresses.

Beginning in privileged EXEC mode, follow these steps to configure the switch to send MAC address notification traps to an NMS host:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**snmp-server host** *host-addr* {**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type*	Specify the recipient of the trap message. • For *host-addr*, specify the name or address of the NMS. • Specify **traps** (the default) to send SNMP traps to the host. Specify **informs** to send SNMP informs to the host. • Specify the SNMP version to support. Version 1, the default, is not available with informs. • For *community-string,* specify the string to send with the notification operation. Though you can set this string by using the **snmp-server host** command, we recommend that you define this string by using the **snmp-server community** command before using the **snmp-server host** command. • For *notification-type*, use the **mac-notification** keyword.
Step 3	**snmp-server enable traps mac-notification**	Enable the switch to send MAC address traps to the NMS.
Step 4	**mac address-table notification**	Enable the MAC address notification feature.
Step 5	**mac address-table notification** [**interval** *value*] | [**history-size** *value*]	Enter the trap interval time and the history table size. • (Optional) For **interval** *value,* specify the notification trap interval in seconds between each set of traps that are generated to the NMS. The range is 0 to 2147483647 seconds; the default is 1 second. • (Optional) For **history-size** *value*, specify the maximum number of entries in the MAC notification history table. The range is 0 to 500; the default is 1.
Step 6	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 2 interface on which to enable the SNMP MAC address notification trap.
Step 7	**snmp trap mac-notification** {**added** | **removed**}	Enable the MAC address notification trap. • Enable the MAC notification trap whenever a MAC address is **added** on this interface. • Enable the MAC notification trap whenever a MAC address is **removed** from this interface.
Step 8	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 9	**show mac address-table notification interface** **show running-config**	Verify your entries.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the switch from sending MAC address notification traps, use the **no snmp-server enable traps mac-notification** global configuration command. To disable the MAC address notification traps on a specific interface, use the **no snmp trap mac-notification** {**added** | **removed**} interface configuration command. To disable the MAC address notification feature, use the **no mac address-table notification** global configuration command.

This example shows how to specify 172.20.10.10 as the NMS, enable the switch to send MAC address notification traps to the NMS, enable the MAC address notification feature, set the interval time to 60 seconds, set the history-size to 100 entries, and enable traps whenever a MAC address is added on the specified port.

```
Switch(config)# snmp-server host 172.20.10.10 traps private
Switch(config)# snmp-server enable traps mac-notification
Switch(config)# mac address-table notification
Switch(config)# mac address-table notification interval 60
Switch(config)# mac address-table notification history-size 100
Switch(config)# interface fastethernet0/4
Switch(config-if)# snmp trap mac-notification added
```

You can verify the previous commands by entering the **show mac address-table notification interface** and the **show mac address-table notification** privileged EXEC commands.

Adding and Removing Static Address Entries

A static address has these characteristics:

- It is manually entered in the address table and must be manually removed.
- It can be a unicast or multicast address.
- It does not age and is retained when the switch restarts.

You can add and remove static addresses and define the forwarding behavior for them. The forwarding behavior determines how a port that receives a packet forwards it to another port for transmission. Because all ports are associated with at least one VLAN, the switch acquires the VLAN ID for the address from the ports that you specify. You can specify a different list of destination ports for each source port.

A static address in one VLAN must be a static address in other VLANs. A packet with a static address that arrives on a VLAN where it has not been statically entered is flooded to all ports and not learned.

You add a static address to the address table by specifying the destination MAC address (unicast or multicast) and the VLAN from which it is received. Packets received with this destination address are forwarded to the interface specified with the *interface-id* option.

Beginning in privileged EXEC mode, follow these steps to add a static address:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id*	Add a static address to the MAC address table. • For *mac-addr*, specify the destination MAC address (unicast or multicast) to add to the address table. Packets with this destination address received in the specified VLAN are forwarded to the specified interface. • For *vlan-id*, specify the VLAN for which the packet with the specified MAC address is received. Valid VLAN IDs are 1 to 4094. • For *interface-id*, specify the interface to which the received packet is forwarded. Valid interfaces include physical ports and port channels. For *interface-id*, specify the interface to which the received packet is forwarded. Valid interfaces include physical ports or port channels. For static multicast addresses, you can enter multiple interface IDs. For static unicast addresses, you can enter only one interface at a time, but you can enter the command multiple times with the same MAC address and VLAN ID.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mac address-table static**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove static entries from the address table, use the **no mac address-table static** *mac-addr* **vlan** *vlan-id* [**interface** *interface-id*] global configuration command.

This example shows how to add the static address c2f3.220a.12f4 to the MAC address table. When a packet is received in VLAN 4 with this MAC address as its destination address, the packets is forwarded to the specified interface:

```
Switch(config)# mac address-table static c2f3.220a.12f4 vlan 4 interface
gigabitethernet0/1
```

Configuring Unicast MAC Address Filtering

When unicast MAC address filtering is enabled, the switch drops packets with specific source or destination MAC addresses. This feature is disabled by default and only supports unicast static addresses.

Follow these guidelines when using this feature:

- Multicast MAC addresses, broadcast MAC addresses, and router MAC addresses are not supported. If you specify one of these addresses when entering the **mac address-table static** *mac-addr* **vlan** *vlan-id* **drop** global configuration command, one of these messages appears:

  ```
  % Only unicast addresses can be configured to be dropped

  % CPU destined address cannot be configured as drop address
  ```

- Packets that are forwarded to the CPU are also not supported.

- If you add a unicast MAC address as a static address and configure unicast MAC address filtering, the switch either adds the MAC address as a static address or drops packets with that MAC address, depending on which command was entered last. The second command that you entered overrides the first command.

  For example, if you enter the **mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id* global configuration command followed by the **mac address-table static** *mac-addr* **vlan** *vlan-id* **drop** command, the switch drops packets with the specified MAC address as a source or destination.

  If you enter the **mac address-table static** *mac-addr* **vlan** *vlan-id* **drop** global configuration command followed by the **mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id* command, the switch adds the MAC address as a static address.

You enable unicast MAC address filtering and configure the switch to drop packets with a specific address by specifying the source or destination unicast MAC address and the VLAN from which it is received.

Beginning in privileged EXEC mode, follow these steps to configure the switch to drop a source or destination unicast static address:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mac address-table static** *mac-addr* **vlan** *vlan-id* **drop**	Enable unicast MAC address filtering and configure the switch to drop a packet with the specified source or destination unicast static address. • For *mac-addr*, specify a source or destination unicast MAC address. Packets with this MAC address are dropped. • For *vlan-id*, specify the VLAN for which the packet with the specified MAC address is received. Valid VLAN IDs are 1 to 4094.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mac address-table static**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable unicast MAC address filtering, use the **no mac address-table static** *mac-addr* **vlan** *vlan-id* global configuration command.

This example shows how to enable unicast MAC address filtering and to configure the switch to drop packets that have a source or destination address of c2f3.220a.12f4. When a packet is received in VLAN 4 with this MAC address as its source or destination, the packet is dropped:

```
Switch(config)# mac address-table static c2f3.220a.12f4 vlan 4 drop
```

Displaying Address Table Entries

You can display the MAC address table by using one or more of the privileged EXEC commands described in Table 6-4:

Table 6-4 Commands for Displaying the MAC Address Table

Command	Description
show mac address-table address	Displays MAC address table information for the specified MAC address.
show mac address-table aging-time	Displays the aging time in all VLANs or the specified VLAN.
show mac address-table count	Displays the number of addresses present in all VLANs or the specified VLAN.
show mac address-table dynamic	Displays dynamic MAC address table entries only.
show mac address-table interface	Displays the MAC address table information for the specified interface.
show mac address-table multicast	Displays the Layer 2 multicast entries for all VLANs or the specified VLAN.
show mac address-table static	Displays static MAC address table entries only.
show mac address-table vlan	Displays the MAC address table information for the specified VLAN.

Optimizing System Resources for User-Selected Features

By using Switch Database Management (SDM) templates, you can configure memory resources in the switch to optimize support for specific features, depending on how the switch is used in your network. You can select one of four templates to specify how system resources are allocated. You can then approximate the maximum number of unicast MAC addresses, Internet Group Management Protocol (IGMP) groups, quality of service (QoS) access control entries (ACEs), security ACEs, unicast routes, multicast routes, subnet VLANs (routed interfaces), and Layer 2 VLANs that can be configured on the switch.

The four templates prioritize system memory to optimize support for these types of features:

- QoS and security ACEs—The access template might typically be used in an access switch at the network edge where the route table sizes might not be substantial. Filtering and QoS might be more important because an access switch is the entry to the whole network.
- Routing—The routing template maximizes system resources for unicast routing, typically required for a router or aggregator in the center of a network.
- VLANs—The VLAN template disables routing and supports the maximum number of unicast MAC addresses. It would typically be selected for a switch used as a Layer 2 switch.
- Default—The default template gives balance to all functionalities (QoS, ACLs, unicast routing, multicast routing, VLANs and MAC addresses).

You can also enable the switch to support 144-bit Layer 3 TCAM, allowing extra fields in the stored routing tables, by reformatting the routing table memory allocation. Using the **extended-match** keyword with the default, access, or routing templates reformats the allocated TCAM by reducing the number of allowed unicast routes, and storing extra routing information in the lower 72 bits of the Layer 3 TCAM. The 144-bit Layer 3 TCAM is required when running the Web Cache Communication Protocol (WCCP) or multiple VPN routing/forwarding (multi-VRF) instances in customer edge (CE) devices (multi-VRF CE) on the switch.

Table 6-5 lists the approximate number of each resource supported in each of the four templates for Catalyst 3550 Gigabit Ethernet switches. Table 6-6 compares the four templates for a Catalyst 3550 switch with primarily Fast Ethernet ports.

The first six rows in the tables (unicast MAC addresses through multicast routes) represent approximate hardware boundaries set when a template is selected. If a section of a hardware resource is full, all processing overflow is sent to the CPU, seriously impacting switch performance.

The last two rows, the total number of routed ports and SVIs and the number of Layer 2 VLANs, are guidelines used to calculate hardware resource consumption related to the other resource parameters.

The number of subnet VLANs (routed ports and SVIs) are not limited by software and can be set to a number higher than indicated in the tables. If the number of subnet VLANs configured is lower or equal to the number in the tables, the number of entries in each category (unicast addresses, IGMP groups, and so on) for each template will be as shown. As the number of subnet VLANs increases, CPU utilization typically increases. If the number of subnet VLANs increases beyond the number shown in the tables, the number of supported entries in each category could decrease depending on features that are enabled. For example, if PIM-DVMRP is enabled with more than 16 subnet VLANs, the number of entries for multicast routes will be in the range of 1K-5K entries for the access template.

Table 6-5 Approximate Resources Allowed in Each Template for Gigabit Ethernet Switches

Resource	Default Template	Access Template	Routing Template	VLAN Template
Unicast MAC addresses	6 K	2 K	6 K	12 K
IGMP groups (managed by Layer 2 multicast features such as MVR or IGMP snooping)	6 K	8 K	6 K	6 K
QoS classification ACEs	2 K	2 K	1 K	2 K
Security ACEs	2 K	4 K	1 K	2 K
Unicast routes	12 K or 6 K[1]	4 K or 2 K[1]	24 K or 12 K[1]	0
Multicast routes	6 K	8 K	6 K	0
Subnet VLANs (routed ports and SVIs)	16	16	16	16
Layer 2 VLANs	1 K	1 K	1 K	1 K

1. When the **extended-match** keyword is used with the listed template. This keyword affects only the number of unicast routes allowed.

Table 6-6 Approximate Resources Allowed in Each Template for Fast Ethernet Switches

Resource	Default Template	Access Template	Routing Template	VLAN Template
Unicast MAC addresses	5 K	1 K	5 K	8 K
IGMP groups (managed by Layer 2 multicast features such as MVR and IGMP snooping)	1 K	2 K	1 K	1 K
QoS classification ACEs	1 K	1K	512	1 K
Security ACEs	1 K	2 K	512	1 K
Unicast routes	8 K or 4 K[1]	2 K or 1 K[1]	16 K or 8 K[1]	0
Multicast routes	1 K	2 K	1 K	0

Table 6-6 Approximate Resources Allowed in Each Template for Fast Ethernet Switches (continued)

Resource	Default Template	Access Template	Routing Template	VLAN Template
Subnet VLANs (routed ports and SVIs)	8	8	8	8
Layer 2 VLANs	1 K	1 K	1 K	1 K

1. When the **extended-match** keyword is used with the listed template. This keyword affects only the number of unicast routes allowed.

Using the Templates

Follow these guidelines when using the SDM templates:

- The maximum number of resources allowed in each template is an approximation and depends upon the actual number of other features configured. For example, in the default template for the Catalyst 3550-12T, if your switch has more than 16 routed interfaces configured, the number of multicast or unicast routes that can be accommodated by hardware might be fewer than shown.
- Using the **sdm prefer vlan** global configuration command disables routing capability in the switch. Any routing configurations are rejected after the reload, and previously configured routing options might be lost. Use the **sdm prefer vlan** global configuration command only on switches intended for Layer 2 switching with no routing.
- Do not use the routing template if you are not enabling routing on your switch. Entering the **sdm prefer routing** global configuration command on a switch does not enable routing, but it would prevent other features from using the memory allocated to unicast and multicast routing in the routing template, which could be up to 30 K in Gigabit Ethernet switches and 17 K in Fast Ethernet switches.
- You must use the **extended-match** keyword to support 144-bit Layer 3 TCAM when WCCP or multi-VRF CE is enabled on the switch. This keyword is not supported on the VLAN template.

This procedure shows how to change the SDM template from the default. The switch must reload before the configuration takes effect. If you use the **show sdm prefer** privileged EXEC command before the switch reloads, the previous configuration (in this case, the default) appears.

Beginning in privileged EXEC mode, follow these steps to use the SDM template to maximize feature usage:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**sdm prefer** {**access** [**extended-match**] | **extended-match** | **routing** [**extended-match**] | **vlan**}	Specify the SDM template to be used on the switch: The keywords have these meanings: • **access**—Maximizes the use of QoS classification ACEs and security ACEs on the switch. • **routing**—Maximizes routing on the switch. • **vlan**—Maximizes VLAN configuration on the switch with no routing allowed. • **extended-match**—Reformats routing memory space to allow 144-bit Layer 3 TCAM support in the default, access, or routing template to support WCCP or multi-VRF CE. The default template (if none of these is configured) balances the use of unicast MAC addresses, IGMP groups, QoS ACEs, security ACEs, unicast and multicast routes, routed interfaces, and Layer 2 VLANs.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**reload**	Reload the operating system.

After the system reboots, you can use the **show sdm prefer** privileged EXEC command to verify the new template configuration. If you use the **show sdm prefer** command before the **reload** privileged EXEC command, the previous template appears instead of the new one.

To return to the default template, use the **no sdm prefer** global configuration command.

This example shows how to configure a switch with the routing template and verify the configuration:

```
Switch(config)# sdm prefer routing
Switch(config)# end
Switch# reload
Proceed with reload? [confirm]
```

Managing the ARP Table

To communicate with a device (over Ethernet, for example), the software first must determine the 48-bit MAC or the local data link address of that device. The process of determining the local data link address from an IP address is called *address resolution*.

The Address Resolution Protocol (ARP) associates a host IP address with the corresponding media or MAC addresses and the VLAN ID. Taking an IP address as input, ARP determines the associated MAC address. Once a MAC address is determined, the IP-MAC address association is stored in an ARP cache for rapid retrieval. Then the IP datagram is encapsulated in a link-layer frame and sent over the network. Encapsulation of IP datagrams and ARP requests and replies on IEEE 802 networks other than Ethernet is specified by the Subnetwork Access Protocol (SNAP). By default, standard Ethernet-style ARP encapsulation (represented by the **arpa** keyword) is enabled on the IP interface.

ARP entries added manually to the table do not age and must be manually removed.

For CLI procedures, see the Cisco IOS Release 12.2 documentation on Cisco.com.

CHAPTER 7

Configuring Switch-Based Authentication

This chapter describes how to configure switch-based authentication on the Catalyst 3550 switch. This chapter consists of these sections:

- Preventing Unauthorized Access to Your Switch, page 7-1
- Protecting Access to Privileged EXEC Commands, page 7-2
- Controlling Switch Access with TACACS+, page 7-9
- Controlling Switch Access with RADIUS, page 7-17
- Controlling Switch Access with Kerberos, page 7-32
- Configuring the Switch for Local Authentication and Authorization, page 7-36
- Configuring the Switch for Secure Shell, page 7-37
- Configuring the Switch for Secure Socket Layer HTTP, page 7-41
- Configuring the Switch for Secure Copy Protocol, page 7-48

Preventing Unauthorized Access to Your Switch

You can prevent unauthorized users from reconfiguring your switch and viewing configuration information. Typically, you want network administrators to have access to your switch while you restrict access to users who dial from outside the network through an asynchronous port, connect from outside the network through a serial port, or connect through a terminal or workstation from within the local network.

To prevent unauthorized access into your switch, you should configure one or more of these security features:

- At a minimum, you should configure passwords and privileges at each switch port. These passwords are locally stored on the switch. When users attempt to access the switch through a port or line, they must enter the password specified for the port or line before they can access the switch. For more information, see the "Protecting Access to Privileged EXEC Commands" section on page 7-2.
- For an additional layer of security, you can also configure username and password pairs, which are locally stored on the switch. These pairs are assigned to lines or interfaces and authenticate each user before that user can access the switch. If you have defined privilege levels, you can also assign a specific privilege level (with associated rights and privileges) to each username and password pair. For more information, see the "Configuring Username and Password Pairs" section on page 7-6.

- If you want to use username and password pairs, but you want to store them centrally on a server instead of locally, you can store them in a database on a security server. Multiple networking devices can then use the same database to obtain user authentication (and, if necessary, authorization) information. For more information, see the "Controlling Switch Access with TACACS+" section on page 7-9.

Protecting Access to Privileged EXEC Commands

A simple way of providing terminal access control in your network is to use passwords and assign privilege levels. Password protection restricts access to a network or network device. Privilege levels define what commands users can enter after they have logged into a network device.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Security Command Reference for Cisco IOS Release 12.2*.

This section describes how to control access to the configuration file and privileged EXEC commands. It contains this configuration information:

- Default Password and Privilege Level Configuration, page 7-2
- Setting or Changing a Static Enable Password, page 7-3
- Protecting Enable and Enable Secret Passwords with Encryption, page 7-3
- Disabling Password Recovery, page 7-5
- Setting a Telnet Password for a Terminal Line, page 7-6
- Configuring Username and Password Pairs, page 7-6
- Configuring Multiple Privilege Levels, page 7-7

Default Password and Privilege Level Configuration

Table 7-1 shows the default password and privilege level configuration.

Table 7-1 Default Password and Privilege Levels

Feature	Default Setting
Enable password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is not encrypted in the configuration file.
Enable secret password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is encrypted before it is written to the configuration file.
Line password	No password is defined.

Setting or Changing a Static Enable Password

The enable password controls access to the privileged EXEC mode. Beginning in privileged EXEC mode, follow these steps to set or change a static enable password:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**enable password** *password*	Define a new password or change an existing password for access to privileged EXEC mode. By default, no password is defined. For *password*, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. It can contain the question mark (?) character if you precede the question mark with the key combination Crtl-v when you create the password; for example, to create the password abc?123, do this: Enter **abc**. Enter **Crtl-v**. Enter **?123**. When the system prompts you to enter the enable password, you need not precede the question mark with the Ctrl-v; you can simply enter abc?123 at the password prompt.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file. The enable password is not encrypted and can be read in the switch configuration file.

To remove the password, use the **no enable password** global configuration command.

This example shows how to change the enable password to *l1u2c3k4y5*. The password is not encrypted and provides access to level 15 (traditional privileged EXEC mode access):

```
Switch(config)# enable password l1u2c3k4y5
```

Protecting Enable and Enable Secret Passwords with Encryption

To provide an additional layer of security, particularly for passwords that cross the network or that are stored on a TFTP server, you can use either the **enable password** or **enable secret** global configuration commands. Both commands accomplish the same thing; that is, you can establish an encrypted password that users must enter to access privileged EXEC mode (the default) or any privilege level you specify.

We recommend that you use the **enable secret** command because it uses an improved encryption algorithm.

If you configure the **enable secret** command, it takes precedence over the **enable password** command; the two commands cannot be in effect simultaneously.

Beginning in privileged EXEC mode, follow these steps to configure encryption for enable and enable secret passwords:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**enable password** [**level** *level*] {*password* | *encryption-type encrypted-password*} or **enable secret** [**level** *level*] {*password* | *encryption-type encrypted-password*}	Define a new password or change an existing password for access to privileged EXEC mode. or Define a secret password, which is saved using a nonreversible encryption method. • (Optional) For *level*, the range is from 0 to 15. Level 1 is normal user EXEC mode privileges. The default level is 15 (privileged EXEC mode privileges). • For *password*, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined. • (Optional) For *encryption-type*, only type 5, a Cisco proprietary encryption algorithm, is available. If you specify an encryption type, you must provide an encrypted password—an encrypted password you copy from another Catalyst 3550 switchconfiguration. **Note** If you specify an encryption type and then enter a clear text password, you can not re-enter privileged EXEC mode. You cannot recover a lost encrypted password by any method.
Step 3	service password-encryption	(Optional) Encrypt the password when the password is defined or when the configuration is written. Encryption prevents the password from being readable in the configuration file.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

If both the enable and enable secret passwords are defined, users must enter the enable secret password.

Use the **level** keyword to define a password for a specific privilege level. After you specify the level and set a password, give the password only to users who need to have access at this level. Use the **privilege level** global configuration command to specify commands accessible at various levels. For more information, see the "Configuring Multiple Privilege Levels" section on page 7-7.

If you enable password encryption, it applies to all passwords including username passwords, authentication key passwords, the privileged command password, and console and virtual terminal line passwords.

To remove a password and level, use the **no enable password** [**level** *level*] or **no enable secret** [**level** *level*] global configuration command. To disable password encryption, use the **no service password-encryption** global configuration command.

This example shows how to configure the encrypted password *1FaD0$Xyti5Rkls3LoyxzS8* for privilege level 2:

```
Switch(config)# enable secret level 2 5 $1$FaD0$Xyti5Rkls3LoyxzS8
```

Disabling Password Recovery

By default, any end user with physical access to the Catalyst 3550 switch can recover from a lost password by interrupting the boot process while the switch is powering on and then by entering a new password.

The password-recovery disable feature protects access to the switch password by disabling part of this functionality. When this feature is enabled, the end user can interrupt the boot process only by agreeing to set the system back to the default configuration. With password recovery disabled, you can still interrupt the boot process and change the password, but the configuration file (config.text) and the VLAN database file (vlan.dat) are deleted.

Note The password recovery disable feature is valid only on Catalyst 3550 Fast Ethernet switches; it is not available for Catalyst 3550 Gigabit Ethernet switches.

Note If you disable password recovery, we recommend that you keep a backup copy of the configuration file on a secure server in case the end user interrupts the boot process and sets the system back to default values. Do not keep a backup copy of the configuration file on the switch. If the switch is operating in VTP transparent mode, we recommend that you also keep a backup copy of the VLAN database file on a secure server. When the switch is returned to the default system configuration, you can download the saved files to the switch by using the Xmodem protocol. For more information, see the "Recovering from a Lost or Forgotten Password" section on page 6-2.

Beginning in privileged EXEC mode, follow these steps to disable password recovery:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no service password-recovery**	Disable password recovery. This setting is saved in an area of the flash memory that is accessible by the boot loader and the software image, but it is not part of the file system and is not accessible by any user.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show version**	Verify the configuration by checking the last few lines of the display.

To re-enable password recovery, use the **service password-recovery** global configuration command.

Note Disabling password recovery will not work if you have set the switch to boot manually by using the **boot manual** global configuration command. This command produces the boot loader prompt (*switch:*) after the switch is power cycled.

Setting a Telnet Password for a Terminal Line

When you power-up your switch for the first time, an automatic setup program runs to assign IP information and to create a default configuration for continued use. The setup program also prompts you to configure your switch for Telnet access through a password. If you neglected to configure this password during the setup program, you can configure it now through the command-line interface (CLI).

Beginning in privileged EXEC mode, follow these steps to configure your switch for Telnet access:

	Command	Purpose
Step 1		Attach a PC or workstation with emulation software to the switch console port. The default data characteristics of the console port are 9600, 8, 1, no parity. You might need to press the Return key several times to see the command-line prompt.
Step 2	**enable password** *password*	Enter privileged EXEC mode.
Step 3	**configure terminal**	Enter global configuration mode.
Step 4	**line vty 0 15**	Configure the number of Telnet sessions (lines), and enter line configuration mode. There are 16 possible sessions on a command-capable switch. The 0 and 15 mean that you are configuring all 16 possible Telnet sessions.
Step 5	**password** *password*	Enter a Telnet password for the line or lines. For *password*, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries. The password is listed under the command **line vty 0 15**.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the password, use the **no password** global configuration command.

This example shows how to set the Telnet password to *let45me67in89*:

```
Switch(config)# line vty 10
Switch(config-line)# password let45me67in89
```

Configuring Username and Password Pairs

You can configure username and password pairs, which are locally stored on the switch. These pairs are assigned to lines or interfaces and authenticate each user before that user can access the switch. If you have defined privilege levels, you can also assign a specific privilege level (with associated rights and privileges) to each username and password pair.

Beginning in privileged EXEC mode, follow these steps to establish a username-based authentication system that requests a login username and a password:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**username** *name* [**privilege** *level*] {**password** *encryption-type password*}	Enter the username, privilege level, and password for each user. • For *name*, specify the user ID as one word. Spaces and quotation marks are not allowed. • (Optional) For *level*, specify the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 1 gives user EXEC mode access. • For *encryption-type*, enter 0 to specify that an unencrypted password will follow. Enter 7 to specify that a hidden password will follow. • For *password*, specify the password the user must enter to gain access to the switch. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the **username** command.
Step 3	**line console 0** or **line vty 0 15**	Enter line configuration mode, and configure the console port (line 0) or the VTY lines (line 0 to 15).
Step 4	**login local**	Enable local password checking at login time. Authentication is based on the username specified in Step 2.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable username authentication for a specific user, use the **no username** *name* global configuration command. To disable password checking and allow connections without a password, use the **no login** line configuration command.

Configuring Multiple Privilege Levels

By default, the software has two modes of password security: user EXEC and privileged EXEC. You can configure up to 16 hierarchical levels of commands for each mode. By configuring multiple passwords, you can allow different sets of users to have access to specified commands.

For example, if you want many users to have access to the **clear line** command, you can assign it level 2 security and distribute the level 2 password fairly widely. But if you want more restricted access to the **configure** command, you can assign it level 3 security and distribute that password to a more restricted group of users.

This section includes this configuration information:

- Setting the Privilege Level for a Command, page 7-8
- Changing the Default Privilege Level for Lines, page 7-9
- Logging into and Exiting a Privilege Level, page 7-9

Setting the Privilege Level for a Command

Beginning in privileged EXEC mode, follow these steps to set the privilege level for a command mode:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**privilege** *mode* **level** *level command*	Set the privilege level for a command. • For *mode*, enter **configure** for global configuration mode, **exec** for EXEC mode, **interface** for interface configuration mode, or **line** for line configuration mode. • For *level*, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the **enable** password. • For *command*, specify the command to which you want to restrict access.
Step 3	**enable password level** *level password*	Specify the enable password for the privilege level. • For *level*, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. • For *password*, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config** or **show privilege**	Verify your entries. The first command displays the password and access level configuration. The second command displays the privilege level configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

When you set a command to a privilege level, all commands whose syntax is a subset of that command are also set to that level. For example, if you set the **show ip traffic** command to level 15, the **show** commands and **show ip** commands are automatically set to privilege level 15 unless you set them individually to different levels.

To return to the default privilege for a given command, use the **no privilege** *mode* **level** *level command* global configuration command.

This example shows how to set the **configure** command to privilege level 14 and define *SecretPswd14* as the password users must enter to use level 14 commands:

```
Switch(config)# privilege exec level 14 configure
Switch(config)# enable password level 14 SecretPswd14
```

Changing the Default Privilege Level for Lines

Beginning in privileged EXEC mode, follow these steps to change the default privilege level for a line:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**line vty** *line*	Select the virtual terminal line on which to restrict access.
Step 3	**privilege level** *level*	Change the default privilege level for the line. For *level*, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the **enable** password.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config** or **show privilege**	Verify your entries. The first command displays the password and access level configuration. The second command displays the privilege level configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Users can override the privilege level you set using the **privilege level** line configuration command by logging in to the line and enabling a different privilege level. They can lower the privilege level by using the **disable** command. If users know the password to a higher privilege level, they can use that password to enable the higher privilege level. You might specify a high level or privilege level for your console line to restrict line usage.

To return to the default line privilege level, use the **no privilege level** line configuration command.

Logging into and Exiting a Privilege Level

Beginning in privileged EXEC mode, follow these steps to log in to a specified privilege level and to exit to a specified privilege level:

	Command	Purpose
Step 1	**enable** *level*	Log in to a specified privilege level. For *level*, the range is 0 to 15.
Step 2	**disable** *level*	Exit to a specified privilege level. For *level*, the range is 0 to 15.

Controlling Switch Access with TACACS+

This section describes how to enable and configure TACACS+, which provides detailed accounting information and flexible administrative control over authentication and authorization processes. TACACS+ is facilitated through authentication, authorization, accounting (AAA) and can be enabled only through AAA commands.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Security Command Reference for Cisco IOS Release 12.2*.

This section contains this configuration information:

- Understanding TACACS+, page 7-10
- TACACS+ Operation, page 7-12
- Configuring TACACS+, page 7-12
- Displaying the TACACS+ Configuration, page 7-17

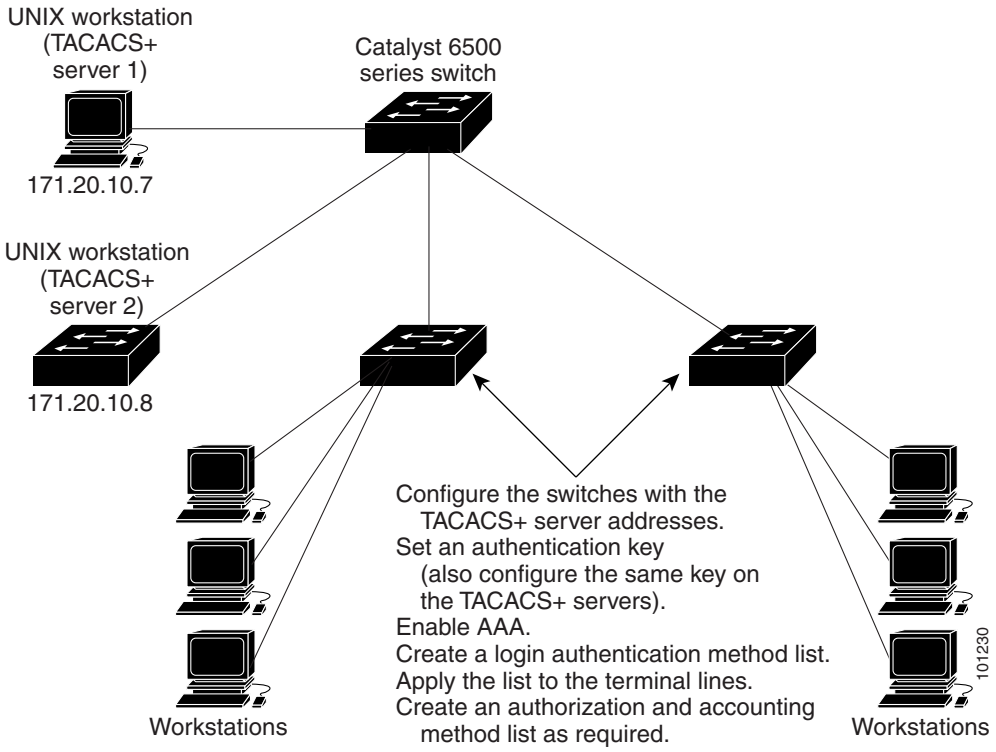
Understanding TACACS+

TACACS+ is a security application that provides centralized validation of users attempting to gain access to your switch. TACACS+ services are maintained in a database on a TACACS+ daemon typically running on a UNIX or Windows NT workstation. You should have access to and should configure a TACACS+ server before the configuring TACACS+ features on your switch.

TACACS+ provides for separate and modular authentication, authorization, and accounting facilities. TACACS+ allows for a single access control server (the TACACS+ daemon) to provide each service—authentication, authorization, and accounting—independently. Each service can be tied into its own database to take advantage of other services available on that server or on the network, depending on the capabilities of the daemon.

The goal of TACACS+ is to provide a method for managing multiple network access points from a single management service. Your switch can be a network access server along with other Cisco routers and access servers. A network access server provides connections to a single user, to a network or subnetwork, and to interconnected networks as shown in Figure 7-1.

Figure 7-1 Typical TACACS+ Network Configuration



TACACS+, administered through the AAA security services, can provide these services:

- Authentication—Provides complete control of authentication through login and password dialog, challenge and response, and messaging support.

  The authentication facility can conduct a dialog with the user (for example, after a username and password are provided, to challenge a user with several questions, such as home address, mother's maiden name, service type, and social security number). The TACACS+ authentication service can also send messages to user screens. For example, a message could notify users that their passwords must be changed because of the company's password aging policy.

- Authorization—Provides fine-grained control over user capabilities for the duration of the user's session, including but not limited to setting autocommands, access control, session duration, or protocol support. You can also enforce restrictions on what commands a user can execute with the TACACS+ authorization feature.

- Accounting—Collects and sends information used for billing, auditing, and reporting to the TACACS+ daemon. Network managers can use the accounting facility to track user activity for a security audit or to provide information for user billing. Accounting records include user identities, start and stop times, executed commands (such as PPP), number of packets, and number of bytes.

The TACACS+ protocol provides authentication between the switch and the TACACS+ daemon, and it ensures confidentiality because all protocol exchanges between the switch and the TACACS+ daemon are encrypted.

You need a system running the TACACS+ daemon software to use TACACS+ on your switch.

TACACS+ Operation

When a user attempts a simple ASCII login by authenticating to a switch by using TACACS+, this process occurs:

1. When the connection is established, the switch contacts the TACACS+ daemon to obtain a username prompt to show to the user. The user enters a username, and the switch then contacts the TACACS+ daemon to obtain a password prompt. The switch displays the password prompt to the user, the user enters a password, and the password is then sent to the TACACS+ daemon.

   TACACS+ allows a dialog between the daemon and the user until the daemon receives enough information to authenticate the user. The daemon prompts for a username and password combination, but can include other items, such as the user's mother's maiden name.

2. The switch eventually receives one of these responses from the TACACS+ daemon:
   - ACCEPT—The user is authenticated and service can begin. If the switch is configured to require authorization, authorization begins at this time.
   - REJECT—The user is not authenticated. The user can be denied access or is prompted to retry the login sequence, depending on the TACACS+ daemon.
   - ERROR—An error occurred at some time during authentication with the daemon or in the network connection between the daemon and the switch. If an ERROR response is received, the switch typically tries to use an alternative method for authenticating the user.
   - CONTINUE—The user is prompted for additional authentication information.

   After authentication, the user undergoes an additional authorization phase if authorization has been enabled on the switch. Users must first successfully complete TACACS+ authentication before proceeding to TACACS+ authorization.

3. If TACACS+ authorization is required, the TACACS+ daemon is again contacted, and it returns an ACCEPT or REJECT authorization response. If an ACCEPT response is returned, the response contains data in the form of attributes that direct the EXEC or NETWORK session for that user, determining the services that the user can access:
   - Telnet, Secure Shell (SSH), rlogin, or privileged EXEC services
   - Connection parameters, including the host or client IP address, access list, and user timeouts

Configuring TACACS+

This section describes how to configure your switch to support TACACS+. At a minimum, you must identify the host or hosts maintaining the TACACS+ daemon and define the method lists for TACACS+ authentication. You can optionally define method lists for TACACS+ authorization and accounting. A method list defines the sequence and methods to be used to authenticate, to authorize, or to keep accounts on a user. You can use method lists to designate one or more security protocols to be used, thus ensuring a backup system if the initial method fails. The software uses the first method listed to authenticate, to authorize, or to keep accounts on users; if that method does not respond, the software selects the next method in the list. This process continues until there is successful communication with a listed method or the method list is exhausted.

This section contains this configuration information:

- Default TACACS+ Configuration, page 7-13
- Identifying the TACACS+ Server Host and Setting the Authentication Key, page 7-13
- Configuring TACACS+ Login Authentication, page 7-14

- Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services, page 7-16
- Starting TACACS+ Accounting, page 7-17

Default TACACS+ Configuration

TACACS+ and AAA are disabled by default.

To prevent a lapse in security, you cannot configure TACACS+ through a network management application.When enabled, TACACS+ can authenticate users accessing the switch through the CLI.

Note Although TACACS+ configuration is performed through the CLI, the TACACS+ server authenticates HTTP connections that have been configured with a privilege level of 15.

Identifying the TACACS+ Server Host and Setting the Authentication Key

You can configure the switch to use a single server or AAA server groups to group existing server hosts for authentication. You can group servers to select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list and contains the list of IP addresses of the selected server hosts.

Beginning in privileged EXEC mode, follow these steps to identify the IP host or host maintaining TACACS+ server and optionally set the encryption key:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**tacacs-server host** *hostname* [**port** *integer*] [**timeout** *integer*] [**key** *string*]	Identify the IP host or hosts maintaining a TACACS+ server. Enter this command multiple times to create a list of preferred hosts. The software searches for hosts in the order in which you specify them. • For *hostname*, specify the name or IP address of the host. • (Optional) For **port** *integer*, specify a server port number. The default is port 49. The range is 1 to 65535. • (Optional) For **timeout** *integer*, specify a time in seconds the switch waits for a response from the daemon before it times out and declares an error. The default is 5 seconds. The range is 1 to 1000 seconds. • (Optional) For **key** *string*, specify the encryption key for encrypting and decrypting all traffic between the switch and the TACACS+ daemon. You must configure the same key on the TACACS+ daemon for encryption to be successful.
Step 3	**aaa new-model**	Enable AAA.
Step 4	**aaa group server tacacs+** *group-name*	(Optional) Define the AAA server-group with a group name. This command puts the switch in a server group subconfiguration mode.
Step 5	**server** *ip-address*	(Optional) Associate a particular TACACS+ server with the defined server group. Repeat this step for each TACACS+ server in the AAA server group. Each server in the group must be previously defined in Step 2.

	Command	Purpose
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show tacacs**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the specified TACACS+ server name or address, use the **no tacacs-server host** *hostname* global configuration command. To remove a server group from the configuration list, use the **no aaa group server tacacs+** *group-name* global configuration command. To remove the IP address of a TACACS+ server, use the **no server ip-address** server group subconfiguration command.

Configuring TACACS+ Login Authentication

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various interfaces. The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific interface before any of the defined authentication methods are performed. The only exception is the default method list (which, by coincidence, is named *default*). The default method list is automatically applied to all interfaces except those that have a named method list explicitly defined. A defined method list overrides the default method list.

A method list describes the sequence and authentication methods to be queried to authenticate a user. You can designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. The software uses the first method listed to authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the authentication process stops, and no other authentication methods are attempted.

Beginning in privileged EXEC mode, follow these steps to configure login authentication:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA.

	Command	Purpose
Step 3	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]	Create a login authentication method list. • To create a default list that is used when a named list is *not* specified in the **login authentication** command, use the **default** keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all interfaces. • For *list-name*, specify a character string to name the list you are creating. • For *method1...*, specify the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. Select one of these methods: • **enable**—Use the enable password for authentication. Before you can use this authentication method, you must define an enable password by using the **enable** *password* global configuration command. • **group tacacs+**—Uses TACACS+ authentication. Before you can use this authentication method, you must configure the TACACS+ server. For more information, see the "Identifying the TACACS+ Server Host and Setting the Authentication Key" section on page 7-13. • **line**—Use the line password for authentication. Before you can use this authentication method, you must define a line password. Use the **password** *password* line configuration command. • **local**—Use the local username database for authentication. You must enter username information in the database. Use the **username** *password* global configuration command. • **local-case**—Use a case-sensitive local username database for authentication. You must enter username information in the database by using the **username** *name* **password** global configuration command. • **none**—Do not use any authentication for login.
Step 4	**line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*]	Enter line configuration mode, and configure the lines to which you want to apply the authentication list.
Step 5	**login authentication** {**default** | *list-name*}	Apply the authentication list to a line or set of lines. • If you specify **default**, use the default list created with the **aaa authentication login** command. • For *list-name*, specify the list created with the **aaa authentication login** command.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable AAA, use the **no aaa new-model** global configuration command. To disable AAA authentication, use the **no aaa authentication login** {**default** | *list-name*} *method1* [*method2...*] global configuration command. To either disable TACACS+ authentication for logins or to return to the default value, use the **no login authentication** {**default** | *list-name*} line configuration command.

Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services

AAA authorization limits the services available to a user. When AAA authorization is enabled, the switch uses information retrieved from the user's profile, which is located either in the local user database or on the security server, to configure the user's session. The user is granted access to a requested service only if the information in the user profile allows it.

You can use the **aaa authorization** global configuration command with the **tacacs+** keyword to set parameters that restrict a user's network access to privileged EXEC mode.

The **aaa authorization exec tacacs+ local** command sets these authorization parameters:

- Use TACACS+ for privileged EXEC access authorization if authentication was performed by using TACACS+.
- Use the local database if authentication was not performed by using TACACS+.

Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

Beginning in privileged EXEC mode, follow these steps to specify TACACS+ authorization for privileged EXEC access and network services:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa authorization network tacacs+**	Configure the switch for user TACACS+ authorization for all network-related service requests.
Step 3	**aaa authorization exec tacacs+**	Configure the switch for user TACACS+ authorization to determine if the user has privileged EXEC access. The **exec** keyword might return user profile information (such as **autocommand** information).
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable authorization, use the **no aaa authorization** {**network** | **exec**} *method1* global configuration command.

Starting TACACS+ Accounting

The AAA accounting feature tracks the services that users are accessing and the amount of network resources that they are consuming. When AAA accounting is enabled, the switch reports user activity to the TACACS+ security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. This data can then be analyzed for network management, client billing, or auditing.

Beginning in privileged EXEC mode, follow these steps to enable TACACS+ accounting for each privilege level and for network services:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa accounting network start-stop tacacs+**	Enable TACACS+ accounting for all network-related service requests.
Step 3	**aaa accounting exec start-stop tacacs+**	Enable TACACS+ accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable accounting, use the **no aaa accounting** {**network** | **exec**} {**start-stop**} *method1...* global configuration command.

Displaying the TACACS+ Configuration

To display TACACS+ server statistics, use the **show tacacs** privileged EXEC command.

Controlling Switch Access with RADIUS

This section describes how to enable and configure the RADIUS, which provides detailed accounting information and flexible administrative control over authentication and authorization processes. RADIUS is facilitated through AAA and can be enabled only through AAA commands.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Security Command Reference for Cisco IOS Release 12.2*.

This section contains this configuration information:

- Understanding RADIUS, page 7-18
- RADIUS Operation, page 7-19
- Configuring RADIUS, page 7-19
- Displaying the RADIUS Configuration, page 7-31

Understanding RADIUS

RADIUS is a distributed client/server system that secures networks against unauthorized access. RADIUS clients run on supported Cisco routers and switches. Clients send authentication requests to a central RADIUS server, which contains all user authentication and network service access information. The RADIUS host is normally a multiuser system running RADIUS server software from Cisco (Cisco Secure Access Control Server version 3.0), Livingston, Merit, Microsoft, or another software provider. For more information, see the RADIUS server documentation.

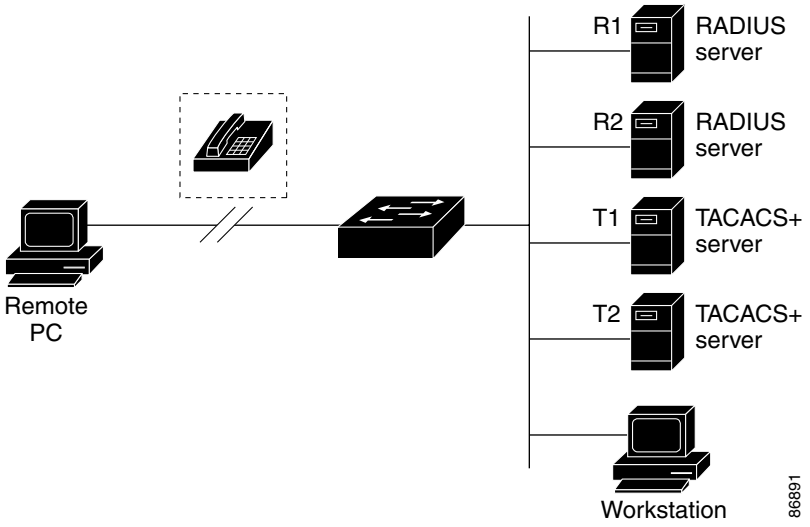
Use RADIUS in these network environments that require access security:

- Networks with multiple-vendor access servers, each supporting RADIUS. For example, access servers from several vendors use a single RADIUS server-based security database. In an IP-based network with multiple vendors' access servers, dial-in users are authenticated through a RADIUS server that has been customized to work with the Kerberos security system.
- Turnkey network security environments in which applications support the RADIUS protocol, such as in an access environment that uses a *smart card* access control system. In one case, RADIUS has been used with Enigma's security cards to validates users and to grant access to network resources.
- Networks already using RADIUS. You can add a Cisco switch containing a RADIUS client to the network. This might be the first step when you make a transition to a TACACS+ server. See Figure 7-2 on page 7-19.
- Network in which the user must only access a single service. Using RADIUS, you can control user access to a single host, to a single utility such as Telnet, or to the network through a protocol such as IEEE 802.1X. For more information about this protocol, see Chapter 8, "Configuring IEEE 802.1x Port-Based Authentication."
- Networks that require resource accounting. You can use RADIUS accounting independently of RADIUS authentication or authorization. The RADIUS accounting functions allow data to be sent at the start and end of services, showing the amount of resources (such as time, packets, bytes, and so forth) used during the session. An Internet service provider might use a freeware-based version of RADIUS access control and accounting software to meet special security and billing needs.

RADIUS is not suitable in these network security situations:

- Multiprotocol access environments. RADIUS does not support AppleTalk Remote Access (ARA), NetBIOS Frame Control Protocol (NBFCP), NetWare Asynchronous Services Interface (NASI), or X.25 PAD connections.
- Switch-to-switch or router-to-router situations. RADIUS does not provide two-way authentication. RADIUS can be used to authenticate from one device to a non-Cisco device if the non-Cisco device requires authentication.
- Networks using a variety of services. RADIUS generally binds a user to one service model.

Figure 7-2 Transitioning from RADIUS to TACACS+ Services



RADIUS Operation

When a user attempts to log in and authenticate to a switch that is access controlled by a RADIUS server, these events occur:

1. The user is prompted to enter a username and password.
2. The username and encrypted password are sent over the network to the RADIUS server.
3. The user receives one of these responses from the RADIUS server:
   a. ACCEPT—The user is authenticated.
   b. REJECT—The user is either not authenticated and is prompted to re-enter the username and password, or access is denied.
   c. CHALLENGE—A challenge requires additional data from the user.
   d. CHALLENGE PASSWORD—A response requests the user to select a new password.

The ACCEPT or REJECT response is bundled with additional data that is used for privileged EXEC or network authorization. Users must first successfully complete RADIUS authentication before proceeding to RADIUS authorization, if it is enabled. The additional data included with the ACCEPT or REJECT packets includes these items:

- Telnet, SSH, rlogin, or privileged EXEC services
- Connection parameters, including the host or client IP address, access list, and user timeouts

Configuring RADIUS

This section describes how to configure your switch to support RADIUS. At a minimum, you must identify the host or hosts that run the RADIUS server software and define the method lists for RADIUS authentication. You can optionally define method lists for RADIUS authorization and accounting.

A method list defines the sequence and methods to be used to authenticate, to authorize, or to keep accounts on a user. You can use method lists to designate one or more security protocols to be used (such as TACACS+ or local username lookup), thus ensuring a backup system if the initial method fails. The

software uses the first method listed to authenticate, to authorize, or to keep accounts on users; if that method does not respond, the software selects the next method in the list. This process continues until there is successful communication with a listed method or the method list is exhausted.

You should have access to and should configure a RADIUS server before configuring RADIUS features on your switch.

This section contains this configuration information:

- Default RADIUS Configuration, page 7-20
- Identifying the RADIUS Server Host, page 7-20 (required)
- Configuring RADIUS Login Authentication, page 7-23 (required)
- Defining AAA Server Groups, page 7-25 (optional)
- Configuring RADIUS Authorization for User Privileged Access and Network Services, page 7-27 (optional)
- Starting RADIUS Accounting, page 7-28 (optional)
- Configuring Settings for All RADIUS Servers, page 7-29 (optional)
- Configuring the Switch to Use Vendor-Specific RADIUS Attributes, page 7-29 (optional)
- Configuring the Switch for Vendor-Proprietary RADIUS Server Communication, page 7-31 (optional)

Default RADIUS Configuration

RADIUS and AAA are disabled by default.

To prevent a lapse in security, you cannot configure RADIUS through a network management application. When enabled, RADIUS can authenticate users accessing the switch through the CLI.

Identifying the RADIUS Server Host

Switch-to-RADIUS-server communication involves several components:

- Host name or IP address
- Authentication destination port
- Accounting destination port
- Key string
- Timeout period
- Retransmission value

You identify RADIUS security servers by their host name or IP address, host name and specific UDP port numbers, or their IP address and specific UDP port numbers. The combination of the IP address and the UDP port number creates a unique identifier, allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. This unique identifier enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address.

If two different host entries on the same RADIUS server are configured for the same service—for example, accounting—the second host entry configured acts as a fail-over backup to the first one. Using this example, if the first host entry fails to provide accounting services, the switch tries the second host entry configured on the same device for accounting services. (The RADIUS host entries are tried in the order that they are configured.)

A RADIUS server and the switch use a shared secret text string to encrypt passwords and exchange responses. To configure RADIUS to use the AAA security commands, you must specify the host running the RADIUS server daemon and a secret text (key) string that it shares with the switch.

The timeout, retransmission, and encryption key values can be configured globally for all RADIUS servers, on a per-server basis, or in some combination of global and per-server settings. To apply these settings globally to all RADIUS servers communicating with the switch, use the three unique global configuration commands: **radius-server timeout**, **radius-server retransmit**, and **radius-server key**. To apply these values on a specific RADIUS server, use the **radius-server host** global configuration command.

Note If you configure both global and per-server functions (timeout, retransmission, and key commands) on the switch, the per-server timer, retransmission, and key value commands override global timer, retransmission, and key value commands. For information on configuring these settings on all RADIUS servers, see the "Configuring Settings for All RADIUS Servers" section on page 7-29.

You can configure the switch to use AAA server groups to group existing server hosts for authentication. For more information, see the "Defining AAA Server Groups" section on page 7-25.

Beginning in privileged EXEC mode, follow these steps to configure per-server RADIUS server communication. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]	Specify the IP address or host name of the remote RADIUS server host. • (Optional) For **auth-port** *port-number*, specify the UDP destination port for authentication requests. • (Optional) For **acct-port** *port-number*, specify the UDP destination port for accounting requests. • (Optional) For **timeout** *seconds*, specify the time interval that the switch waits for the RADIUS server to reply before resending. The range is 1 to 1000. This setting overrides the **radius-server timeout** global configuration command setting. If no timeout is set with the **radius-server host** command, the setting of the **radius-server timeout** command is used. • (Optional) For **retransmit** *retries*, specify the number of times a RADIUS request is resent to a server if that server is not responding or responding slowly. The range is 1 to 1000. If no retransmit value is set with the **radius-server host** command, the setting of the **radius-server retransmit** global configuration command is used. • (Optional) For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the **radius-server host** command. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key. To configure the switch to recognize more than one host entry associated with a single IP address, enter this command as many times as necessary, making sure that each UDP port number is different. The switch software searches for hosts in the order in which you specify them. Set the timeout, retransmit, and encryption key values to use with the specific RADIUS host.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the specified RADIUS server, use the **no radius-server host** *hostname* | *ip-address* global configuration command.

This example shows how to configure one RADIUS server to be used for authentication and another to be used for accounting:

```
Switch(config)# radius-server host 172.29.36.49 auth-port 1612 key rad1
Switch(config)# radius-server host 172.20.36.50 acct-port 1618 key rad2
```

This example shows how to configure *host1* as the RADIUS server and to use the default ports for both authentication and accounting:

```
Switch(config)# radius-server host host1
```

Note You also need to configure some settings on the RADIUS server. These settings include the IP address of the switch and the key string to be shared by both the server and the switch. For more information, see the RADIUS server documentation.

Configuring RADIUS Login Authentication

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various interfaces. The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific interface before any of the defined authentication methods are performed. The only exception is the default method list (which, by coincidence, is named *default*). The default method list is automatically applied to all interfaces except those that have a named method list explicitly defined.

A method list describes the sequence and authentication methods to be queried to authenticate a user. You can designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. The software uses the first method listed to authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the authentication process stops, and no other authentication methods are attempted.

Beginning in privileged EXEC mode, follow these steps to configure login authentication. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA.

	Command	Purpose
Step 3	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]	Create a login authentication method list. • To create a default list that is used when a named list is *not* specified in the **login authentication** command, use the **default** keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all interfaces. • For *list-name*, specify a character string to name the list you are creating. • For *method1...*, specify the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. Select one of these methods: – **enable**—Use the enable password for authentication. Before you can use this authentication method, you must define an enable password by using the **enable** *password* global configuration command. – **group radius**—Use RADIUS authentication. Before you can use this authentication method, you must configure the RADIUS server. For more information, see the "Identifying the RADIUS Server Host" section on page 7-20. – **line**—Use the line password for authentication. Before you can use this authentication method, you must define a line password. Use the **password** *password* line configuration command. – **local**—Use the local username database for authentication. You must enter username information in the database. Use the **username** *name* **password** global configuration command. – **local-case**—Use a case-sensitive local username database for authentication. You must enter username information in the database by using the **username** *password* global configuration command. – **none**—Do not use any authentication for login.
Step 4	**line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*]	Enter line configuration mode, and configure the lines to which you want to apply the authentication list.
Step 5	**login authentication** {**default** | *list-name*}	Apply the authentication list to a line or set of lines. • If you specify **default**, use the default list created with the **aaa authentication login** command. • For *list-name*, specify the list created with the **aaa authentication login** command.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable AAA, use the **no aaa new-model** global configuration command. To disable AAA authentication, use the **no aaa authentication login** {**default** | *list-name*} *method1* [*method2...*] global configuration command. To either disable RADIUS authentication for logins or to return to the default value, use the **no login authentication** {**default** | *list-name*} line configuration command.

Defining AAA Server Groups

You can configure the switch to use AAA server groups to group existing server hosts for authentication. You select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list, which lists the IP addresses of the selected server hosts.

Server groups also can include multiple host entries for the same server if each entry has a unique identifier (the combination of the IP address and UDP port number), allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. If you configure two different host entries on the same RADIUS server for the same service, (for example, accounting), the second configured host entry acts as a fail-over backup to the first one.

You use the **server** group server configuration command to associate a particular server with a defined group server. You can either identify the server by its IP address or identify multiple host instances or entries by using the optional **auth-port** and **acct-port** keywords.

Beginning in privileged EXEC mode, follow these steps to define the AAA server group and associate a particular RADIUS server with it:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]	Specify the IP address or host name of the remote RADIUS server host. • (Optional) For **auth-port** *port-number*, specify the UDP destination port for authentication requests. • (Optional) For **acct-port** *port-number*, specify the UDP destination port for accounting requests. • (Optional) For **timeout** *seconds*, specify the time interval that the switch waits for the RADIUS server to reply before resending. The range is 1 to 1000. This setting overrides the **radius-server timeout** global configuration command setting. If no timeout is set with the **radius-server host** command, the setting of the **radius-server timeout** command is used. • (Optional) For **retransmit** *retries*, specify the number of times a RADIUS request is resent to a server if that server is not responding or responding slowly. The range is 1 to 1000. If no retransmit value is set with the **radius-server host** command, the setting of the **radius-server retransmit** global configuration command is used. • (Optional) For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the **radius-server host** command. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key. To configure the switch to recognize more than one host entry associated with a single IP address, enter this command as many times as necessary, making sure that each UDP port number is different. The switch software searches for hosts in the order in which you specify them. Set the timeout, retransmit, and encryption key values to use with the specific RADIUS host.
Step 3	**aaa new-model**	Enable AAA.
Step 4	**aaa group server radius** *group-name*	Define the AAA server-group with a group name. This command puts the switch in a server group configuration mode.
Step 5	**server** *ip-address*	Associate a particular RADIUS server with the defined server group. Repeat this step for each RADIUS server in the AAA server group. Each server in the group must be previously defined in Step 2.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries.

	Command	Purpose
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.
Step 9		Enable RADIUS login authentication. See the "Configuring RADIUS Login Authentication" section on page 7-23.

To remove the specified RADIUS server, use the **no radius-server host** *hostname* | *ip-address* global configuration command. To remove a server group from the configuration list, use the **no aaa group server radius** *group-name* global configuration command. To remove the IP address of a RADIUS server, use the **no server** *ip-address* server group configuration command.

In this example, the switch is configured to recognize two different RADIUS group servers (*group1* and *group2*). Group1 has two different host entries on the same RADIUS server configured for the same services. The second host entry acts as a fail-over backup to the first entry.

```
Switch(config)# radius-server host 172.20.0.1 auth-port 1000 acct-port 1001
Switch(config)# radius-server host 172.10.0.1 auth-port 1645 acct-port 1646
Switch(config)# aaa new-model
Switch(config)# aaa group server radius group1
Switch(config-sg-radius)# server 172.20.0.1 auth-port 1000 acct-port 1001
Switch(config-sg-radius)# exit
Switch(config)# aaa group server radius group2
Switch(config-sg-radius)# server 172.20.0.1 auth-port 2000 acct-port 2001
Switch(config-sg-radius)# exit
```

Configuring RADIUS Authorization for User Privileged Access and Network Services

AAA authorization limits the services available to a user. When AAA authorization is enabled, the switch uses information retrieved from the user's profile, which is in the local user database or on the security server, to configure the user's session. The user is granted access to a requested service only if the information in the user profile allows it.

You can use the **aaa authorization** global configuration command with the **radius** keyword to set parameters that restrict a user's network access to privileged EXEC mode.

The **aaa authorization exec radius local** command sets these authorization parameters:

- Use RADIUS for privileged EXEC access authorization if authentication was performed by using RADIUS.
- Use the local database if authentication was not performed by using RADIUS.

Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

Beginning in privileged EXEC mode, follow these steps to specify RADIUS authorization for privileged EXEC access and network services:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa authorization network radius**	Configure the switch for user RADIUS authorization for all network-related service requests.

	Command	Purpose
Step 3	**aaa authorization exec radius**	Configure the switch for user RADIUS authorization to determine if the user has privileged EXEC access. The **exec** keyword might return user profile information (such as **autocommand** information).
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable authorization, use the **no aaa authorization** {**network** | **exec**} *method1* global configuration command.

Starting RADIUS Accounting

The AAA accounting feature tracks the services that users are accessing and the amount of network resources that they are consuming. When AAA accounting is enabled, the switch reports user activity to the RADIUS security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. This data can then be analyzed for network management, client billing, or auditing.

Beginning in privileged EXEC mode, follow these steps to enable RADIUS accounting for each Cisco IOS privilege level and for network services:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa accounting network start-stop radius**	Enable RADIUS accounting for all network-related service requests.
Step 3	**aaa accounting exec start-stop radius**	Enable RADIUS accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable accounting, use the **no aaa accounting** {**network** | **exec**} {**start-stop**} *method1*... global configuration command.

Configuring Settings for All RADIUS Servers

Beginning in privileged EXEC mode, follow these steps to configure global communication settings between the switch and all RADIUS servers:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server key** *string*	Specify the shared secret text string used between the switch and all RADIUS servers. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
Step 3	**radius-server retransmit** *retries*	Specify the number of times the switch sends each RADIUS request to the server before giving up. The default is 3; the range 1 to 1000.
Step 4	**radius-server timeout** *seconds*	Specify the number of seconds a switch waits for a reply to a RADIUS request before resending the request. The default is 5 seconds; the range is 1 to 1000.
Step 5	**radius-server deadtime** *minutes*	Specify the number of minutes a RADIUS server, which is not responding to authentication requests, to be skipped, thus avoiding the wait for the request to timeout before trying the next configured server. The default is 0; the range is 1 to 1440 minutes.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your settings.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting for the retransmit, timeout, and deadtime, use the **no** forms of these commands.

Configuring the Switch to Use Vendor-Specific RADIUS Attributes

The Internet Engineering Task Force (IETF) draft standard specifies a method for communicating vendor-specific information between the switch and the RADIUS server by using the vendor-specific attribute (attribute 26). Vendor-specific attributes (VSAs) allow vendors to support their own extended attributes not suitable for general use. The Cisco RADIUS implementation supports one vendor-specific option by using the format recommended in the specification. Cisco's vendor-ID is 9, and the supported option has vendor-type 1, which is named *cisco-avpair*. The value is a string with this format:

```
protocol : attribute sep value *
```

Protocol is a value of the Cisco protocol attribute for a particular type of authorization. *Attribute* and *value* are an appropriate attribute-value (AV) pair defined in the Cisco TACACS+ specification, and *sep* is = for mandatory attributes and is * for optional attributes. The full set of features available for TACACS+ authorization can then be used for RADIUS.

For example, this AV pair activates Cisco's *multiple named ip address pools* feature during IP authorization (during PPP's IPCP address assignment):

```
cisco-avpair= "ip:addr-pool=first"
```

This example shows how to provide a user logging in from a switch with immediate access to privileged EXEC commands:

```
cisco-avpair= "shell:priv-lvl=15"
```

This example shows how to specify an authorized VLAN in the RADIUS server database:

```
cisco-avpair= "tunnel-type(#64)=VLAN(13)"
cisco-avpair= "tunnel-medium-type(#65)=802 media(6)"
cisco-avpair= "tunnel-private-group-ID(#81)=vlanid"
```

This example shows how to apply an input ACL, in ASCII format, to an interface for the duration of this connection:

```
cisco-avpair= "ip:inacl#1=deny ip 10.10.10.10 0.0.255.255 20.20.20.20 255.255.0.0"
cisco-avpair= "ip:inacl#2=deny ip 10.10.10.10 0.0.255.255 any"
cisco-avpair= "mac:inacl#3=deny any any decnet-iv"
```

This example shows how to apply an output ACL, in ASCII format, to an interface for the duration of this connection:

```
cisco-avpair= "ip:outacl#2=deny ip 10.10.10.10 0.0.255.255 any"
```

Other vendors have their own unique vendor-IDs, options, and associated VSAs. For more information about vendor-IDs and VSAs, see RFC 2138, "Remote Authentication Dial-In User Service (RADIUS)."

Beginning in privileged EXEC mode, follow these steps to configure the switch to recognize and use VSAs:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server vsa send** [**accounting** | **authentication**]	Enable the switch to recognize and use VSAs as defined by RADIUS IETF attribute 26. • (Optional) Use the **accounting** keyword to limit the set of recognized vendor-specific attributes to only accounting attributes. • (Optional) Use the **authentication** keyword to limit the set of recognized vendor-specific attributes to only authentication attributes. If you enter this command without keywords, both accounting and authentication vendor-specific attributes are used.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your settings.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

For a complete list of RADIUS attributes or more information about vendor-specific attribute 26, see the "RADIUS Attributes" appendix in the *Cisco IOS Security Configuration Guide for Cisco IOS Release 12.2*.

Configuring the Switch for Vendor-Proprietary RADIUS Server Communication

Although an IETF draft standard for RADIUS specifies a method for communicating vendor-proprietary information between the switch and the RADIUS server, some vendors have extended the RADIUS attribute set in a unique way. Cisco IOS software supports a subset of vendor-proprietary RADIUS attributes.

As mentioned earlier, to configure RADIUS (whether vendor-proprietary or IETF draft-compliant), you must specify the host running the RADIUS server daemon and the secret text string it shares with the switch. You specify the RADIUS host and secret text string by using the **radius-server** global configuration commands.

Beginning in privileged EXEC mode, follow these steps to specify a vendor-proprietary RADIUS server host and a shared secret text string:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server host** {*hostname* | *ip-address*} **non-standard**	Specify the IP address or host name of the remote RADIUS server host and identify that it is using a vendor-proprietary implementation of RADIUS.
Step 3	**radius-server key** *string*	Specify the shared secret text string used between the switch and the vendor-proprietary RADIUS server. The switch and the RADIUS server use this text string to encrypt passwords and exchange responses. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your settings.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete the vendor-proprietary RADIUS host, use the **no radius-server host** {*hostname* | *ip-address*} **non-standard** global configuration command. To disable the key, use the **no radius-server key** global configuration command.

This example shows how to specify a vendor-proprietary RADIUS host and to use a secret key of *rad124* between the switch and the server:

```
Switch(config)# radius-server host 172.20.30.15 nonstandard
Switch(config)# radius-server key rad124
```

Displaying the RADIUS Configuration

To display the RADIUS configuration, use the **show running-config** privileged EXEC command.

Controlling Switch Access with Kerberos

This section describes how to enable and configure the Kerberos security system, which authenticates requests for network resources by using a trusted third party. To use this feature, the cryptographic (encrypted) multilayer software image must be installed on your switch. You must obtain authorization to use this feature and to download the cryptographic software files from Cisco.com.For more information, see the release notes for this release.

This section consists of these topics:

- Understanding Kerberos, page 7-32
- Kerberos Operation, page 7-34
- Configuring Kerberos, page 7-35

For Kerberos configuration examples, see the "Kerberos Configuration Examples" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Configuration Guide, Release 12.2,* at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fsecsp/

Note For complete syntax and usage information for the commands used in this section, see the "Kerberos Commands" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Command Reference, Release 12.2,* at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fsecsp/index.htm.

Note In the Kerberos configuration examples and in the *Cisco IOS Security Command Reference, Release 12.2,* the trusted third party can be a Catalyst 3550 switch that supports Kerberos, that is configured as a network security server, and that can authenticate users by using the Kerberos protocol.

Understanding Kerberos

Kerberos is a secret-key network authentication protocol, which was developed at the Massachusetts Institute of Technology (MIT). It uses the Data Encryption Standard (DES) cryptographic algorithm for encryption and authentication and authenticates requests for network resources. Kerberos uses the concept of a trusted third party to perform secure verification of users and services. This trusted third party is called the *key distribution center* (KDC).

The main purpose of Kerberos is to verify that users are who they claim to be and the network services that they use are what the services claim to be. To do this, a KDC or trusted Kerberos server issues tickets to users. These tickets, which have a limited lifespan, are stored in user credential caches. The Kerberos server uses the tickets instead of usernames and passwords to authenticate users and network services.

Note A Kerberos server can be a Catalyst 3550 switch that is configured as a network security server and that can authenticate users by using the Kerberos protocol.

The Kerberos credential scheme uses a process called *single logon*. This process authenticates a user once and then allows secure authentication (without encrypting another password) wherever that user credential is accepted.

This software release supports Kerberos 5, which allows organizations that are already using Kerberos 5 to use the same Kerberos authentication database on the KDC that they are already using on their other network hosts (such as UNIX servers and PCs).

In this software release, Kerberos supports these network services:

- Telnet
- rlogin
- rsh (Remote Shell Protocol)

Table 7-2 lists the common Kerberos-related terms and definitions:

Table 7-2 Kerberos Terms

Term	Definition
Authentication	A process by which a user or service identifies itself to another service. For example, a client can authenticate to a switch or a switch can authenticate to another switch.
Authorization	A means by which the switch determines what privileges the user has in a network or on the switch and what actions the user can perform.
Credential	A general term that refers to authentication tickets, such as TGTs[1] and service credentials. Kerberos credentials verify the identity of a user or service. If a network service decides to trust the Kerberos server that issued a ticket, it can be used in place of re-entering a username and password. Credentials have a default lifespan of eight hours.
Instance	An authorization level label for Kerberos principals. Most Kerberos principals are of the form *user@REALM* (for example, smith@EXAMPLE.COM). A Kerberos principal with a Kerberos instance has the form *user/instance@REALM* (for example, smith/admin@EXAMPLE.COM). The Kerberos instance can be used to specify the authorization level for the user if authentication is successful. The server of each network service might implement and enforce the authorization mappings of Kerberos instances but is not required to do so. **Note** The Kerberos principal and instance names *must* be in all lowercase characters. **Note** The Kerberos realm name *must* be in all uppercase characters.
KDC[2]	Key distribution center that consists of a Kerberos server and database program that is running on a network host.
Kerberized	A term that describes applications and services that have been modified to support the Kerberos credential infrastructure.
Kerberos realm	A domain consisting of users, hosts, and network services that are registered to a Kerberos server. The Kerberos server is trusted to verify the identity of a user or network service to another user or network service. **Note** The Kerberos realm name *must* be in all uppercase characters.
Kerberos server	A daemon that is running on a network host. Users and network services register their identity with the Kerberos server. Network services query the Kerberos server to authenticate to other network services.

Table 7-2 Kerberos Terms (continued)

Term	Definition
KEYTAB[3]	A password that a network service shares with the KDC. In Kerberos 5 and later Kerberos versions, the network service authenticates an encrypted service credential by using the KEYTAB to decrypt it. In Kerberos versions earlier than Kerberos 5, KEYTAB is referred to as SRVTAB[4].
Principal	Also known as a Kerberos identity, this is who you are or what a service is according to the Kerberos server. **Note** The Kerberos principal name *must* be in all lowercase characters.
Service credential	A credential for a network service. When issued from the KDC, this credential is encrypted with the password shared by the network service and the KDC. The password is also shared with the user TGT.
SRVTAB	A password that a network service shares with the KDC. In Kerberos 5 or later Kerberos versions, SRVTAB is referred to as KEYTAB.
TGT	Ticket granting ticket that is a credential that the KDC issues to authenticated users. When users receive a TGT, they can authenticate to network services within the Kerberos realm represented by the KDC.

1. TGT = ticket granting ticket
2. KDC = key distribution center
3. KEYTAB = key table
4. SRVTAB = server table

Kerberos Operation

This section describes how Kerberos operates with a switch that is configured as a network security server. Although you can customize Kerberos in a number of ways, remote users attempting to access network services must pass through three layers of security before they can access network services.

To authenticate to network services by using a switch as a Kerberos server, remote users must follow these steps:

1. Authenticating to a Boundary Switch, page 7-34
2. Obtaining a TGT from a KDC, page 7-35
3. Authenticating to Network Services, page 7-35

Note A Kerberos server can be a switch that is configured as a network security server and that can authenticate remote users by using the Kerberos protocol.

Authenticating to a Boundary Switch

This section describes the first layer of security through which a remote user must pass. The user must first authenticate to the boundary switch. When the remote user authenticates to a boundary switch, this process occurs:

1. The user opens an un-Kerberized Telnet connection to the boundary switch.
2. The switch prompts the user for a username and password.

3. The switch requests a TGT from the KDC for this user.
4. The KDC sends an encrypted TGT to the switch that includes the user identity.
5. The switch attempts to decrypt the TGT by using the password that the user entered.
   - If the decryption is successful, the user is authenticated to the switch.
   - If the decryption is not successful, the user repeats Step 2 by re-entering the username and password (noting if Caps Lock or Num Lock is on or off) or by entering a different usernname and password.

A remote user who initiates a un-Kerberized Telnet session and authenticates to a boundary switch is inside the firewall, but the user must still authenticate directly to the KDC before getting access to the network services. The user must authenticate to the KDC because the TGT that the KDC issues is stored on the switch and cannot be used for additional authentication until the user logs on to the switch.

Obtaining a TGT from a KDC

This section describes the second layer of security through which a remote user must pass. The user must now authenticate to a KDC and obtain a TGT from the KDC to access network services.

For instructions about how to authenticate to a KDC, see the "Obtaining a TGT from a KDC" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Configuration Guide, Release 12.2,* at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fsecsp/scfkerb.htm#1000999.

Authenticating to Network Services

This section describes the third layer of security through which a remote user must pass. The user with a TGT must now authenticate to the network services in a Kerberos realm.

For instructions about how to authenticate to a network service, see the "Authenticating to Network Services" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Configuration Guide, Release 12.2,* at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fsecsp/scfkerb.htm#1001010.

Configuring Kerberos

So that remote users can authenticate to network services, you must configure the hosts and the KDC in the Kerberos realm to communicate and mutually authenticate users and network services. To do this, you must identify them to each other. You add entries for the hosts to the Kerberos database on the KDC and add KEYTAB files generated by the KDC to all hosts in the Kerberos realm. You also create entries for the users in the KDC database.

When you add or create entries for the hosts and users, follow these guidelines:

- The Kerberos principal name *must* be in all lowercase characters.
- The Kerberos instance name *must* be in all lowercase characters.
- The Kerberos realm name *must* be in all uppercase characters.

Note A Kerberos server can be a switch that is configured as a network security server and that can authenticate users by using the Kerberos protocol.

To set up a Kerberos-authenticated server-client system, follow these steps:

- Configure the KDC by using Kerberos commands.
- Configure the switch to use the Kerberos protocol.

For instructions, see the "Kerberos Configuration Task List" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Configuration Guide, Release 12.2,* at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fsecsp/scfkerb.htm#1001027.

Configuring the Switch for Local Authentication and Authorization

You can configure AAA to operate without a server by setting the switch to implement AAA in local mode. The switch then handles authentication and authorization. No accounting is available in this configuration.

Beginning in privileged EXEC mode, follow these steps to configure the switch for local AAA:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA.
Step 3	**aaa authentication login default local**	Set the login authentication to use the local username database. The **default** keyword applies the local user database authentication to all interfaces.
Step 4	**aaa authorization exec local**	Configure user AAA authorization to determine if the user is allowed to run an EXEC shell by checking the local database.
Step 5	**aaa authorization network local**	Configure user AAA authorization for all network-related service requests.

	Command	Purpose
Step 6	**username** *name* [**privilege** *level*] {**password** *encryption-type password*}	Enter the local database, and establish a username-based authentication system. Repeat this command for each user. • For *name*, specify the user ID as one word. Spaces and quotation marks are not allowed. • (Optional) For *level*, specify the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 0 gives user EXEC mode access. • For *encryption-type*, enter 0 to specify that an unencrypted password follows. Enter 7 to specify that a hidden password follows. • For *password*, specify the password the user must enter to gain access to the switch. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the **username** command.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show running-config**	Verify your entries.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable AAA, use the **no aaa new-model** global configuration command. To disable authorization, use the **no aaa authorization** {**network** | **exec**} *method1* global configuration command.

Configuring the Switch for Secure Shell

This section describes how to configure the Secure Shell (SSH) feature. SSH is a cryptographic security feature that is subject to export restrictions. To use this feature, the cryptographic (encrypted) IP services image (formerly known as the enhanced multilayer software image [EMI]) must be installed on your switch. You must obtain authorization to use this feature and to download the cryptographic software files from Cisco.com. For more information, see the release notes for this release.

This section contains this information:

- Understanding SSH, page 7-38
- Configuring SSH, page 7-39
- Displaying the SSH Configuration and Status, page 7-41

For SSH configuration examples, see the “SSH Configuration Examples” section in the “Configuring Secure Shell” chapter of the *Cisco IOS Security Configuration Guide, Cisco IOS Release 12.2*, at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fothersf/scfssh.htm

Note For complete syntax and usage information for the commands used in this section, see the command reference for this release and the command reference for Cisco IOS Release 12.2 at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/index.htm.

Understanding SSH

SSH is a protocol that provides a secure, remote connection to a device. SSH provides more security for remote connections than Telnet does by providing strong encryption when a device is authenticated. This software release supports SSH version 1 (SSHv1) and SSH version 2 (SSHv2).

This section consists of these topics:

- SSH Servers, Integrated Clients, and Supported Versions, page 7-38
- Limitations, page 7-38

SSH Servers, Integrated Clients, and Supported Versions

The SSH feature has an SSH server and an SSH integrated client, which are applications that run on the switch. You can use an SSH client to connect to a switch running the SSH server. The SSH server works with the SSH client supported in this release and with non-Cisco SSH clients. The SSH client also works with the SSH server supported in this release and with non-Cisco SSH servers.

The switch supports an SSHv1 or an SSHv2 server.

The switch supports an SSHv1 client.

SSH supports the Data Encryption Standard (DES) encryption algorithm, the Triple DES (3DES) encryption algorithm, and password-based user authentication.

SSH also supports these user authentication methods:

- TACACS+ (for more information, see the "Controlling Switch Access with TACACS+" section on page 7-9)
- RADIUS (for more information, see the "Controlling Switch Access with RADIUS" section on page 7-17)
- Local authentication and authorization (for more information, see the "Configuring the Switch for Local Authentication and Authorization" section on page 7-36)

Note This software release does not support IP Security (IPSec).

Limitations

These limitations apply to SSH:

- The switch supports Rivest, Shamir, and Adelman (RSA) authentication.
- SSH supports only the execution-shell application.
- The SSH server and the SSH client are supported only on DES (56-bit) and 3DES (168-bit) data encryption software.
- The switch does not support the Advanced Encryption Standard (AES) symmetric encryption algorithm.

Configuring SSH

This section has this configuration information:

- Configuration Guidelines, page 7-39
- Setting Up the Switch to Run SSH, page 7-39 (required)
- Configuring the SSH Server, page 7-40 (required only if you are configuring the switch as an SSH server)

Configuration Guidelines

Follow these guidelines when configuring the switch as an SSH server or SSH client:

- An RSA key pair generated by a SSHv1 server can be used by an SSHv2 server, and the reverse.
- If you get CLI error messages after entering the **crypto key generate rsa** global configuration command, an RSA key pair has not been generated. Reconfigure the host name and domain, and then enter the **crypto key generate rsa** command. For more information, see the "Setting Up the Switch to Run SSH" section on page 7-39.
- When generating the RSA key pair, the message "No host name specified" might appear. If it does, you must configure a host name by using the **hostname** global configuration command.
- When generating the RSA key pair, the message "No domain specified" might appear. If it does, you must configure an IP domain name by using the **ip domain-name** global configuration command.
- When configuring the local authentication and authorization authentication method, make sure that AAA is disabled on the console.

Setting Up the Switch to Run SSH

Follow these steps to set up your switch to run SSH:

1. Download the cryptographic software image from Cisco.com. This step is required. For more information, see the release notes for this release.
2. Configure a host name and IP domain name for the switch. Follow this procedure only if you are configuring the switch as an SSH server.
3. Generate an RSA key pair for the switch, which automatically enables SSH. Follow this procedure only if you are configuring the switch as an SSH server.
4. Configure user authentication for local or remote access. This step is required. For more information, see the "Configuring the Switch for Local Authentication and Authorization" section on page 7-36.

Beginning in privileged EXEC mode, follow these steps to configure a host name and an IP domain name and to generate an RSA key pair. This procedure is required if you are configuring the switch as an SSH server.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**hostname** *hostname*	Configure a host name for your switch.
Step 3	**ip domain-name** *domain_name*	Configure a host domain for your switch.

	Command	Purpose
Step 4	**crypto key generate rsa**	Enable the SSH server for local and remote authentication on the switch and generate an RSA key pair. We recommend that a minimum modulus size of 1024 bits. When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip ssh** or **show ssh**	Show the version and configuration information for your SSH server. Show the status of the SSH server on the switch.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete the RSA key pair, use the **crypto key zeroize rsa** global configuration command. After the RSA key pair is deleted, the SSH server is automatically disabled.

Configuring the SSH Server

Beginning in privileged EXEC mode, follow these steps to configure the SSH server:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip ssh version** [**1** | **2**]	(Optional) Configure the switch to run SSH version 1 or SSH version 2. • **1**—Configure the switch to run SSH version 1. • **2**—Configure the switch to run SSH version 2. If you do not enter this command or do not specify a keyword, the SSH server selects the latest SSH version supported by the SSH client. For example, if the SSH client sports SSHv1 and SSHv2, the SSH server selects SSHv2.
Step 3	**ip ssh** {**timeout** *seconds* | **authentication-retries** *number*}	Configure the SSH control parameters: • Specify the time-out value in seconds; the default is 120 seconds. The range is 0 to 120 seconds. This parameter applies to the SSH negotiation phase. After the connection is established, the switch uses the default time-out values of the CLI-based sessions. By default, up to five simultaneous, encrypted SSH connections for multiple CLI-based sessions over the network are available (session 0 to session 4). After the execution shell starts, the CLI-based session time-out value returns to the default of 10 minutes. • Specify the number of times that a client can re-authenticate to the server. The default is 3; the range is 0 to 5. Repeat this step when configuring both parameters.
Step 4	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 5	**show ip ssh** or **show ssh**	Display the version and configuration information for your SSH server. Display the status of the SSH server connections on the switch.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default SSH control parameters, use the **no ip ssh** {**timeout** | **authentication-retries**} global configuration command.

Displaying the SSH Configuration and Status

To display the SSH server configuration and status, use one or more of the privileged EXEC commands in Table 7-3:

Table 7-3 Commands for Displaying the SSH Server Configuration and Status

Command	Purpose
show ip ssh	Shows the version and configuration information for the SSH server.
show ssh	Shows the status of the SSH server.

For more information about these commands, see the "Secure Shell Commands" section in the "Other Security Features" chapter of the *Cisco IOS Security Command Reference, Cisco IOS Release 12.2,* at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_r/fothercr/srfssh.htm.

Configuring the Switch for Secure Socket Layer HTTP

This section describes how to configure Secure Socket Layer (SSL) version 3.0 support for the HTTP1.1 server and client. SSL provides server authentication, encryption, and message integrity, as well as HTTP client authentication, to allow secure HTTP communications.To use this feature, the cryptographic (encrypted) software image must be installed on your switch. You must obtain authorization to use this feature and to download the cryptographic software files from Cisco.com. For more information about the crypto image, see the release notes for this release.

This section contains this information:

- Understanding Secure HTTP Servers and Clients, page 7-42
- Configuring Secure HTTP Servers and Clients, page 7-44
- Displaying Secure HTTP Server and Client Status, page 7-48

For configuration examples and complete syntax and usage information for the commands used in this section, see the "HTTPS - HTTP Server and Client with SSL 3.0" feature description for Cisco IOS Release 12.2(15)T at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t15/ftsslsht.htm

Understanding Secure HTTP Servers and Clients

On a secure HTTP connection, data to and from an HTTP server is encrypted before being sent over the Internet. HTTP with SSL encryption provides a secure connection to allow such functions as configuring a switch from a Web browser. Cisco's implementation of the secure HTTP server and secure HTTP client uses an implementation of SSL version 3.0 with application-layer encryption. HTTP over SSL is abbreviated as HTTPS; the URL of a secure connection begins with https:// instead of http://.

The primary role of the HTTP secure server (the switch) is to listen for HTTPS requests on a designated port (the default HTTPS port is 443) and pass the request to the HTTP 1.1 Web server. The HTTP 1.1 server processes requests and passes responses (pages) back to the HTTP secure server, which, in turn, responds to the original request.

The primary role of HTTP secure client (the web browser) is to respond to Cisco IOS application requests for HTTPS User Agent services, perform HTTPS User Agent services for the application, and pass the response back to the application.

Certificate Authority Trustpoints

Certificate authorities (CAs) manage certificate requests and issue certificates to participating network devices. These services provide centralized security key and certificate management for the participating devices. Specific CA servers are referred to as *trustpoints*.

When a connection attempt is made, the HTTPS server provides a secure connection by issuing a certified X.509v3 certificate, obtained from a specified CA trustpoint, to the client. The client (usually a Web browser), in turn, has a public key that allows it to authenticate the certificate.

For secure HTTP connections, we highly recommend that you configure a CA trustpoint. If a CA trustpoint is not configured for the device running the HTTPS server, the server certifies itself and generates the needed RSA key pair. Because a self-certified (self-signed) certificate does not provide adequate security, the connecting client generates a notification that the certificate is self-certified, and the user has the opportunity to accept or reject the connection. This option is useful for internal network topologies (such as testing).

If you do not configure a CA trustpoint, when you enable a secure HTTP connection, either a temporary or a persistent self-signed certificate for the secure HTTP server (or client) is automatically generated.

- If the switch is not configured with a hostname and a domain name, a temporary self-signed certificate is generated. If the switch reboots, any temporary self-signed certificate is lost, and a new temporary new self-signed certificate is assigned.
- If the switch has been configured with a host and domain name, a persistent self-signed certificate is generated. This certificate remains active if you reboot the switch or if you disable the secure HTTP server so that it will be there the next time you re-enable a secure HTTP connection.

Note The certificate authorities and trustpoints must be configured on each device individually. Copying them from other devices makes them invalid on the switch.

If a self-signed certificate has been generated, this information is included in the output of the **show running-config** privileged EXEC command. This is a partial sample output from that command displaying a self-signed certificate.

```
Switch# show running-config
Building configuration...

<output truncated>
```

```
crypto pki trustpoint TP-self-signed-3080755072
 enrollment selfsigned
 subject-name cn=IOS-Self-Signed-Certificate-3080755072
 revocation-check none
 rsakeypair TP-self-signed-3080755072
!
!
crypto ca certificate chain TP-self-signed-3080755072
 certificate self-signed 01
  3082029F 30820208 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  59312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274
  69666963 6174652D 33303830 37353530 37323126 30240609 2A864886 F70D0109
  02161743 45322D33 3535302D 31332E73 756D6D30 342D3335 3530301E 170D3933
  30333031 30303030 35395A17 0D323030 31303130 30303030 305A3059 312F302D

<output truncated>
```

You can remove this self-signed certificate by disabling the secure HTTP server and entering the **no crypto pki trustpoint TP-self-signed-30890755072** global configuration command. If you later re-enable a secure HTTP server, a new self-signed certificate is generated.

Note The values following *TP self-signed* depend on the serial number of the device.

You can use an optional command (**ip http secure-client-auth**) to allow the HTTPS server to request an X.509v3 certificate from the client. Authenticating the client provides more security than server authentication by itself.

For additional information on Certificate Authorities, see the "Configuring Certification Authority Interoperability" chapter in the *Cisco IOS Security Configuration Guide, Release 12.2*.

CipherSuites

A CipherSuite specifies the encryption algorithm and the digest algorithm to use on a SSL connection. When connecting to the HTTPS server, the client Web browser offers a list of supported CipherSuites, and the client and server negotiate the best encryption algorithm to use from those on the list that are supported by both. For example, Netscape Communicator 4.76 supports U.S. security with RSA Public Key Cryptography, MD2, MD5, RC2-CBC, RC4, DES-CBC, and DES-EDE3-CBC.

For the best possible encryption, you should use a client browser that supports 128-bit encryption, such as Microsoft Internet Explorer version 5.5 (or later) or Netscape Communicator version 4.76 (or later). The SSL_RSA_WITH_DES_CBC_SHA CipherSuite provides less security than the other CipherSuites, as it does not offer 128-bit encryption.

The more secure and more complex CipherSuites require slightly more processing time. This list defines the CipherSuites supported by the switch and ranks them from fastest to slowest in terms of router processing load (speed):

1. SSL_RSA_WITH_DES_CBC_SHA—RSA key exchange (RSA Public Key Cryptography) with DES-CBC for message encryption and SHA for message digest
2. SSL_RSA_WITH_RC4_128_MD5—RSA key exchange with RC4 128-bit encryption and MD5 for message digest

3. SSL_RSA_WITH_RC4_128_SHA—RSA key exchange with RC4 128-bit encryption and SHA for message digest
4. SSL_RSA_WITH_3DES_EDE_CBC_SHA—RSA key exchange with 3DES and DES-EDE3-CBC for message encryption and SHA for message digest

RSA (in conjunction with the specified encryption and digest algorithm combinations) is used for both key generation and authentication on SSL connections. This usage is independent of whether or not a CA trustpoint is configured.

Configuring Secure HTTP Servers and Clients

This section incudes procedures for configuring SSL on HTTP servers and clients. These procedures are included:

- Default SSL Configuration, page 7-44
- SSL Configuration Guidelines, page 7-44
- Configuring a CA Trustpoint, page 7-44
- Configuring the Secure HTTP Server, page 7-45
- Configuring the Secure HTTP Client, page 7-47

Default SSL Configuration

The standard HTTP server is enabled.

SSL is enabled.

No CA trustpoints are configured.

No self-signed certificates are generated.

SSL Configuration Guidelines

When SSL is used in a switch cluster, the SSL session terminates at the cluster commander. Cluster member switches must run standard HTTP.

Before you configure a CA trustpoint, you should ensure that the system clock is set. If the clock is not set, the certificate is rejected due to an incorrect date.

Configuring a CA Trustpoint

For secure HTTP connections, we recommend that you configure an official CA trustpoint. A CA trustpoint is more secure than a self-signed certificate.

Beginning in privileged EXEC mode, follow these steps to configure a CA trustpoint:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**hostname** *hostname*	Specify the hostname of the switch (required only if you have not previously configured a hostname). The hostname is required for security keys and certificates.

	Command	Purpose
Step 3	**ip domain-name** *domain-name*	Specify the IP domain name of the switch (required only if you have not previously configured an IP domain name). The domain name is required for security keys and certificates.
Step 4	**crypto key generate rsa**	(Optional) Generate an RSA key pair. RSA key pairs are required before you can obtain a certificate for the switch. RSA key pairs are generated automatically. You can use this command to regenerate the keys, if needed.
Step 5	**crypto ca trustpoint** *name*	Specify a local configuration name for the CA trustpoint and enter CA trustpoint configuration mode.
Step 6	**enrollment url** *url*	Specify the URL to which the switch should send certificate requests.
Step 7	**enrollment http-proxy** *host-name port-number*	(Optional) Configure the switch to obtain certificates from the CA through an HTTP proxy server.
Step 8	**crl query** *url*	Configure the switch to request a certificate revocation list (CRL) to ensure that the certificate of the peer has not been revoked.
Step 9	**primary**	(Optional) Specify that the trustpoint should be used as the primary (default) trustpoint for CA requests.
Step 10	**exit**	Exit CA trustpoint configuration mode and return to global configuration mode.
Step 11	**crypto ca authentication** *name*	Authenticate the CA by getting the public key of the CA. Use the same name used in Step 5.
Step 12	**crypto ca enroll** *name*	Obtain the certificate from the specified CA trustpoint. This command requests a signed certificate for each RSA key pair.
Step 13	**end**	Return to privileged EXEC mode.
Step 14	**show crypto ca trustpoints**	Verify the configuration.
Step 15	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no crypto ca trustpoint** *name* global configuration command to delete all identity information and certificates associated with the CA.

Configuring the Secure HTTP Server

If you are using a certificate authority for certification, you should use the previous procedure to configure the CA trustpoint on the switch before enabling the HTTP server. If you have not configured a CA trustpoint, a self-signed certificate is generated the first time that you enable the secure HTTP server. After you have configured the server, you can optionally configure options (path, access list to apply, maximum number of connections, or timeout policy) that apply to both standard and secure HTTP servers.

Beginning in privileged EXEC mode, follow these steps to configure a secure HTTP server:

	Command	Purpose

Step 1	show ip http server status	(Optional) Display the status of the HTTP server to determine if the secure HTTP server feature is supported in the software. You should see one of these lines in the output:
		`HTTP secure server capability: Present`
		`or`
		`HTTP secure server capability: Not present`

Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**ip http secure-server**	Enable the HTTPS server if it has been disabled. The HTTPS server is enabled by default.
Step 4	**ip http secure-port** *port-number*	(Optional) Specify the port number to be used for the HTTPS server. The default port number is 443. Valid options are 443 or any number in the range 1025 to 65535.
Step 5	**ip http secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]}	(Optional) Specify the CipherSuites (encryption algorithms) to be used for encryption over the HTTPS connection. If you do not have a reason to specify a particularly CipherSuite, you should allow the server and client to negotiate a CipherSuite that they both support. This is the default.
Step 6	**ip http secure-client-auth**	(Optional) Configure the HTTP server to request an X.509v3 certificate from the client for authentication during the connection process. The default is for the client to request a certificate from the server, but the server does not attempt to authenticate the client.
Step 7	**ip http secure-trustpoint** *name*	Specify the CA trustpoint to use to get an X.509v3 security certificate and to authenticate the client certificate connection. **Note** Use of this command assumes you have already configured a CA trustpoint according to the previous procedure.
Step 8	**ip http path** *path-name*	(Optional) Set a base HTTP path for HTML files. The path specifies the location of the HTTP server files on the local system (usually located in system flash memory).
Step 9	**ip http access-class** *access-list-number*	(Optional) Specify an access list to use to allow access to the HTTP server.
Step 10	**ip http max-connections** *value*	(Optional) Set the maximum number of concurrent connections that are allowed to the HTTP server. The range is 1 to 16; the default value is 5.
Step 11	**ip http timeout-policy idle** *seconds* **life** *seconds* **requests** *value*	(Optional) Specify how long a connection to the HTTP server can remain open under the defined circumstances: • **idle**—the maximum time period when no data is received or response data cannot be sent. The range is 1 to 600 seconds. The default is 180 seconds (3 minutes). • **life**—the maximum time period from the time that the connection is established. The range is 1 to 86400 seconds (24 hours). The default is 180 seconds. • **requests**—the maximum number of requests processed on a persistent connection. The maximum value is 86400. The default is 1.
Step 12	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 13	**show ip http server secure status**	Display the status of the HTTP secure server to verify the configuration.
Step 14	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip http server** global configuration command to disable the standard HTTP server. Use the **no ip http secure-server** global configuration command to disable the secure HTTP server. Use the **no ip http secure-port** and the **no ip http secure-ciphersuite** global configuration commands to return to the default settings. Use the **no ip http secure-client-auth** global configuration command to remove the requirement for client authentication.

To verify the secure HTTP connection by using a Web browser, enter https://*URL*, where the URL is the IP address or hostname of the server switch. If you configure a port other than the default port, you must also specify the port number after the URL. For example:

```
https://209.165.129:1026
```
or
```
https://host.domain.com:1026
```

Configuring the Secure HTTP Client

The standard HTTP client and secure HTTP client are always enabled. A certificate authority is required for secure HTTP client certification. This procedure assumes that you have previously configured a CA trustpoint on the switch. If a CA trustpoint is not configured and the remote HTTPS server requires client authentication, connections to the secure HTTP client fail.

Beginning in privileged EXEC mode, follow these steps to configure a secure HTTP client:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip http client secure-trustpoint** *name*	(Optional) Specify the CA trustpoint to be used if the remote HTTP server requests client authentication. Using this command assumes that you have already configured a CA trustpoint by using the previous procedure. The command is optional if client authentication is not needed or if a primary trustpoint has been configured.
Step 3	**ip http client secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]}	(Optional) Specify the CipherSuites (encryption algorithms) to be used for encryption over the HTTPS connection. If you do not have a reason to specify a particular CipherSuite, you should allow the server and client to negotiate a CipherSuite that they both support. This is the default.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip http client secure status**	Display the status of the HTTP secure server to verify the configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip http client secure-trustpoint** *name* to remove a client trustpoint configuration. Use the **no ip http client secure-ciphersuite** to remove a previously configured CipherSuite specification for the client.

Displaying Secure HTTP Server and Client Status

To display the SSL secure server and client status, use the privileged EXEC commands in Table 7-4:

Table 7-4 Commands for Displaying the SSL Secure Server and Client Status

Command	Purpose
show ip http client secure status	Shows the HTTP secure client configuration.
show ip http server secure status	Shows the HTTP secure server configuration.
show running-config	Shows the generated self-signed certificate for secure HTTP connections.

Configuring the Switch for Secure Copy Protocol

The Secure Copy (SCP) feature provides a secure and authenticated method for copying switch configurations or switch image files. SCP relies on Secure Shell (SSH), an application and a protocol that provide a secure replacement for the Berkeley r-tools.

For SSH to work, the switch needs an RSA public/private key pair. This is the same with SCP, which relies on SSH for its secure transport.

Because SSH relies on AAA authentication, and SCP relies further on AAA authorization, correct configuration is necessary.

- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the switch.
- Because SCP relies on SSH for its secure transport, the switch must have an Rivest, Shamir, and Adelman (RSA) key pair.

Note When using SCP, you cannot enter the password into the copy command. You must enter the password when prompted.

To configure Secure Copy feature, you should understand these concepts.

- The behavior of SCP is similar to that of remote copy (rcp), which comes from the Berkeley r-tools suite, except that SCP relies on SSH for security. SCP also requires that authentication, authorization, and accounting (AAA) authorization be configured so the router can determine whether the user has the correct privilege level.
- A user who has appropriate authorization can use SCP to copy any file in the Cisco IOS File System (IFS) to and from a switch by using the **copy** command. An authorized administrator can also do this from a workstation.

For more information on how to configure and verify SCP, see the "Secure Copy Protocol" chapter of the *Cisco IOS New Features, Cisco IOS Release 12.2,* at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t2/ftscp.htm

CHAPTER 8

Configuring IEEE 802.1x Port-Based Authentication

This chapter describes how to configure IEEE 802.1x port-based authentication on the Catalyst 3550 switch. IEEE 802.1x authentication prevents unauthorized devices (clients) from gaining access to the network.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release and the "RADIUS Commands" section in the *Cisco IOS Security Command Reference, Release 12.2*.

This chapter consists of these sections:

- Understanding IEEE 802.1x Port-Based Authentication, page 8-1
- Configuring IEEE 802.1x Authentication, page 8-19
- Displaying IEEE 802.1x Statistics and Status, page 8-42

Understanding IEEE 802.1x Port-Based Authentication

The IEEE 802.1x standard defines a client-server-based access control and authentication protocol that prevents unauthorized clients from connecting to a LAN through publicly accessible ports unless they are properly authenticated. The authentication server authenticates each client connected to a switch port before making available any services offered by the switch or the LAN.

Until the client is authenticated, IEEE 802.1x access control allows only Extensible Authentication Protocol over LAN (EAPOL), Cisco Discovery Protocol (CDP), and Spanning Tree Protocol (STP) traffic through the port to which the client is connected. After authentication is successful, normal traffic can pass through the port.

These sections describe IEEE 802.1x port-based authentication:

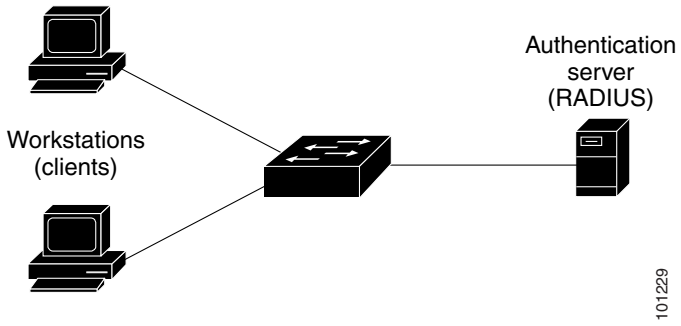
- Device Roles, page 8-2
- Authentication Process, page 8-3
- Authentication Initiation and Message Exchange, page 8-5
- Ports in Authorized and Unauthorized States, page 8-7
- IEEE 802.1x Host Mode, page 8-7
- IEEE 802.1x Accounting, page 8-8

- IEEE 802.1x Accounting Attribute-Value Pairs, page 8-8
- Using IEEE 802.1x Authentication with VLAN Assignment, page 8-9
- Using IEEE 802.1x Authentication with Per-User ACLs, page 8-10
- Using IEEE 802.1x Authentication with Guest VLAN, page 8-11
- Using IEEE 802.1x Authentication with Restricted VLAN, page 8-13
- Using IEEE 802.1x Authentication with Inaccessible Authentication Bypass, page 8-13
- Using IEEE 802.1x Authentication with Voice VLAN Ports, page 8-15
- Using IEEE 802.1x Authentication with Port Security, page 8-15
- Using IEEE 802.1x Authentication with Wake-on-LAN, page 8-16
- Using IEEE 802.1x Authentication with MAC Authentication Bypass, page 8-17
- Network Admission Control Layer 2 IEEE 802.1x Validation, page 8-18
- Web Authentication with Automatic MAC Check, page 8-18

Device Roles

With IEEE 802.1x port-based authentication, the devices in the network have specific roles as shown in Figure 8-1.

Figure 8-1 IEEE 802.1x Device Roles



- *Client*—the device (workstation) that requests access to the LAN and switch services and responds to requests from the switch.The workstation must be running IEEE 802.1x-compliant client software such as that offered in the Microsoft Windows XP operating system. (The client is the *supplicant* in the IEEE 802.1x standard.)

  Note To resolve Windows XP network connectivity and IEEE 802.1x authentication issues, read the Microsoft Knowledge Base article at this URL: http://support.microsoft.com/support/kb/articles/Q303/5/97.ASP

- *Authentication server*—performs the actual authentication of the client. The authentication server validates the identity of the client and notifies the switch whether or not the client is authorized to access the LAN and switch services. Because the switch acts as the proxy, the authentication service is transparent to the client. In this release, the RADIUS security system with Extensible Authentication Protocol (EAP) extensions is the only supported authentication server. It is available

in Cisco Secure Access Control Server Version 3.0 or later. RADIUS operates in a client/server model in which secure authentication information is exchanged between the RADIUS server and one or more RADIUS clients.

- *Switch* (edge switch or wireless access point)—controls the physical access to the network based on the authentication status of the client. The switch acts as an intermediary (proxy) between the client and the authentication server, requesting identity information from the client, verifying that information with the authentication server, and relaying a response to the client. The switch includes the RADIUS client, which is responsible for encapsulating and decapsulating the EAP frames and interacting with the authentication server.

  When the switch receives EAPOL frames and relays them to the authentication server, the Ethernet header is stripped, and the remaining EAP frame is re-encapsulated in the RADIUS format. The EAP frames are not modified during encapsulation, and the authentication server must support EAP within the native frame format. When the switch receives frames from the authentication server, the server frame header is removed, leaving the EAP frame, which is then encapsulated for Ethernet and sent to the client.

The devices that can act as intermediaries include the Catalyst 3750, 3560, 3550, 2970, 2955, 2950, 2940 switches, or a wireless access point. These devices must be running software that supports the RADIUS client and IEEE 802.1x authentication.

Authentication Process

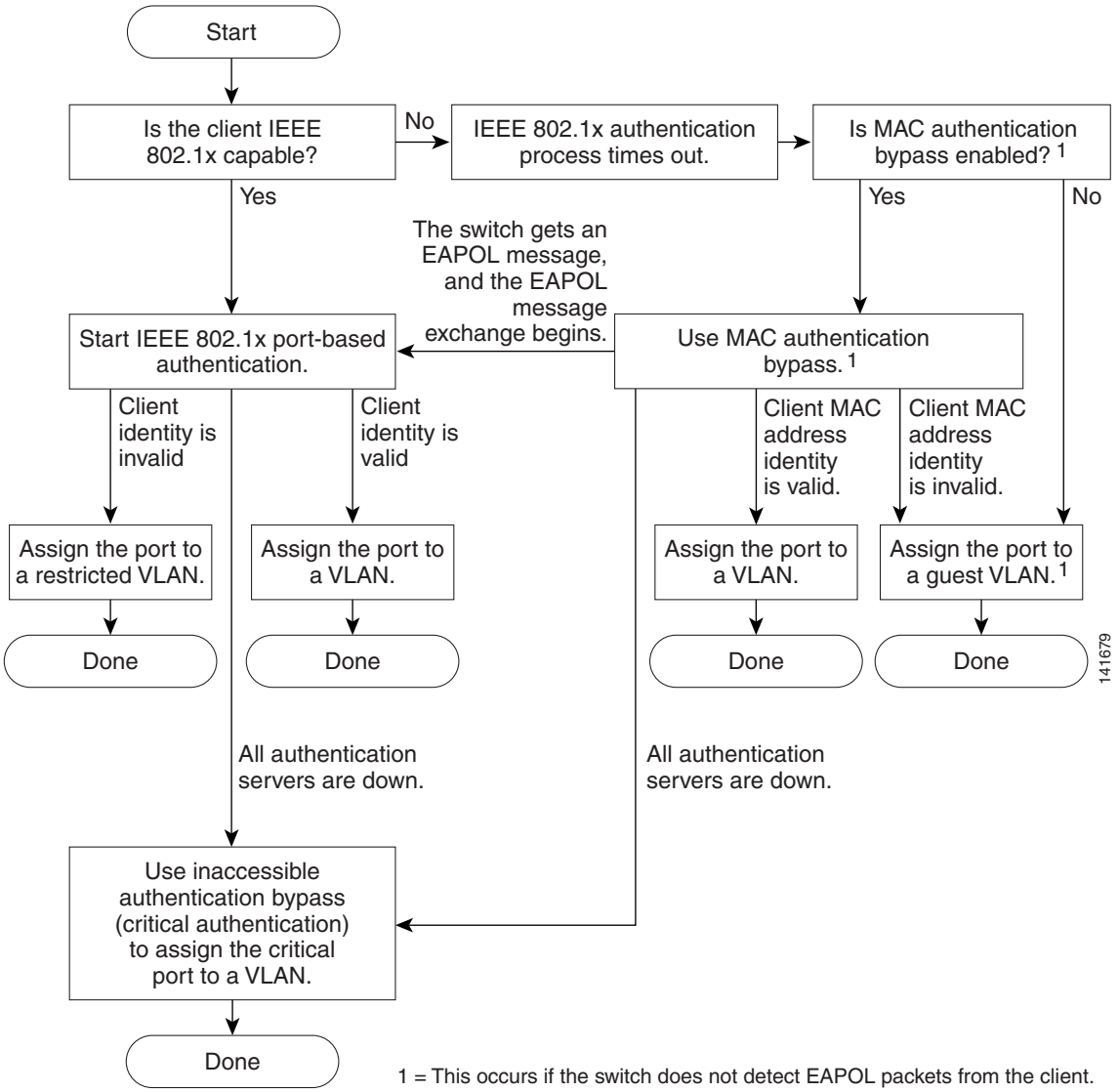
When IEEE 802.1x port-based authentication is enabled and the client supports IEEE 802.1x-compliant client software, these events occur:

- If the client identity is valid and the IEEE 802.1x authentication succeeds, the switch grants the client access to the network.
- If IEEE 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can use the client MAC address for authorization. If the client MAC address is valid and the authorization succeeds, the switch grants the client access to the network. If the client MAC address is invalid and the authorization fails, the switch assigns the client to a guest VLAN that provides limited services if a guest VLAN is configured.
- If the switch gets an invalid identity from an IEEE 802.1x-capable client and a restricted VLAN is specified, the switch can assign the client to a restricted VLAN that provides limited services.
- If the RADIUS authentication server is unavailable (down) and inaccessible authentication bypass is enabled, the switch grants the client access to the network by putting the port in the critical-authentication state in the RADIUS-configured or the user-specified access VLAN.

Note Inaccessible authentication bypass is also referred to as critical authentication or the authentication, authorization, and accounting (AAA) fail policy.

Figure 8-2 shows the authentication process.

Figure 8-2 Authentication Flowchart



The switch re-authenticates a client when one of these situations occurs:

- Periodic re-authentication is enabled, and the re-authentication timer expires.

  You can configure the re-authentication timer to use a switch-specific value or to be based on values from the RADIUS server.

  After IEEE 802.1x authentication using a RADIUS server is configured, the switch uses timers based on the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute [29]).

  The Session-Timeout RADIUS attribute (Attribute[27]) specifies the time after which re-authentication occurs.

The Termination-Action RADIUS attribute (Attribute [29]) specifies the action to take during re-authentication. The actions can be *Initialize* or *ReAuthenticate*. When the *Initialize* action is set (the attribute value is *DEFAULT*), the IEEE 802.1x session ends, and connectivity is lost during re-authentication. When the *ReAuthenticate* action is set (the attribute value is RADIUS-Request), the session is not affected during re-authentication.

- You manually re-authenticate the client by entering the **dot1x re-authenticate interface** *interface-id* privileged EXEC command.

Authentication Initiation and Message Exchange

During IEEE 8021.x authentication, the switch or the client can initiate authentication. If you enable authentication on a port by using the **dot1x port-control auto** interface configuration command, the switch initiates authentication when the link state changes from down to up or periodically as long as the port remains up and unauthenticated. The switch sends an EAP-request/identity frame to the client to request its identity. Upon receipt of the frame, the client responds with an EAP-response/identity frame.

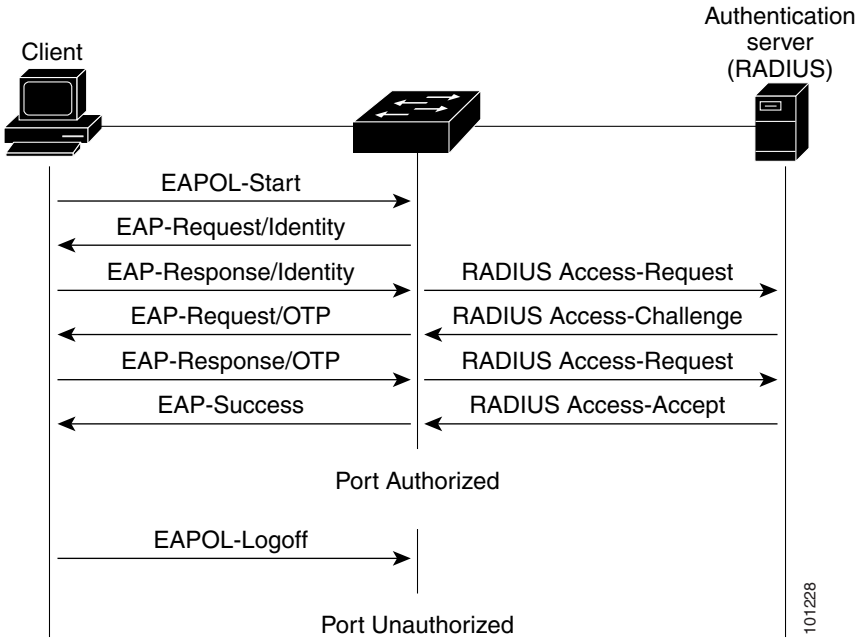
However, if during bootup, the client does not receive an EAP-request/identity frame from the switch, the client can initiate authentication by sending an EAPOL-start frame, which prompts the switch to request the client's identity.

Note If IEEE 802.1x authentication is not enabled or supported on the network access device, any EAPOL frames from the client are dropped. If the client does not receive an EAP-request/identity frame after three attempts to start authentication, the client sends frames as if the port is in the authorized state. A port in the authorized state effectively means that the client has been successfully authenticated. For more information, see the "Ports in Authorized and Unauthorized States" section on page 8-7.

When the client supplies its identity, the switch begins its role as the intermediary, passing EAP frames between the client and the authentication server until authentication succeeds or fails. If the authentication succeeds, the switch port becomes authorized. If the authentication fails, authentication can be retried, the port might be assigned to a VLAN that provides limited services, or network access is not granted. For more information, see the "Ports in Authorized and Unauthorized States" section on page 8-7.

The specific exchange of EAP frames depends on the authentication method being used. Figure 8-3 shows a message exchange initiated by the client using the One-Time-Password (OTP) authentication method with a RADIUS server.

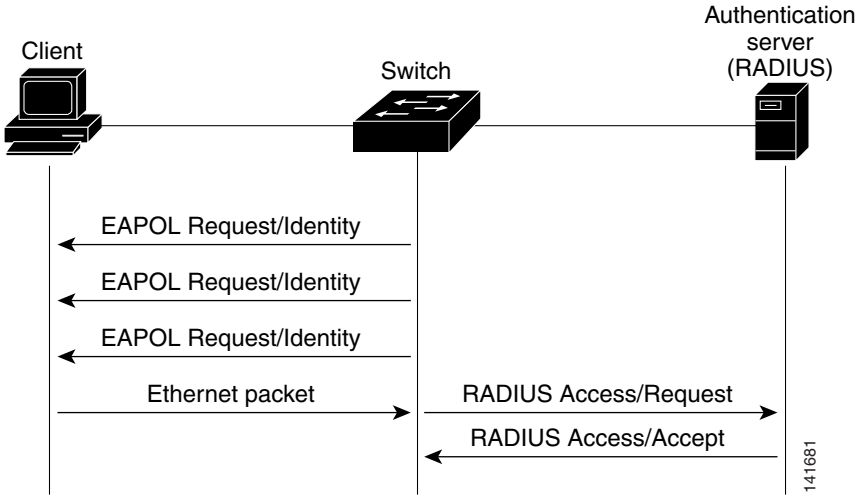
Figure 8-3 Message Exchange



If IEEE 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can authorize the client when the switch detects an Ethernet packet from the client. The switch uses the MAC address of the client as its identity and includes this information in the RADIUS-access/request frame that is sent to the RADIUS server. After the server sends the switch the RADIUS-access/accept frame (authorization is successful), the port becomes authorized. If authorization fails and a guest VLAN is specified, the switch assigns the port to the guest VLAN. If the switch detects an EAPOL packet while waiting for an Ethernet packet, the switch stops the MAC authentication bypass process and initiates authentication using IEEE 802.1x authentication.

Figure 8-4 shows the message exchange during MAC authentication bypass.

Figure 8-4 Message Exchange During MAC Authentication Bypass



Ports in Authorized and Unauthorized States

During IEEE 802.1x authentication, depending on the switch port state, the switch can grant a client access to the network. The port starts in the *unauthorized* state. While in this state, the port that is not configured as a voice VLAN port disallows all ingress and egress traffic except for IEEE 802.1x authentication, CDP, and STP packets. When a client is successfully authenticated, the port changes to the *authorized* state, allowing all traffic for the client to flow normally. If the port is configured as a voice VLAN port, the port allows VoIP traffic and IEEE 802.1x protocol packets before the client is successfully authenticated.

If a client that does not support IEEE 802.1x authentication connects to an unauthorized IEEE 802.1x port, the switch requests the client's identity. In this situation, the client does not respond to the request, the port remains in the unauthorized state, and the client is not granted access to the network.

In contrast, when an IEEE 802.1x-enabled client connects to a port that is not running the IEEE 802.1x standard, the client initiates the authentication process by sending the EAPOL-start frame. When no response is received, the client sends the request for a fixed number of times. Because no response is received, the client begins sending frames as if the port is in the authorized state.

You control the port authorization state by using the **dot1x port-control** interface configuration command and these keywords:

- **force-authorized**—disables IEEE 802.1x authentication and causes the port to transition to the authorized state without any authentication exchange required. The port sends and receives normal traffic without IEEE 802.1x-based authentication of the client. This is the default setting.
- **force-unauthorized**—causes the port to remain in the unauthorized state, ignoring all attempts by the client to authenticate. The switch cannot provide authentication services to the client through the interface.
- **auto**—enables IEEE 802.1x authentication and causes the port to begin in the unauthorized state, allowing only EAPOL frames to be sent and received through the port. The authentication process begins when the link state of the port transitions from down to up or when an EAPOL-start frame is received. The switch requests the identity of the client and begins relaying authentication messages between the client and the authentication server. Each client attempting to access the network is uniquely identified by the switch by using the client's MAC address.

If the client is successfully authenticated (receives an Accept frame from the authentication server), the port state changes to authorized, and all frames from the authenticated client are allowed through the port. If the authentication fails, the port remains in the unauthorized state, but authentication can be retried. If the authentication server cannot be reached, the switch can resend the request. If no response is received from the server after the specified number of attempts, authentication fails, and network access is not granted.

When a client logs off, it sends an EAPOL-logoff message, causing the switch port to transition to the unauthorized state.

If the link state of a port transitions from up to down, or if an EAPOL-logoff frame is received, the port returns to the unauthorized state.

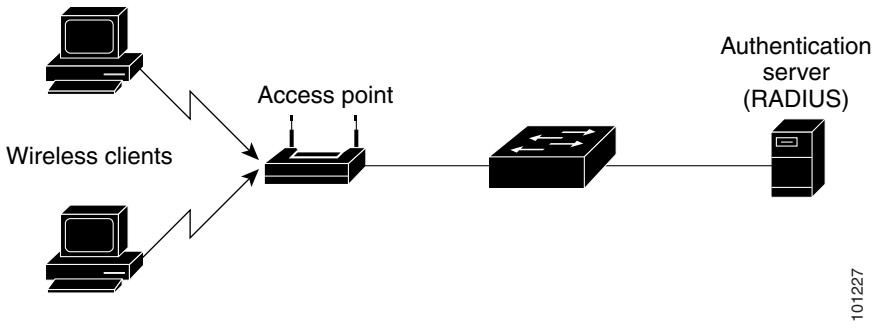
IEEE 802.1x Host Mode

You can configure an IEEE 802.1x port for single-host or for multiple-hosts mode. In single-host mode (see Figure 8-1 on page 8-2), only one client can be connected to the IEEE 802.1x-enabled switch port. The switch detects the client by sending an EAPOL frame when the port link state changes to the up state. If a client leaves or is replaced with another client, the switch changes the port link state to down, and the port returns to the unauthorized state.

In multiple-hosts mode, you can attach multiple hosts to a single IEEE 802.1x-enabled port. Figure 8-5 on page 8-8 shows IEEE 802.1x port-based authentication in a wireless LAN. In this mode, only one of the attached clients must be authorized for all clients to be granted network access. If the port becomes unauthorized (re-authentication fails or an EAPOL-logoff message is received), the switch denies network access to all of the attached clients. In this topology, the wireless access point is responsible for authenticating the clients attached to it, and it also acts as a client to the switch.

With the multiple-hosts mode enabled, you can use IEEE 802.1x authentication to authenticate the port and port security to manage network access for all MAC addresses, including that of the client.

Figure 8-5 Multiple Host Mode Example



IEEE 802.1x Accounting

The IEEE 802.1x standard defines how users are authorized and authenticated for network access but does not keep track of network usage. IEEE 802.1x accounting is disabled by default. You can enable IEEE 802.1x accounting to monitor this activity on IEEE 802.1x-enabled ports:

- User successfully authenticates.
- User logs off.
- Link-down occurs.
- Re-authentication successfully occurs.
- Re-authentication fails.

The switch does not log IEEE 802.1x accounting information. Instead, it sends this information to the RADIUS server, which must be configured to log accounting messages.

IEEE 802.1x Accounting Attribute-Value Pairs

The information sent to the RADIUS server is represented in the form of Attribute-Value (AV) pairs. These AV pairs provide data for different applications. (For example, a billing application might require information that is in the Acct-Input-Octets or the Acct-Output-Octets attributes of a RADIUS packet.)

AV pairs are automatically sent by a switch that is configured for IEEE 802.1x accounting. Three types of RADIUS accounting packets are sent by a switch:

- START–sent when a new user session starts
- INTERIM–sent during an existing session for updates
- STOP–sent when a session terminates

Table 8-1 lists the AV pairs and when they are sent are sent by the switch:

Table 8-1 Accounting AV Pairs

Attribute Number	AV Pair Name	START	INTERIM	STOP
Attribute[1]	User-Name	Always	Always	Always
Attribute[4]	NAS-IP-Address	Always	Always	Always
Attribute[5]	NAS-Port	Always	Always	Always
Attribute[8]	Framed-IP-Address	Never	Sometimes[1]	Sometimes[1]
Attribute[25]	Class	Always	Always	Always
Attribute[30]	Called-Station-ID	Always	Always	Always
Attribute[31]	Calling-Station-ID	Always	Always	Always
Attribute[40]	Acct-Status-Type	Always	Always	Always
Attribute[41]	Acct-Delay-Time	Always	Always	Always
Attribute[42]	Acct-Input-Octets	Never	Always	Always
Attribute[43]	Acct-Output-Octets	Never	Always	Always
Attribute[44]	Acct-Session-ID	Always	Always	Always
Attribute[45]	Acct-Authentic	Always	Always	Always
Attribute[46]	Acct-Session-Time	Never	Always	Always
Attribute[49]	Acct-Terminate-Cause	Never	Never	Always
Attribute[61]	NAS-Port-Type	Always	Always	Always

1. The Framed-IP-Address AV pair is sent only if a valid Dynamic Host Control Protocol (DHCP) binding exists for the host in the DHCP snooping bindings table.

You can view the AV pairs that are being sent by the switch by entering the **debug radius accounting** privileged EXEC command. For more information about this command, see the *Cisco IOS Debug Command Reference, Release 12.2* at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122sup/122debug

For more information about AV pairs, see RFC 3580, “IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines.”

Using IEEE 802.1x Authentication with VLAN Assignment

You can limit network access for certain users by using VLAN assignment. After successful IEEE 802.1x authentication of a port, the RADIUS server sends the VLAN assignment to configure the switch port. The RADIUS server database maintains the username-to-VLAN mappings, assigning the VLAN based on the username of the client connected to the switch port.

When configured on the switch and the RADIUS server, IEEE 802.1x authentication with VLAN assignment has these characteristics:

- If no VLAN is supplied by the RADIUS server or if IEEE 802.1x authentication is disabled, the port is configured in its access VLAN after successful authentication.
- If IEEE 802.1x authentication is enabled but the VLAN information from the RADIUS server is not valid, the port returns to the unauthorized state and remains in the configured access VLAN. This prevents ports from appearing unexpectedly in an inappropriate VLAN because of a configuration error.

  Configuration errors could include specifying a VLAN for a routed port, a malformed VLAN ID, a nonexistent or internal (routed port) VLAN ID, or attempted assignment to a voice VLAN ID.
- If IEEE 802.1x authentication is enabled and all information from the RADIUS server is valid, the port is placed in the specified VLAN after authentication.
- If the multiple-hosts mode is enabled on an IEEE 802.1x port, all hosts are placed in the same VLAN (specified by the RADIUS server) as the first authenticated host.
- If IEEE 802.1x authentication and port security are enabled on a port, the port is placed in the RADIUS-server assigned VLAN.
- If IEEE 802.1x authentication is disabled on the port, it is returned to the configured access VLAN.

When the port is in the force authorized, force unauthorized, unauthorized, or shutdown state, it is placed in the configured access VLAN.

If an IEEE 802.1x port is authenticated and put in the RADIUS-server assigned VLAN, any change to the port access VLAN configuration does not take effect.

The IEEE 802.1x authentication with VLAN assignment feature is not supported on trunk ports, dynamic ports, or with dynamic-access port assignment through a VLAN Membership Policy Server (VMPS).

To configure VLAN assignment you need to perform these tasks:

- Enable AAA authorization.
- Enable IEEE 802.1x authentication (the VLAN assignment feature is automatically enabled when you configure IEEE 802.1x authentication on an access port).
- Assign vendor-specific tunnel attributes in the RADIUS server. The RADIUS server must return these attributes to the switch:
  - [64] Tunnel-Type = VLAN
  - [65] Tunnel-Medium-Type = IEEE 802
  - [81] Tunnel-Private-Group-ID = VLAN name or VLAN ID

  Attribute [64] must contain the value *VLAN* (type 13). Attribute [65] must contain the value *IEEE 802* (type 6). Attribute [81] specifies the *VLAN name* or *VLAN ID* assigned to the IEEE 802.1x-authenticated user.

For examples of tunnel attributes, see the "Configuring the Switch to Use Vendor-Specific RADIUS Attributes" section on page 7-29.

Using IEEE 802.1x Authentication with Per-User ACLs

You can enable per-user access control lists (ACLs) to provide different levels of network access and service to an IEEE 802.1x-authenticated user. When the RADIUS server authenticates a user connected to an IEEE 802.1x port, it retrieves the ACL attributes based on the user identity and sends them to the switch. The switch applies the attributes to the IEEE 802.1x port for the duration of the user session. The

switch removes the per-user ACL configuration when the session is over, if authentication fails, or if a link-down condition occurs. The switch does not save RADIUS-specified ACLs in the running configuration. When the port is unauthorized, the switch removes the ACL from the port.

You can configure only one type of per-user ACLs on a switch port: router ACLs or port ACLs. Router ACLs apply to Layer 3 interfaces, and port ACLs apply to Layer 2 interfaces. If a port is configured with a port-based ACL, the switch rejects any attempt to configure a router-based ACL on the same port. However, if a port is configured with a router-based ACL and then a port-based ACL, the port-based ACL overwrites the router ACL. To avoid configuration conflicts, you should carefully plan the user profiles stored on the RADIUS server.

RADIUS supports per-user attributes, including vendor-specific attributes. These vendor-specific attributes (VSAs) are in octet-string format and are passed to the switch during the authentication process. The VSAs used for per-user ACLs are `inacl#<n>` for ingress direction.

Note Port-based ACLs are only supported in the ingress direction.

Use only extended ACL syntax style to define the per-user configuration stored on the RADIUS server. When the definitions are passed from the RADIUS server, they are created by using the extended naming convention. However, if you use the Filter-Id attribute, it can point to a standard ACL.

You can use the Filter-Id attribute to specify an inbound or outbound ACL that is already configured on the switch. The attribute contains the ACL number followed by *.in* or *.out* for ingress filtering or egress filtering. If the RADIUS server does not allow *.in* or *.out* syntax, the access list is applied to the outbound ACL by default. Because of limited support of Cisco IOS access lists on the switch, the Filter-Id attribute is supported only for IP ACLs numbered 1 to 199 and 1300 to 2699 (IP standard and IP extended ACLs).

Only one IEEE 802.1x-authenticated user is supported on a port. If the multiple-hosts mode is enabled on the port, the per-user ACL attribute is disabled for the associated port.

The maximum size of the per-user ACL is 4000 ACSII characters.

For examples of vendor-specific attributes, see the "Configuring the Switch to Use Vendor-Specific RADIUS Attributes" section on page 7-29. For more information about configuring ACLs, see Chapter 29, "Configuring Network Security with ACLs."

To configure per-user ACLs, you need to perform these tasks:

- Enable AAA authentication
- Enable AAA authorization by using the **network** keyword to allow interface configuration from the RADIUS server
- Enable IEEE 802.1x authentication
- Configure the user profile and VSAs on the RADIUS server
- Configure the IEEE 802.1x port for single-host mode

Using IEEE 802.1x Authentication with Guest VLAN

You can configure a guest VLAN for each IEEE 802.1x port on the switch to provide limited services to clients, such as downloading the IEEE 802.1x client. These clients might be upgrading their system for IEEE 802.1x authentication, and some hosts, such as Windows 98 systems, might not be IEEE 802.1x-capable.

When you enable a guest VLAN on an IEEE 802.1x port, the switch assigns clients to a guest VLAN when the switch does not receive a response to its EAP request/identity frame or when EAPOL packets are not sent by the client.

With Cisco IOS Release 12.2(25)SE and later, the switch maintains the EAPOL packet history. If an EAPOL packet is detected on the interface during the lifetime of the link, the switch determines that the device connected to that interface is an IEEE 802.1x-capable supplicant, and the interface does not change to the guest VLAN state. EAPOL history is cleared if the interface link status goes down. If no EAPOL packet is detected on the interface, the interface changes to the guest VLAN state.

Before Cisco IOS Release 12.2(25)SE, the switch did not maintain the EAPOL packet history and allowed clients that failed authentication access to the guest VLAN, regardless of whether EAPOL packets had been detected on the interface. You can enable this behavior by using the dot1x guest-vlan supplicant global configuration command.

In Cisco IOS Release 12.2(25)SEE and later, if devices send EAPOL packets to the switch during the lifetime of the link, the switch no longer allows clients that fail authentication access to the guest VLAN.

If the switch is trying to authorize an IEEE 802.1x-capable voice device and the AAA server is unavailable, the authorization attempt fails, but the detection of the EAPOL packet is saved in the EAPOL history. When the AAA server becomes available, the switch authorizes the voice device. However, the switch no longer allows other devices access to the guest VLAN. To prevent this situation, use one of these command sequences:

- Enter the dot1x guest-vlan supplicant global configuration command to allow access to the guest VLAN.
- Enter the shutdown interface configuration command followed by the no shutdown interface configuration command to restart the port.

Note If an EAPOL packet is detected after the interface has changed to the guest VLAN, the interface reverts to an unauthorized state, and IEEE 802.1x authentication restarts.

Any number of IEEE 802.1x-incapable clients are allowed access when the switch port is moved to the guest VLAN. If an IEEE 802.1x-capable client joins the same port on which the guest VLAN is configured, the port is put into the unauthorized state in the user-configured access VLAN, and authentication is restarted.

Guest VLANs are supported on IEEE 802.1x ports in single-host or multiple-hosts mode.

You can configure any active VLAN except an RSPAN VLAN, a private VLAN, or a voice VLAN as an IEEE 802.1x guest VLAN. The guest VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access port-based authentication:guest VLAN:configuration guidelinesports..

The switch supports *MAC authentication bypass* in Cisco IOS Release 12.2(25)SEE and later. When MAC authentication bypass is enabled on an IEEE 802.1x port, the switch can authorize clients based on the client MAC address when IEEE 802.1x authentication times out while waiting for an EAPOL message exchange. After detecting a client on an IEEE 802.1x port, the switch waits for an Ethernet packet from the client. The switch sends the authentication server a RADIUS-access/request frame with a username and password based on the MAC address. If authorization succeeds, the switch grants the client access to the network. If authorization fails, the switch assigns the port to the guest VLAN if one is specified. For more information, see the"Using IEEE 802.1x Authentication with MAC Authentication Bypass" section on page 8-17.

For configuration steps, see the "Configuring a Guest VLAN" section on page 8-32.

Using IEEE 802.1x Authentication with Restricted VLAN

You can configure a restricted VLAN for each IEEE 802.1x port on a switch to provide limited services to clients that cannot access the guest VLAN. These clients are IEEE 802.1x-compliant and cannot access another VLAN because they fail the authentication process. A restricted VLAN allows users without valid credentials in an authentication server (typically, visitors to an enterprise) to access a limited set of services. The administrator can control the services available to the restricted VLAN.

Note You can configure a VLAN to be both the guest VLAN and the restricted VLAN if you want to provide the same services to both types of users.

Without this feature, the client attempts and fails authentication indefinitely, and the switch port remains in the spanning-tree blocking state. With this feature, you can configure the switch port to be in the restricted VLAN after a specified number of authentication attempts (the default value is 3 attempts).

The authenticator keeps a count of failed authentication attempts for the client. When this count exceeds the configured maximum number of authentication attempts, the port moves to the restricted VLAN. The failed attempt count is incremented when RADIUS replies with either an *EAP failure* or an empty response that contains no EAP packet. When the port moves into the restricted VLAN, the failed attempt counter resets.

Users who fail authentication remain in the restricted VLAN until the next re-authentication attempt. A port in the restricted VLAN tries to re-authenticate at configured intervals (the default is 60 seconds). If re-authentication fails, the port remains in the restricted VLAN. If re-authentication is successful, the port moves to either the configured VLAN or to a VLAN sent by the RADIUS server. You can disable re-authentication. If you do this, the only way to start the authentication process again is for the port to receive a *link dow*n or *EAP logoff* event. We recommend that you keep re-authentication enabled if a client might connect through a hub. When a client disconnects from the hub, the port might not receive the *link down* or *EAP logoff* event.

After a port moves to the restricted VLAN, it sends a simulated EAP success message to the client. This prevents clients from attempting authentication indefinitely. Some clients (for example, devices running Windows XP) cannot implement DHCP without EAP success.

Restricted VLANs are supported only on IEEE 802.1x ports in single-host mode and on Layer 2 ports.

You can configure any active VLAN except an RSPAN VLAN or a voice VLAN as an IEEE 802.1x restricted VLAN. The restricted VLAN feature is not supported on trunk ports; it is supported only on access ports.

This feature works with port security. As soon as the port is authorized, a MAC address is provided to port security. If port security does not permit the MAC address or if the maximum secure address count is reached, the port becomes unauthorized and error disabled.

Other port security features such as Dynamic ARP Inspection, DHCP snooping, and IP source guard can be configured independently on a restricted VLAN.

For more information, see the "Configuring a Restricted VLAN" section on page 8-33.

Using IEEE 802.1x Authentication with Inaccessible Authentication Bypass

In Cisco IOS Release 12.2(25)SEE and later, when the switch cannot reach the configured RADIUS servers and hosts cannot be authenticated, you can configure the switch to allow network access to the hosts connected to *critical* ports. A critical port is enabled for the inaccessible authentication bypass feature, also referred to as *critical authentication* or the *AAA fail policy.*

When this feature is enabled, the switch checks the status of the configured RADIUS servers whenever the switch tries to authenticate a host connected to a critical port. If a server is available, the switch can authenticate the host. However, if all the RADIUS servers are unavailable, the switch grants network access to the host and puts the port in the critical-authentication state, which is a special case of the authentication state.

The behavior of the inaccessible authentication bypass feature depends on the authorization state of the port:

- If the port is unauthorized when a host connected to a critical port tries to authenticate and all servers are unavailable, the switch puts the port in the critical-authentication state in the RADIUS-configured or user-specified access VLAN.
- If the port is already authorized and re-authentication occurs, the switch puts the critical port in the critical-authentication state in the current VLAN, which might be the one previously assigned by the RADIUS server.
- If the RADIUS server becomes unavailable during an authentication exchange, the current exchanges times out, and the switch puts the critical port in the critical-authentication state during the next authentication attempt.

When a RADIUS server that can authenticate the host is available, all critical ports in the critical-authentication state are automatically re-authenticated.

Inaccessible authentication bypass interacts with these features:

- Guest VLAN—Inaccessible authentication bypass is compatible with guest VLAN. When a guest VLAN is enabled on IEEE 8021.x port, the features interact as follows:
  - If at least one RADIUS server is available, the switch assigns a client to a guest VLAN when the switch does not receive a response to its EAP request/identity frame or when EAPOL packets are not sent by the client.
  - If all the RADIUS servers are not available and the client is connected to a critical port, the switch authenticates the client and puts the critical port in the critical-authentication state in the RADIUS-configured or user-specified access VLAN.
  - If all the RADIUS servers are not available and the client is not connected to a critical port, the switch might not assign clients to the guest VLAN if one is configured.
  - If all the RADIUS servers are not available and if a client is connected to a critical port and was previously assigned to a guest VLAN, the switch keeps the port in the guest VLAN.

- Restricted VLAN—If the port is already authorized in a restricted VLAN and the RADIUS servers are unavailable, the switch puts the critical port in the critical-authentication state in the restricted VLAN.
- IEEE 802.1x accounting—Accounting is not affected if the RADIUS servers are unavailable.
- Private VLAN—You can configure inaccessible authentication bypass on private VLAN host port. The access VLAN must be a secondary private VLAN.
- Voice VLAN—Inaccessible authentication bypass is compatible with voice VLAN, but the RADIUS-configured or user-specified access VLAN and the voice VLAN must be different.
- Remote Switched Port Analyzer (RSPAN)—Do not configure an RSPAN VLAN as the RADIUS-configured or user-specified access VLAN for inaccessible authentication bypass.

Using IEEE 802.1x Authentication with Voice VLAN Ports

A voice VLAN port is a special access port associated with two VLAN identifiers:

- VVID to carry voice traffic to and from the IP phone. The VVID is used to configure the IP phone connected to the port.
- PVID to carry the data traffic to and from the workstation connected to the switch through the IP phone. The PVID is the native VLAN of the port.

In single-host mode, only the IP phone is allowed on the voice VLAN. In multiple-hosts mode, additional clients can send traffic on the voice VLAN after a supplicant is authenticated on the PVID. When multiple-hosts mode is enabled, the supplicant authentication affects both the PVID and the VVID.

A voice VLAN port becomes active when there is a link, and the device MAC address appears after the first CDP message from the IP phone. Cisco IP phones do not relay CDP messages from other devices. As a result, if several Cisco IP phones are connected in series, the switch recognizes only the one directly connected to it. When IEEE 802.1x authentication is enabled on a voice VLAN port, the switch drops packets from unrecognized Cisco IP phones more than one hop away.

When IEEE 802.1x authentication is enabled on a port, you cannot configure a port VLAN that is equal to a voice VLAN.

For more information about voice VLANs, see Chapter 13, “Configuring Voice VLAN.”

Using IEEE 802.1x Authentication with Port Security

You can configure an IEEE 802.1x port with port security in either single-host or multiple-hosts mode. (You must also configure port security on the port by using the **switchport port-security** interface configuration command.) When you enable port security and IEEE 802.1x on a port, IEEE 802.1x authentication authenticates the port, and port security manages network access for all MAC addresses, including that of the client. You can then limit the number or group of clients that can access the network through an IEEE 802.1x port.

These are some examples of the interaction between IEEE 802.1x authentication and port security on the switch:

- When a client is authenticated, and the port security table is not full, the client's MAC address is added to the port security list of secure hosts. The port then proceeds to come up normally.

  When a client is authenticated and manually configured for port security, it is guaranteed an entry in the secure host table (unless port security static aging has been enabled).

  A security violation occurs if the client is authenticated, but port security table is full. This can happen if the maximum number of secure hosts has been statically configured, or if the client ages out of the secure host table. If the client's address is aged out, its place in the secure host table can be taken by another host.

  The port security violation modes determine the action for security violations. For more information, see the "Security Violations" section on page 21-8.

- When you manually remove an IEEE 802.1x client address from the port security table by using the **no switchport port-security mac-address** *mac-address* interface configuration command, you should re-authenticate the IEEE 802.1x client by using the **dot1x re-authenticate interface** *interface-id* privileged EXEC command.
- When an IEEE 802.1x client logs off, the port changes to an unauthenticated state, and all dynamic entries in the secure host table are cleared, including the entry for the client. Normal authentication then takes place.
- If the port is administratively shut down, the port becomes unauthenticated, and all dynamic entries are removed from the secure host table.
- Port security and a voice VLAN can be configured simultaneously on an IEEE 802.1x port that is in either single-host or multiple-hosts mode. Port security applies to both the voice VLAN identifier (VVID) and the port VLAN identifier (PVID).

For more information about enabling port security on your switch, see the "Configuring Port Security" section on page 21-7.

Using IEEE 802.1x Authentication with Wake-on-LAN

The IEEE 802.1x wake-on-LAN (WoL) feature allows dormant PCs to be powered when the switch receives a specific Ethernet frame, known as the *magic packet*. You can use this feature in environments where administrators need to connect to systems that have been powered down. This feature is also known as the *unidirectional controlled port* in the IEEE 802.1x standard.

When a host that uses WoL are attached through an IEEE 802.1x port and the host powers off, the IEEE 802.1x port becomes unauthorized. The port can only receive and send EAPOL packets, and WoL magic packets cannot reach the host. When the PC is powered off, it is not authorized, and the switch port is not opened.

When the switch uses IEEE 802.1x authentication with WoL, the switch forwards traffic to unauthorized IEEE 802.1x ports, including magic packets. While the port is unauthorized, the switch continues to block ingress traffic other than EAPOL packets. The host can receive packets but cannot send packets to other devices in the network.

Note If PortFast is not enabled on the port, the port is forced to the bidirectional state.

When you configure a port as unidirectional by using the **dot1x control-direction in** interface configuration command, the port changes to the spanning-tree forwarding state. The port can send packets to the host but cannot receive packets from the host.

When you configure a port as bidirectional by using the **dot1x control-direction both** interface configuration command, the port is access-controlled in both directions. The port does not receive packets from or send packets to the host.

Using IEEE 802.1x Authentication with MAC Authentication Bypass

You can configure the switch to authorize clients based on the client MAC address (see Figure 8-2 on page 8-4) by using the MAC authentication bypass feature. For example, you can enable this feature on IEEE 802.1x ports connected to devices such as printers.

If IEEE 802.1x authentication times out while waiting for an EAPOL response from the client, the switch tries to authorize the client by using MAC authentication bypass.

When the MAC authentication bypass feature is enabled on an IEEE 802.1x port, the switch uses the MAC address as the client identity. The authentication server has a database of client MAC addresses that are allowed network access. After detecting a client on an IEEE 802.1x port, the switch waits for an Ethernet packet from the client. The switch sends the authentication server a RADIUS-access/request frame with a username and password based on the MAC address. If authorization succeeds, the switch grants the client access to the network. If authorization fails, the switch assigns the port to the guest VLAN if one is configured.

If an EAPOL packet is detected on the interface during the lifetime of the link, the switch determines that the device connected to that interface is an IEEE 802.1x-capable supplicant and uses IEEE 802.1x authentication (not MAC authentication bypass) to authorize the interface. EAPOL history is cleared if the interface link status goes down.

If the switch already authorized a port by using MAC authentication bypass and detects an IEEE 802.1x supplicant, the switch does not unauthorize the client connected to the port. When re-authentication occurs, the switch uses IEEE 802.1x authentication as the preferred re-authentication process if the previous session ended because the Termination-Action RADIUS attribute value is DEFAULT.

Clients that were authorized with MAC authentication bypass can be re-authenticated. The re-authentication process is the same as that for clients that were authenticated with IEEE 802.1x authentication. During re-authentication, the port remains in the previously assigned VLAN. If re-authentication is successful, the switch keeps the port in the same VLAN. If re-authentication fails, the switch assigns the port to the guest VLAN, if one is configured.

If re-authentication is based on the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute [29]) and if the Termination-Action RADIUS attribute (Attribute [29]) action is *Initialize,* (the attribute value is *DEFAULT*), the MAC authentication bypass session ends, and connectivity is lost during re-authentication. If MAC authentication bypass is enabled and the IEEE 802.1x authentication times out, the switch uses the MAC authentication bypass feature to initiate re-authorization. For more information about these AV pairs, see RFC 3580, “IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines.”

MAC authentication bypass interacts with the features:

- IEEE 802.1x authentication—You can enable MAC authentication bypass only if IEEE 802.1x authentication is enabled on the port.
- Guest VLAN—If a client has an invalid MAC address identity, the switch assigns the client to a guest VLAN if one is configured.

- Restricted VLAN—This feature is not supported when the client connected to an IEEE 802.lx port is authenticated with MAC authentication bypass.
- Port security—See the "Using IEEE 802.1x Authentication with Port Security" section on page 8-15.
- Voice VLAN—See the "Using IEEE 802.1x Authentication with Voice VLAN Ports" section on page 8-15.
- VLAN Membership Policy Server (VMPS)—IEEE802.1x and VMPS are mutually exclusive.
- Private VLAN—You can assign a client to a private VLAN.
- Network admission control (NAC) Layer 2 IP validation—This feature takes effect after an IEEE 802.1x port is authenticated with MAC authentication bypass, including hosts in the exception list.

Network Admission Control Layer 2 IEEE 802.1x Validation

In Cisco IOS Release 12.2(25)SED and later, the switch supports the Network Admission Control (NAC) Layer 2 IEEE 802.1x validation, which checks the antivirus condition or *posture* of endpoint systems or clients before granting the devices network access. With NAC Layer 2 IEEE 802.1x validation, you can do these tasks:

- Download the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute[29]) from the authentication server.
- Set the number of seconds between re-authentication attempts as the value of the Session-Timeout RADIUS attribute (Attribute[27]) and get an access policy against the client from the RADIUS server.
- Set the action to be taken when the switch tries to re-authenticate the client by using the Termination-Action RADIUS attribute (Attribute[29]). If the value is *DEFAULT* or is not set, the session ends. If the value is RADIUS-Request, the re-authentication process starts.
- View the NAC posture token, which shows the posture of the client, by using the **show dot1x** privileged EXEC command.
- Configure secondary private VLANs as guest VLANs.

Configuring NAC Layer 2 IEEE 802.1x validation is similar to configuring IEEE 802.1x port-based authentication except that you must configure a posture token on the RADIUS server. For information about configuring NAC Layer 2 IEEE 802.1x validation, see the "Configuring NAC Layer 2 IEEE 802.1x Validation" section on page 8-38 and the "Enabling Periodic Re-Authentication" section on page 8-27.

For more information about NAC, see the *Network Admission Control Software Configuration Guide*.

Web Authentication with Automatic MAC Check

You can use web authentication with automatic MAC check to authenticate a client that does not support IEEE 802.1x or web browser functionality. This allows end hosts, such as printers, to automatically authenticate by using the MAC address without any additional required configuration.

You can configure a port to use only web authentication. You can also configure the port to first try and use IEEE 802.1x authentication and then to use web authorization if the client does not support IEEE 802.1x authentication.

Note The interoperability of web authentication (with automatic MAC check) and IEEE 802.1x MAC authentication configured on different ports of the same switch is not supported.

Web authentication requires two Cisco Attribute-Value (AV) pair attributes:

- The first attribute, `priv-lvl=15`, must always be set to *15*. This sets the privilege level of the user who is logging into the switch.
- The second attribute is an access list to be applied for web authenticated hosts. The syntax is similar to IEEE 802.1X per-user ACLs. However, instead of `ip:inacl`, this attribute must begin with `proxyacl`, and the `source` field in each entry must be `any`. (After authentication, the client IP address replaces the `any` field when the ACL is applied.)

  For example:

```
proxyacl# 10=permit ip any 10.0.0.0 255.0.0.0
proxyacl# 20=permit ip any 11.1.0.0 255.255.0.0
proxyacl# 30=permit udp any any eq syslog
proxyacl# 40=permit udp any any eq tftp
```

Note The *proxyacl* entry determines the type of allowed network access.

For more information, see the "Configuring Web Authentication" section on page 8-39.

Configuring IEEE 802.1x Authentication

These sections describe how to configure IEEE 802.1x port-based authentication on your switch:

- Default IEEE 802.1x Authentication Configuration, page 8-20
- IEEE 802.1x Authentication Configuration Guidelines, page 8-21
- Upgrading from a Previous Software Release, page 8-23
- Configuring IEEE 802.1x Authentication, page 8-24 (required)
- Configuring the Switch-to-RADIUS-Server Communication, page 8-25 (required)
- Configuring the Host Mode, page 8-27 (optional)
- Enabling Periodic Re-Authentication, page 8-27 (optional)
- Manually Re-Authenticating a Client Connected to a Port, page 8-28 (optional)
- Changing the Quiet Period, page 8-28 (optional)
- Changing the Switch-to-Client Retransmission Time, page 8-29 (optional)
- Setting the Switch-to-Client Frame-Retransmission Number, page 8-30 (optional)
- Setting the Re-Authentication Number, page 8-30 (optional)
- Configuring IEEE 802.1x Accounting, page 8-31 (optional)
- Configuring a Guest VLAN, page 8-32 (optional)
- Configuring a Restricted VLAN, page 8-33 (optional)

- Configuring the Inaccessible Authentication Bypass Feature, page 8-34
- Configuring IEEE 802.1x Authentication with WoL, page 8-37
- Configuring MAC Authentication Bypass, page 8-37
- Configuring NAC Layer 2 IEEE 802.1x Validation, page 8-38
- Configuring Web Authentication, page 8-39
- Disabling IEEE 802.1x on the Port, page 8-41
- Resetting the IEEE 802.1x Configuration to the Default Values, page 8-42 (optional)

Default IEEE 802.1x Authentication Configuration

Table 8-2 shows the default IEEE 802.1x authentication configuration.

Table 8-2 Default IEEE 802.1x Authentication Configuration

Feature	Default Setting
Switch IEEE 802.1x enable state	Disabled.
Per-interface IEEE 802.1x enable state	Disabled (force-authorized). The port sends and receives normal traffic without IEEE 802.1x-based authentication of the client.
AAA	Disabled.
RADIUS server • IP address • UDP authentication port • Key	 • None specified. • 1812. • None specified.
Host mode	Single-host mode.
Control direction	Bidirectional control.
Periodic re-authentication	Disabled.
Number of seconds between re-authentication attempts	3600 seconds.
Re-authentication number	2 times (number of times that the switch restarts the authentication process before the port changes to the unauthorized state).
Quiet period	60 seconds (number of seconds that the switch remains in the quiet state following a failed authentication exchange with the client).
Retransmission time	30 seconds (number of seconds that the switch should wait for a response to an EAP request/identity frame from the client before resending the request).
Maximum retransmission number	2 times (number of times that the switch will send an EAP-request/identity frame before restarting the authentication process).

Table 8-2 Default IEEE 802.1x Authentication Configuration (continued)

Feature	Default Setting
Client timeout period	30 seconds (when relaying a request from the authentication server to the client, the amount of time the switch waits for a response before resending the request to the client.
Authentication server timeout period	30 seconds (when relaying a response from the client to the authentication server, the amount of time the switch waits for a reply before resending the response to the server.) You can change this timeout period by using the **dot1x timeout server-timeout** interface configuration command.
Guest VLAN	None specified.
Inaccessible authentication bypass	Disabled.
Restricted VLAN	None specified.
Authenticator (switch) mode	None specified.
MAC authentication bypass	Disabled.

IEEE 802.1x Authentication Configuration Guidelines

These section has configuration guidelines for these features:

- IEEE 802.1x Authentication, page 8-21
- VLAN Assignment, Guest VLAN, Restricted VLAN, and Inaccessible Authentication Bypass, page 8-22
- MAC Authentication Bypass, page 8-23

IEEE 802.1x Authentication

These are the IEEE 802.1x authentication configuration guidelines:

- When IEEE 802.1x authentication is enabled, ports are authenticated before any other Layer 2 or Layer 3 features are enabled.
- The IEEE 802.1x protocol is supported on Layer 2 static-access ports, voice VLAN ports, and Layer 3 routed ports, but it is not supported on these port types:
  - Trunk port—If you try to enable IEEE 802.1x authentication on a trunk port, an error message appears, and IEEE 802.1x authentication is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to trunk, the port mode is not changed.
  - Dynamic ports—A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable IEEE 802.1x authentication on a dynamic port, an error message appears, and IEEE 802.1x authentication is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to dynamic, the port mode is not changed.

- Dynamic-access ports—If you try to enable IEEE 802.1x authentication on a dynamic-access (VLAN Query Protocol [VQP]) port, an error message appears, and IEEE 802.1x authentication is not enabled. If you try to change an IEEE 802.1x-enabled port to dynamic VLAN assignment, an error message appears, and the VLAN configuration is not changed.
- EtherChannel ports—Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an IEEE 802.1x port. If you try to enable IEEE 802.1x authentication on an EtherChannel port, an error message appears, and IEEE 802.1x authentication is not enabled.

  Note In software releases earlier than Cisco IOS Release 12.2(25)SE, if IEEE 802.1x authentication is enabled on a not-yet active port of an EtherChannel, the port does not join the EtherChannel.

- Switched Port Analyzer (SPAN) and Remote SPAN (RSPAN) destination ports—You cannot enable IEEE 802.1x authentication on a port that is a SPAN or RSPAN destination port or that is an RSPAN reflector port. However, you can enable IEEE 802.1x authentication on a SPAN or RSPAN source port.

- Before globally enabling IEEE 802.1x authentication on a switch by entering the **dot1x system-auth-control** global configuration command, remove the EtherChannel configuration from the interfaces on which IEEE 802.1x authentication and EtherChannel are configured.
- If you are using a device running the Cisco Access Control Server (ACS) application for IEEE 802.1x authentication with EAP-Transparent LAN Services (TLS) and EAP-MD5 and your switch is running Cisco IOS Release 12.1(14)EA1, make sure that the device is running ACS Version 3.2.1 or later.

VLAN Assignment, Guest VLAN, Restricted VLAN, and Inaccessible Authentication Bypass

These are the configuration guidelines for VLAN assignment, guest VLAN, restricted VLAN, and inaccessible authentication bypass:

- When IEEE 802.1x authentication is enabled on a port, you cannot configure a port VLAN that is equal to a voice VLAN.
- The IEEE 802.1x authentication with VLAN assignment feature is not supported on trunk ports, dynamic ports, or with dynamic-access port assignment through a VMPS.
- You can configure IEEE 802.1x authentication on a private-VLAN port, but do not configure IEEE 802.1x authentication with port security, voice VLAN, guest VLAN, restricted VLAN, or a per-user ACL on private-VLAN ports.
- You can configure any VLAN, except an RSPAN VLAN or a voice VLAN, as an IEEE 802.1x guest VLAN. The guest VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.
- After you configure a guest VLAN for an IEEE 802.1x port to which a DHCP client is connected, you might need to get a host IP address from a DHCP server. You can also change the settings for restarting the IEEE 802.1x authentication process on the switch before the DHCP process on the client times out and tries to get a host IP address from the DHCP server. Decrease the settings for the IEEE 802.1x authentication process (**dot1x timeout quiet-period** and **dot1x timeout tx-period** interface configuration commands). The amount to decrease the settings depends on the connected IEEE 802.1x client type.

- When configuring the inaccessible authentication bypass feature, follow these guidelines:
  - The feature is supported on IEEE 802.1x port in single-host mode and multihosts mode.
  - If the client is running Windows XP and the port to which the client is connected is in the critical-authentication state, Windows XP might report that the interface is not authenticated.
  - If the Windows XP client is configured for DHCP and has an IP address from the DHCP server, receiving an EAP-Success message on a critical port might not re-initiate the DHCP configuration process.
  - You can configure the inaccessible authentication bypass feature and the restricted VLAN on an IEEE 802.1x port. If the switch tries to re-authenticate a critical port in a restricted VLAN and all the RADIUS servers are unavailable, switch changes the port state to the critical authentication state and remains in the restricted VLAN.
  - You can configure the inaccessible bypass feature and port security on the same switch port.
- You can configure any VLAN except an RSPAN VLAN or a voice VLAN as an IEEE 802.1x authentication restricted VLAN. The restricted VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

MAC Authentication Bypass

These are the MAC authentication bypass configuration guidelines:

- Unless otherwise stated, the MAC authentication bypass guidelines are the same as the IEEE 802.1x authentication guidelines. For more information, see the "IEEE 802.1x Authentication" section on page 8-21.
- If you disable MAC authentication bypass from a port after the port has been authorized with its MAC address, the port state is not affected.
- If the port is in the unauthorized state and the client MAC address is not the authentication-server database, the port remains in the unauthorized state. However, if the client MAC address is added to the database, the switch can use MAC authentication bypass to re-authorize the port.
- If the port is in the authorized state, the port remains in this state until re-authorization occurs.

Upgrading from a Previous Software Release

In Cisco IOS Release 12.1(14)EA1, the implementation for IEEE 802.1x authentication changed from the previous release. Some global configuration commands became interface configuration commands, and new commands were added.

If you have IEEE 802.1x authentication configured on the switch and you upgrade to Cisco IOS Release 12.1(14)EA1 or later, the configuration file will not contain the new commands, and IEEE 802.1x authentication will not operate. After the upgrade is complete, make sure to globally enable IEEE 802.1x authentication by using the **dot1x system-auth-control** global configuration command. If IEEE 802.1x authentication was running in multiple-hosts mode on an interface in the previous release, make sure to reconfigure it by using the **dot1x host-mode multi-host** interface configuration command.

In Cisco IOS Release 12.2(25)SEE, the implementation for IEEE 802.1x authentication changed from the previous releases. When IEEE 802.1x authentication is enabled, information about Port Fast is no longer added to the configuration and this information appears in the running configuration:

```
dot1x pae authenticator
```

Configuring IEEE 802.1x Authentication

To configure IEEE 802.1x port-based authentication, you must enable AAA and specify the authentication method list. A method list describes the sequence and authentication methods to be queried to authenticate a user.

The software uses the first method listed to authenticate users. If that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle, the authentication process stops, and no other authentication methods are attempted.

To allow per-user ACLs or VLAN assignment, you must enable AAA authorization to configure the switch for all network-related service requests.

This is the IEEE 802.1x AAA process:

Step 1 A user connects to a port on the switch.

Step 2 Authentication is performed.

Step 3 VLAN assignment is enabled, as appropriate, based on the RADIUS server configuration.

Step 4 The switch sends a start message to an accounting server.

Step 5 Re-authentication is performed, as necessary.

Step 6 The switch sends an interim accounting update to the accounting server that is based on the result of re-authentication.

Step 7 The user disconnects from the port.

Step 8 The switch sends a stop message to the accounting server.

Beginning in privileged EXEC mode, follow these steps to configure IEEE 802.1x port-based authentication. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA.
Step 3	**aaa authentication dot1x** {**default**} *method1*	Create an IEEE 802.1x authentication method list. To create a default list that is used when a named list is *not* specified in the **authentication** command, use the **default** keyword followed by the method that is to be used in default situations. The default method list is automatically applied to all ports. For *method1*, enter the **group radius** keyword to use the list of all RADIUS servers for authentication. **Note** Though other keywords are visible in the command-line help string, only the **default** and **group radius** keywords are supported.
Step 4	**dot1x system-auth-control**	Enable IEEE 802.1x authentication globally on the switch.

	Command	Purpose
Step 5	**aaa authorization network** {**default**} **group radius**	(Optional) Configure the switch for user RADIUS authorization for all network-related service requests, such as per-user ACLs or VLAN assignment. To configure per-user ACLs, single-host mode must be enabled. This setting is the default.
Step 6	**radius-server host** *ip-address*	(Optional) Specify the IP address of the RADIUS server.
Step 7	**radius-server key** *string*	(Optional) Specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.
Step 8	**interface** *interface-id*	Specify the port connected to the client that is to be enabled for IEEE 802.1x authentication, and enter interface configuration mode.
Step 9	**switchport mode access**	(Optional) Set the port to access mode only if you configured the RADIUS server in Step 6 and Step 7.
Step 10	**dot1x port-control auto**	Enable IEEE 802.1x authentication on the interface. For feature interaction information, see the "IEEE 802.1x Authentication Configuration Guidelines" section on page 8-21.
Step 11	**end**	Return to privileged EXEC mode.
Step 12	**show dot1x**	Verify your entries. Check the Status column in the IEEE 802.1x Port Summary section of the display. An *enabled* status means the port-control value is set to either **auto** or to **force-unauthorized**.
Step 13	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable AAA, use the **no aaa new-model** global configuration command. To disable IEEE 802.1x AAA authentication, use the **no aaa authentication dot1x** {**default** | *list-name*} global configuration command. To disable IEEE 802.1x AAA authorization, use the **no aaa authorization** global configuration command. To disable IEEE 802.1x authentication on the switch, use the **no dot1x system-auth-control** global configuration command.

This example shows how to enable AAA and IEEE 802.1x authentication on a port:

```
Switch# configure terminal
Switch(config)# aaa new-model
Switch(config)# aaa authentication dot1x default group radius
Switch(config)# dot1x system-auth-control
Switch(config)# interface fastethernet0/1
Switch(config-if)# switchport mode access
Switch(config-if)# dot1x port-control auto
Switch(config-if)# end
```

Configuring the Switch-to-RADIUS-Server Communication

RADIUS security servers are identified by their host name or IP address, host name and specific UDP port numbers, or IP address and specific UDP port numbers. The combination of the IP address and UDP port number creates a unique identifier, which enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service—for example, authentication—the second host entry configured acts as the fail-over backup to the first one. The RADIUS host entries are tried in the order that they were configured.

Beginning in privileged EXEC mode, follow these steps to configure the RADIUS server parameters on the switch. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server host** {*hostname* | *ip-address*} **auth-port** *port-number* **key** *string*	Configure the RADIUS server parameters on the switch. For *hostname* | *ip-address,* specify the host name or IP address of the remote RADIUS server. For **auth-port** *port-number*, specify the UDP destination port for authentication requests. The default is 1812. For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. The key is a text string that must match the encryption key used on the RADIUS server. **Note** Always configure the key as the last item in the **radius-server host** command syntax because leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon. If you want to use multiple RADIUS servers, re-enter this command.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete the specified RADIUS server, use the **no radius-server host** {*hostname* | *ip-address*} global configuration command.

This example shows how to specify the server with IP address 172.20.39.46 as the RADIUS server, to use port 1612 as the authorization port, and to set the encryption key to *rad123*, matching the key on the RADIUS server:

```
Switch(config)# radius-server host 172.120.39.46 auth-port 1612 key rad123
```

You can globally configure the timeout, retransmission, and encryption key values for all RADIUS servers by using the **radius-server host** global configuration command. If you want to configure these options on a per-server basis, use the **radius-server timeout**, **radius-server retransmit**, and the **radius-server key** global configuration commands. For more information, see the "Configuring Settings for All RADIUS Servers" section on page 7-29.

You also need to configure some settings on the RADIUS server. These settings include the IP address of the switch and the key string to be shared by both the server and the switch. For more information, see the RADIUS server documentation.

Configuring the Host Mode

Beginning in privileged EXEC mode, follow these steps to allow multiple hosts (clients) on an IEEE 802.1x-authorized port that has the **dot1x port-control** interface configuration command set to **auto**. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to which multiple hosts are indirectly attached, and enter interface configuration mode.
Step 3	**dot1x host-mode multi-host**	Allow multiple hosts (clients) on an IEEE 802.1x-authorized port. Make sure that the **dot1x port-control** interface configuration command set is set to **auto** for the specified interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show dot1x interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable multiple hosts on the port, use the **no dot1x host-mode multi-host** interface configuration command.

This example shows how to enable IEEE 802.1x authentication and to allow multiple hosts:

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# dot1x port-control auto
Switch(config-if)# dot1x host-mode multi-host
```

Enabling Periodic Re-Authentication

You can enable periodic IEEE 802.1x client re-authentication and specify how often it occurs. If you do not specify a time period before enabling re-authentication, the number of seconds between re-authentication attempts is 3600.

Beginning in privileged EXEC mode, follow these steps to enable periodic re-authentication of the client and to configure the number of seconds between re-authentication attempts. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**dot1x reauthentication**	Enable periodic re-authentication of the client, which is disabled by default.

	Command	Purpose
Step 4	**dot1x timeout reauth-period** {*seconds* | **server**}	The keywords have these meanings: • *seconds*—Sets the number of seconds from 1 to 65535; the default is 3600 seconds. • **server**—Sets the number of seconds based on the value of the Session-Timeout RADIUS attribute (Attribute[27]) and Termination-Action RADIUS attribute (Attribute [29]). This command affects the behavior of the switch only if periodic re-authentication is enabled.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show dot1x interface** *interface-id*	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable periodic re-authentication, use the **no dot1x reauthentication** interface configuration command. To return to the default number of seconds between re-authentication attempts, use the **no dot1x timeout reauth-period** global configuration command.

This example shows how to enable periodic re-authentication and set the number of seconds between re-authentication attempts to 4000:

```
Switch(config-if)# dot1x reauthentication
Switch(config-if)# dot1x timeout reauth-period 4000
```

Manually Re-Authenticating a Client Connected to a Port

You can manually re-authenticate the client connected to a specific port at any time by entering the **dot1x re-authenticate interface** *interface-id* privileged EXEC command. This step is optional. If you want to enable or disable periodic re-authentication, see the "Enabling Periodic Re-Authentication" section on page 8-27.

This example shows how to manually re-authenticate the client connected to a port:

```
Switch# dot1x re-authenticate interface fastethernet0/1
```

Changing the Quiet Period

When the switch cannot authenticate the client, the switch remains idle for a set period of time, and then tries again. The idle time is determined by the quiet-period value. A failed authentication of the client might occur because the client provided an invalid password. You can provide a faster response time to the user by entering a smaller number than the default.

Beginning in privileged EXEC mode, follow these steps to change the quiet period. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.

	Command	Purpose
Step 3	**dot1x timeout quiet-period** *seconds*	Set the number of seconds that the switch remains in the quiet state following a failed authentication exchange with the client. The range is 1 to 65535 seconds; the default is 60.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show dot1x interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default quiet time, use the **no dot1x timeout quiet-period** interface configuration command.

This example shows how to set the quiet time on the switch to 30 seconds:

```
Switch(config-if)# dot1x timeout quiet-period 30
```

Changing the Switch-to-Client Retransmission Time

The client responds to the EAP-request/identity frame from the switch with an EAP-response/identity frame. If the switch does not receive this response, it waits a set period of time (known as the retransmission time) and then resends the frame.

Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to change the amount of time that the switch waits for client notification. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**dot1x timeout tx-period** *seconds*	Set the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before resending the request. The range is 1 to 65535 seconds; the default is 30.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show dot1x interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default retransmission time, use the **no dot1x timeout tx-period** interface configuration command.

This example shows how to set 60 as the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before resending the request:

```
Switch(config-if)# dot1x timeout tx-period 60
```

Setting the Switch-to-Client Frame-Retransmission Number

In addition to changing the switch-to-client retransmission time, you can change the number of times that the switch sends an EAP-request/identity frame (assuming no response is received) to the client before restarting the authentication process.

Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to set the switch-to-client frame-retransmission number. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**dot1x max-req** *count*	Set the number of times that the switch sends an EAP-request/identity frame to the client before restarting the authentication process. The range is 1 to 10; the default is 2.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show dot1x interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default retransmission number, use the **no dot1x max-req** interface configuration command.

This example shows how to set 5 as the number of times that the switch sends an EAP-request/identity request before restarting the authentication process:

```
Switch(config-if)# dot1x max-req 5
```

Setting the Re-Authentication Number

You can also change the number of times that the switch restarts the authentication process before the port changes to the unauthorized state.

Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to set the re-authentication number. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.

	Command	Purpose
Step 3	**dot1x max-reauth-req** *count*	Set the number of times that the switch restarts the authentication process before the port changes to the unauthorized state. The range is 1 to 10; the default is 2.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show dot1x interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default re-authentication number, use the **no dot1x max-reauth-req** interface configuration command.

This example shows how to set 4 as the number of times that the switch restarts the authentication process before the port changes to the unauthorized state:

```
Switch(config-if)# dot1x max-reauth-req 4
```

Configuring IEEE 802.1x Accounting

Enabling AAA system accounting with IEEE 802.1x accounting allows system reload events to be sent to the accounting RADIUS server for logging. The server can then infer that all active IEEE 802.1x sessions are closed.

Because RADIUS uses the unreliable UDP transport protocol, accounting messages might be lost due to poor network conditions. If the switch does not receive the accounting response message from the RADIUS server after a configurable number of retransmissions of an accounting request, this system message appears:

```
Accounting message %s for session %s failed to receive Accounting Response.
```

When the stop message is not sent successfully, this message appears:

```
00:09:55: %RADIUS-4-RADIUS_DEAD: RADIUS server 172.20.246.201:1645,1646 is not responding.
```

Note You must configure the RADIUS server to perform accounting tasks, such as logging start, stop, and interim-update messages and time stamps. To turn on these functions, enable logging of "Update/Watchdog packets from this AAA client" in your RADIUS server Network Configuration tab. Next, enable "CVS RADIUS Accounting" in your RADIUS server System Configuration tab.

Beginning in privileged EXEC mode, follow these steps to configure IEEE 802.1x accounting after AAA is enabled on your switch. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**aaa accounting dot1x default start-stop group radius**	Enable IEEE 802.1x accounting using the list of all RADIUS servers.
Step 4	**aaa accounting system default start-stop group radius**	(Optional) Enables system accounting (using the list of all RADIUS servers) and generates system accounting reload event messages when the switch reloads.

	Command	Purpose
Step 5	**end**	Return to privileged EXEc mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Use the **show radius statistics** privileged EXEC command to display the number of RADIUS messages that do not receive the accounting response message.

This example shows how to configure IEEE 802.1x accounting. The first command configures the RADIUS server, specifying 1813 as the UDP port for accounting:

```
Switch(config)# radius-server host 172.120.39.46 auth-port 1812 acct-port 1813 key rad123
Switch(config)# aaa accounting dot1x default start-stop group radius
Switch(config)# aaa accounting system default start-stop group radius
```

Configuring a Guest VLAN

When you configure a guest VLAN, clients that are not IEEE 802.1x-capable are put into the guest VLAN when the server does not receive a response to its EAPOL request/identity frame. Clients that are IEEE 802.1x-capable but fail authentication are not granted access to the network. The switch supports guest VLANs in single-host or multiple-hosts mode.

Note Depending on the switch configuration, assigning the client to a guest VLAN can take up to several minutes.

Beginning in privileged EXEC mode, follow these steps to configure a guest VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode. For the supported interface types, see the "IEEE 802.1x Authentication Configuration Guidelines" section on page 8-21.
Step 3	**switchport mode access**	Set the port to access mode.
Step 4	**dot1x port-control auto**	Enable IEEE 802.1x authentication on the port.
Step 5	**dot1x guest-vlan** *vlan-id*	Specify an active VLAN as an IEEE 802.1x guest VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN, or a voice VLAN as an IEEE 802.1x guest VLAN.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show dot1x interface** *interface-id*	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable and remove the guest VLAN, use the **no dot1x guest-vlan** interface configuration command. The port returns to the unauthorized state.

This example shows how to enable VLAN 9 as an IEEE 802.1x guest VLAN on a port:

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# dot1x guest-vlan 9
```

This example shows how to set 3 as the quiet time on the switch, to set 15 as the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before resending the request, and to enable VLAN 2 as an IEEE 802.1x guest VLAN when an IEEE 802.1x port is connected to a DHCP client:

```
Switch(config-if)# dot1x timeout quiet-period 3
Switch(config-if)# dot1x timeout tx-period 15
Switch(config-if)# dot1x guest-vlan 2
```

Configuring a Restricted VLAN

When you configure a restricted VLAN on a switch, clients that are IEEE 802.1x-compliant are moved into the restricted VLAN when the authentication server does not receive a valid username and password. The switch supports restricted VLANs only in single-host mode.

Beginning in privileged EXEC mode, follow these steps to configure a restricted VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode. For the supported port types, see the "IEEE 802.1x Authentication Configuration Guidelines" section on page 8-21.
Step 3	**switchport mode access** or **switchport mode private-vlan host**	Set the port to access mode, or Configure the port as a private-VLAN host port.
Step 4	**dot1x port-control auto**	Enable IEEE 802.1x authentication on the port.
Step 5	**dot1x auth-fail vlan** *vlan-id*	Specify an active VLAN as an IEEE 802.1x restricted VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN, a primary private VLAN or a voice VLAN as an IEEE 802.1x restricted VLAN.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show dot1x interface** *interface-id*	(Optional) Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable and remove the restricted VLAN, use the **no dot1x auth-fail vlan** interface configuration command. The port returns to the unauthorized state.

This example shows how to enable VLAN 2 as an IEEE 802.1x restricted VLAN:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# dot1x auth-fail vlan 2
```

You can configure the maximum number of authentication attempts allowed before a user is assigned to the restricted VLAN by using the **dot1x auth-fail max-attempts** interface configuration command. The range of allowable authentication attempts is 1 to 3. The default is 3 attempts.

Beginning in privileged EXEC mode, follow these steps to configure the maximum number of allowed authentication attempts. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access** or **switchport mode private-vlan host**	Set the port to access mode, or Configure the port as a private-VLAN host port.
Step 4	**dot1x port-control auto**	Enable IEEE 802.1x authentication on the port.
Step 5	**dot1x auth-fail vlan** *vlan-id*	Specify an active VLAN as an IEEE 802.1x restricted VLAN. The range is 1 to 4094. You can configure any active VLAN except an RSPAN VLAN or a voice VLAN as an IEEE 802.1x restricted VLAN.
Step 6	**dot1x auth-fail max-attempts** *max attempts*	Specify a number of authentication attempts to allow before a port moves to the restricted VLAN. The range is 1 to 3, and the default is 3.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show dot1x interface** *interface-id*	(Optional) Verify your entries.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default value, use the **no dot1x auth-fail max-attempts** interface configuration command.

This example shows how to set 2 as the number of authentication attempts allowed before the port moves to the restricted VLAN:

```
Switch(config-if)# dot1x auth-fail max-attempts 2
```

Configuring the Inaccessible Authentication Bypass Feature

You can configure the inaccessible bypass feature, also referred to as critical authentication or the AAA fail policy.

Beginning in privileged EXEC mode, follow these steps to configure the port as a critical port and enable the inaccessible authentication bypass feature. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**radius-server dead-criteria time** *time* **tries** *tries*	(Optional) Set the conditions that are used to decide when a RADIUS server is considered unavailable or *dead*. The range for *time* is from 1 to 120 seconds. The switch dynamically determines the default *seconds* value that is 10 to 60 seconds. The range for *tries* is from 1 to 100. The switch dynamically determines the default *tries* parameter that is 10 to 100.
Step 3	**radius-server deadtime** *minutes*	(Optional) Sets the number of minutes that a RADIUS server is not sent requests. The range is from 0 to 1440 minutes (24 hours). The default is 0 minutes.
Step 4	**radius-server host** *ip-address* [**acct-port** *udp-port*] [**auth-port** *udp-port*] [**key** *string*] [**test username** *name* [**idle-time** *time*] [**ignore-acct-port**] [**ignore-auth-port**]]	(Optional) Configure the RADIUS server parameters by using these keywords: • **acct-port** *udp-port*—Specify the UDP port for the RADIUS accounting server. The range for the UDP port number is from 0 to 65536. The default is 1646. • **auth-port** *udp-port*—Specify the UDP port for the RADIUS authentication server. The range for the UDP port number is from 0 to 65536. The default is 1645. **Note** You should configure the UDP port for the RADIUS accounting server and the UDP port for the RADIUS authentication server to nondefault values. • **key** *string*—Specify the authentication and encryption key for all RADIUS communication between the switch and the RADIUS daemon. **Note** You can also configure the authentication and encryption key by using the **radius-server key** {**0** *string* | **7** *string* | *string*} global configuration command. • **test username** *name*—Enable automated testing of the RADIUS server status, and specify the username to be used. • **idle-time** *time*—Set the interval of time in minutes after which the switch sends test packets to the server. The range is from 1 to 35791 minutes. The default is 60 minutes (1 hour). • **ignore-acct-port**—Disable testing on the RADIUS-server accounting port. • **ignore-auth-port**—Disable testing on the RADIUS-server authentication port.

	Command	Purpose
Step 5	dot1x critical {**eapol** | **recovery delay** *milliseconds*}	(Optional) Configure the parameters for inaccessible authentication bypass: **eapol**—Specify that the switch sends an EAPOL-Success message when the switch successfully authenticates the critical port. **recovery delay** *milliseconds*—Set the recovery delay period during which the switch waits to re-initialize a critical port when a RADIUS server that was unavailable becomes available. The range is from 1 to 10000 milliseconds. The default is 1000 milliseconds (a port can be re-initialized every second).
Step 6	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode. For the supported port types, see the "IEEE 802.1x Authentication Configuration Guidelines" section on page 8-21.
Step 7	**dot1x critical** [**recovery action reinitialize** | **vlan** *vlan-id*]	Enable the inaccessible authentication bypass feature, and use these keywords to configure the feature: • **recovery action reinitialize**—Enable the recovery feature, and specify that the recovery action is to authenticate the port when an authentication server is available. • **vlan** *vlan-id*—Specify the access VLAN to which the switch can assign a critical port. The range is from 1 to 4094.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show dot1x** [**interface** *interface-id*]	(Optional) Verify your entries.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the RADIUS server default settings, use the **no radius-server dead-criteria**, the **no radius-server deadtime**, and the **no radius-server host** global configuration commands. To return to the default settings of inaccessible authentication bypass, use the **no dot1x critical** {**eapol** | **recovery delay**} global configuration command. To disable inaccessible authentication bypass, use the **no dot1x critical** interface configuration command.

This example shows how to configure the inaccessible authentication bypass feature:

```
Switch(config)# radius-server dead-criteria time 30 tries 20
Switch(config)# radius-server deadtime 60
Switch(config)# radius-server host 1.1.1.2 acct-port 1550 auth-port 1560 key abc1234 test
username user1 idle-time 30
Switch(config)# dot1x critical eapol
Switch(config)# dot1x critical recovery delay 2000
Switch(config)# interface fastethernet0/1
Switch(config)# radius-server deadtime 60
Switch(config-if)# dot1x critical
Switch(config-if)# dot1x critical recovery action reinitialize
Switch(config-if)# dot1x critical vlan 20
Switch(config-if)# end
```

Configuring IEEE 802.1x Authentication with WoL

Beginning in privileged EXEC mode, follow these steps to enable IEEE 802.1x authentication with WoL. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode. For the supported port types, see the "IEEE 802.1x Authentication Configuration Guidelines" section on page 8-21.
Step 3	**dot1x control-direction** {**both** | **in**}	Enable IEEE 802.1x authentication with WoL on the port, and use these keywords to configure the port as bidirectional or unidirectional. • **both**—Sets the port as bidirectional. The port cannot receive packets from or send packets to the host. By default, the port is bidirectional. • **in**—Sets the port as unidirectional. The port can send packets to the host but cannot receive packets from the host.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show dot1x interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IEEE 802.1x authentication with WoL, use the **no dot1x control-direction** interface configuration command.

This example shows how to enable IEEE 802.1x authentication with WoL and set the port as bidirectional:

```
Switch(config-if)# dot1x control-direction both
```

Configuring MAC Authentication Bypass

Beginning in privileged EXEC mode, follow these steps to enable MAC authentication bypass. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode. For the supported port types, see the "IEEE 802.1x Authentication Configuration Guidelines" section on page 8-21.
Step 3	**dot1x port-control auto**	Enable IEEE 802.1x authentication on the port.
Step 4	**dot1x mac-auth-bypass** [**eap**]	Enable MAC authentication bypass. (Optional) Use the **eap** keyword to configure the switch to use EAP for authorization.
Step 5	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 6	**show dot1x interface** *interface-id*	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable MAC authentication bypass, use the **no dot1x mac-auth-bypass** interface configuration command.

This example shows how to enable MAC authentication bypass:

```
Switch(config-if)# dot1x mac-auth-bypass
```

Configuring NAC Layer 2 IEEE 802.1x Validation

In Cisco IOS Release 12.2(25)SED or later, you can configure NAC Layer 2 IEEE 802.1x validation, which is also referred to as IEEE 802.1x authentication with a RADIUS server.

Beginning in privileged EXEC mode, follow these steps to configure NAC Layer 2 IEEE 802.1x validation. The procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**dot1x guest-vlan** *vlan-id*	Specify an active VLAN as an IEEE 802.1x guest VLAN. The range is 1 to 4094. You can configure any active VLAN except an RSPAN VLAN or a voice VLAN as an IEEE 802.1x guest VLAN.
Step 4	**dot1x reauthentication**	Enable periodic re-authentication of the client, which is disabled by default.
Step 5	**dot1x timeout reauth-period** {*seconds* | **server**}	Set the number of seconds between re-authentication attempts. The keywords have these meanings: • *seconds*—Sets the number of seconds from 1 to 65535; the default is 3600 seconds. • **server**—Sets the number of seconds as the value of the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute [29]). This command affects the behavior of the switch only if periodic re-authentication is enabled.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show dot1x interface** *interface-id*	Verify your IEEE 802.1x authentication configuration.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to configure NAC Layer 2 IEEE 802.1x validation:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# dot1x reauthentication
Switch(config-if)# dot1x timeout reauth-period server
```

Configuring Web Authentication

Beginning in privileged EXEC mode, follow these steps to configure authentication, authorization, accounting (AAA) and RADIUS on a switch before configuring web authentication. The steps enable AAA by using RADIUS authentication and enable device tracking.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**aaa new-model**	Enable AAA.

Step 3	**aaa authentication login default group radius**	Use RADIUS authentication. Before you can use this authentication method, you must configure the RADIUS server. For more information, see Chapter 7, "Configuring Switch-Based Authentication."
		The console prompts you for a username and password on future attempts to access the switch console after entering the **aaa authentication login** command. If you do not want to be prompted for a username and password, configure a second login authentication list:
		`Switch# config t`
		`Switch(config)# aaa authentication login line-console none`
		`Switch(config)# line console 0`
		`Switch(config-line)# login authentication line-console`
		`Switch(config-line)# end`

Step 4	**aaa authorization auth-proxy default group radius**	Use RADIUS for authentication-proxy (auth-proxy) authorization.
Step 5	**radius-server host key** *radius-key*	Specify the authentication and encryption key for RADIUS communication between the switch and the RADIUS daemon.
Step 6	**radius-server attribute 8 include-in-access-req**	Configure the switch to send the Framed-IP-Address RADIUS attribute (Attribute[8]) in access-request or accounting-request packets.
Step 7	**radius-server vsa send authentication**	Configure the network access server to recognize and use vendor-specific attributes (VSAs).
Step 8	**ip device tracking**	Enable the IP device tracking table. To disable the IP device tracking table, use the **no ip device tracking** global configuration commands.
Step 9	**end**	Return to privileged EXEC mode.

This example shows how to enable AAA, use RADIUS authentication and enable device tracking:

```
Switch(config) configure terminal
Switch(config)# aaa new-model
Switch(config)# aaa authentication login default group radius
Switch(config)# aaa authorization auth-proxy default group radius
Switch(config)# radius-server host 1.1.1.2 key key1
Switch(config)# radius-server attribute 8 include-in-access-req
Switch(config)# radius-server vsa send authentication
Switch(config)# ip device tracking
Switch(config) end
```

Beginning in privileged EXEC mode, follow these steps to configure a port to use web authentication:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip admission name** *rule* **proxy http**	Define a web authentication rule. **Note** The same rule cannot be used for both web authentication and NAC Layer 2 IP validation. For more information, see the *Network Admission Control Software Configuration Guide* on Cisco.com.
Step 3	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 4	**switchport mode access**	Set the port to access mode.
Step 5	**ip access-group** *access-list* **in**	Specify the default access control list to be applied to network traffic before web authentication.
Step 6	**ip admission** *rule*	Apply an IP admission rule to the interface.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show running-config interface** *interface-id*	Verify your configuration.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to configure only web authentication on a switch port:

```
Switch# configure terminal
Switch(config)# ip admission name rule1 proxy http
Switch(config)# interface gigabit1/0/1
Switch(config-if)# switchport mode access
Switch(config-if)# ip access-group policy1 in
Switch(config-if)# ip admission rule1
Switch(config-if)# end
```

Beginning in privileged EXEC mode, follow these steps to configure a switch port for IEEE 802.1x authentication with web authentication as a fallback method:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip admission name** *rule* **proxy http**	Define a web authentication rule.
Step 3	**fallback profile** *fallback-profile*	Define a fallback profile to allow an IEEE 802.1x port to authenticate a client by using web authentication.
Step 4	**ip access-group** *policy* **in**	Specify the default access control list to apply to network traffic before web authentication.
Step 5	**ip admission** *rule*	Associate an IP admission rule with the profile, and specify that a client connecting by web authentication uses this rule.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 8	**switchport mode access**	Set the port to access mode.

	Command	Purpose
Step 9	**dot1x port-control auto**	Enable IEEE 802.1x authentication on the interface.
Step 10	**dot1x fallback** *fallback-profile*	Configure the port to authenticate a client by using web authentication when no IEEE 802.1x supplicant is detected on the port. Any change to the fallback-profile global configuration takes effect the next time IEEE 802.1x fallback is invoked on the interface. **Note** Web authorization cannot be used as a fallback method for IEEE 802.1x if the port is configured for multidomain authentication.
Step 11	**exit**	Return to privileged EXEC mode.
Step 12	**show dot1x interface** *interface-id*	Verify your configuration.
Step 13	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to configure IEEE 802.1x authentication with web authentication as a fallback method.

```
Switch(config) configure terminal
Switch(config)# ip admission name rule1 proxy http
Switch(config)# fallback profile fallback1
Switch(config-fallback-profile)# ip access-group default-policy in
Switch(config-fallback-profile)# ip admission rule1
Switch(config-fallback-profile)# exit
Switch(config)# interface gigabit1/0/1
Switch(config-if)# switchport mode access
Switch(config-if)# dot1x port-control auto
Switch(config-if)# dot1x fallback fallback1
Switch(config-if)# end
```

For more information about the **dot1x fallback** command, see the command reference for this release. For more information about the **ip admission name** and **ip access-group** commands, see the *Network Admission Control Software Configuration Guide* on Cisco.com.

Disabling IEEE 802.1x on the Port

You can disable IEEE 802.1x authentication on the port by using the **no dot1x pae** interface configuration command.

Beginning in privileged EXEC mode, follow these steps to disable IEEE 802.1x authentication on the port. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**no dot1x pae**	Disable IEEE 802.1x authentication on the port.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show dot1x interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To configure the port as an IEEE 802.1x port access entity (PAE) authenticator, which enables IEEE 802.1x on the port but does not allow clients connected to the port to be authorized, use the **dot1x pae authenticator** interface configuration command.

This example shows how to disable IEEE 802.1x authentication on the port:

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# no dot1x pae authenticator
```

Resetting the IEEE 802.1x Configuration to the Default Values

Beginning in privileged EXEC mode, follow these steps to reset the IEEE 802.1x configuration to the default values.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**dot1x default**	Reset the configurable IEEE 802.1x parameters to the default values.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show dot1x interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Displaying IEEE 802.1x Statistics and Status

To display IEEE 802.1x statistics for all interfaces, use the **show dot1x all statistics** privileged EXEC command. To display IEEE 802.1x statistics for a specific interface, use the **show dot1x statistics interface** *interface-id* privileged EXEC command.

To display the IEEE 802.1x administrative and operational status for the switch, use the **show dot1x all** privileged EXEC command. To display the IEEE 802.1x administrative and operational status for a specific interface, use the **show dot1x interface** *interface-id* privileged EXEC command.

For detailed information about the fields in these displays, see the command reference for this release.

CHAPTER 9

Configuring Interface Characteristics

This chapter describes the types of interfaces on a Catalyst 3550 switch and how to configure them. The chapter has these sections:

- Understanding Interface Types, page 9-1
- Using the Interface Command, page 9-9
- Configuring Ethernet Interfaces, page 9-13
- Configuring Layer 3 Interfaces, page 9-19
- Monitoring and Maintaining the Interfaces, page 9-21

Note For complete syntax and usage information for the commands used in this chapter, see the switch command reference for this release and the online *Cisco IOS Interface Command Reference, Release 12.2.*

Understanding Interface Types

This section describes the different types of interfaces supported by the switch with references to chapters that contain more detailed information about configuring these interface types. The rest of the chapter describes configuration procedures for physical interface characteristics.

These sections are included:

- Port-Based VLANs, page 9-2
- Switch Ports, page 9-2
- Switch Virtual Interfaces, page 9-4
- Routed Ports, page 9-4
- EtherChannel Port Groups, page 9-5
- Power Over Ethernet Ports, page 9-5
- Connecting Interfaces, page 9-7

Port-Based VLANs

A VLAN is a switched network that is logically segmented by function, team, or application, without regard to the physical location of the users. For more information about VLANs, see Chapter 11, "Configuring VLANs." Packets received on a port are forwarded only to ports that belong to the same VLAN as the receiving port. Network devices in different VLANs cannot communicate with one another without a Layer 3 device to route traffic between the VLANs.

VLAN partitions provide hard firewalls for traffic in the VLAN, and each VLAN has its own MAC address table. A VLAN comes into existence when a local port is configured to be associated with the VLAN, when the VLAN Trunking Protocol (VTP) learns of its existence from a neighbor on a trunk, or when a user creates a VLAN.

To configure normal-range VLANs (VLAN IDs 1 to 1005), use the **vlan** *vlan-id* global configuration command to enter config-vlan mode or the **vlan database** privileged EXEC command to enter VLAN configuration mode. The VLAN configurations for VLAN IDs 1 to 1005 are saved in the VLAN database. To configure extended-range VLANs (VLAN IDs 1006 to 4094), you must use config-vlan mode with VTP mode set to transparent. Extended-range VLANs are not added to the VLAN database. When VTP mode is transparent, the VTP and VLAN configuration is saved in the switch running configuration, and you can save it in the switch startup configuration file by entering the **copy running-config startup-config** privileged EXEC command.

Add ports to a VLAN by using the **switchport** interface configuration commands:

- Identify the interface.
- For a trunk port, set trunk characteristics, and if desired, define the VLANs to which it can belong.
- For an access port, set and define the VLAN to which it belongs.
- For a tunnel port, set and define the VLAN ID for the customer-specific VLAN tag. See Chapter 14, "Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling."

Switch Ports

Switch ports are Layer 2-only interfaces associated with a physical port. A switch port can be an access port, a trunk port, or a tunnel port. You can configure a port as an access port or trunk port or let the Dynamic Trunking Protocol (DTP) operate on a per-port basis to determine if a switch port should be an access port or a trunk port by negotiating with the port on the other end of the link. You must manually configure tunnel ports as part of an asymmetric link connected to an IEEE 802.1Q trunk port. Switch ports are used for managing the physical interface and associated Layer 2 protocols and do not handle routing or bridging.

Configure switch ports by using the **switchport** interface configuration commands. For detailed information about configuring access port and trunk port characteristics, see Chapter 11, "Configuring VLANs." For more information about tunnel ports, see Chapter 14, "Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling."

Access Ports

An access port belongs to and carries the traffic of only one VLAN (unless it is configured as a voice VLAN port). Traffic is received and sent in native formats with no VLAN tagging. Traffic arriving on an access port is assumed to belong to the VLAN assigned to the port. If an access port receives a tagged

packet (Inter-Switch Link [ISL] or IEEE 802.1Q tagged) for the VLAN assigned to the port, the packet is forwarded. If the port receives a tagged packet for another VLAN, the packet is dropped, the source address is not learned, and the frame is counted in the *No destination* statistic.

Two types of access ports are supported:

- Static access ports are manually assigned to a VLAN.
- VLAN membership of dynamic access ports is learned through incoming packets. By default, a dynamic access port is not a member of any VLAN, and forwarding to and from the port is enabled only when the VLAN membership of the port is discovered. Dynamic access ports on the switch are assigned to a VLAN by a VLAN Membership Policy Server (VMPS). The VMPS can be a Catalyst 6000 series switch; the Catalyst 3550 switch does not support the function of a VMPS.

  You can also configure an access port with an attached Cisco IP Phone to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone. For more information about voice VLAN ports, see Chapter 13, "Configuring Voice VLAN."

Trunk Ports

A trunk port carries the traffic of multiple VLANs and by default is a member of all VLANs in the VLAN database. Two types of trunk ports are supported:

- In an ISL trunk port, all received packets are expected to be encapsulated with an ISL header, and all transmitted packets are sent with an ISL header. Native (non-tagged) frames received from an ISL trunk port are dropped.
- An IEEE 802.1Q trunk port supports simultaneous tagged and untagged traffic. An IEEE 802.1Q trunk port is assigned a default Port VLAN ID (PVID), and all untagged traffic travels on the port default PVID. All untagged traffic and tagged traffic with a NULL VLAN ID are assumed to belong to the port default PVID. A packet with a VLAN ID equal to the outgoing port default PVID is sent untagged. All other traffic is sent with a VLAN tag.

Although by default, a trunk port is a member of every VLAN known to the VTP, you can limit VLAN membership by configuring an allowed list of VLANs for each trunk port. The list of allowed VLANs does not affect any other port but the associated trunk port. By default, all possible VLANs (VLAN ID 1 to 4094) are in the allowed list. A trunk port can only become a member of a VLAN if VTP knows of the VLAN and the VLAN is in the enabled state. If VTP learns of a new, enabled VLAN and the VLAN is in the allowed list for a trunk port, the trunk port automatically becomes a member of that VLAN and traffic is forwarded to and from the trunk port for that VLAN. If VTP learns of a new, enabled VLAN that is not in the allowed list for a trunk port, the port does not become a member of the VLAN, and no traffic for the VLAN is forwarded to or from the port.

For more information about trunk ports, see Chapter 11, "Configuring VLANs."

Tunnel Ports

Tunnel ports are used in IEEE 802.1Q tunneling to segregate the traffic of customers in a service-provider network from other customers who seem to be on the same VLAN. You configure an asymmetric link from a tunnel port on a service-provider edge switch to an IEEE 802.1Q trunk port on the customer switch. Packets entering the tunnel port on the edge switch, already IEEE 802.1Q-tagged with the customer VLANs, are encapsulated with another layer of an IEEE 802.1Q tag (called the metro tag), containing a VLAN ID unique in the service-provider network, for each customer. The double-tagged packets go through the service-provider network, keeping the original customer VLANs separate from those of other customers. At the outbound interface, also a tunnel port, the metro tag is removed, and the original VLAN numbers from the customer network are retrieved.

Tunnel ports cannot be trunk ports or access ports and must belong to a VLAN unique to each customer.

For more information about tunnel ports, see Chapter 14, "Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling."

Switch Virtual Interfaces

A switch virtual interface (SVI) represents a VLAN of switch ports as one interface to the routing or bridging function in the system. Only one SVI can be associated with a VLAN, but you need to configure an SVI for a VLAN only when you wish to route between VLANs, fallback-bridge nonroutable protocols between VLANs, or to provide IP host connectivity to the switch. By default, an SVI is created for the default VLAN (VLAN 1) to permit remote switch administration. You cannot delete interface VLAN 1. Additional SVIs must be explicitly configured. In Layer 2 mode, SVIs provide IP host connectivity only to the system; in Layer 3 mode, you can configure routing across SVIs.

SVIs are created the first time that you enter the **vlan** interface configuration command for a VLAN interface. The VLAN corresponds to the VLAN tag associated with data frames on an ISL or IEEE 802.1Q encapsulated trunk or the VLAN ID configured for an access port. Configure a VLAN interface for each VLAN for which you want to route traffic, and assign it an IP address. For more information, see the "Configuring IP Addressing on Layer 3 Interfaces" section on page 31-4.

Note When you create an SVI, it does not become active until it is associated with a physical port.

SVIs support routing protocols and bridging configurations. For more information about configuring IP routing, see Chapter 31, "Configuring IP Unicast Routing," Chapter 3, "Configuring IP Multicast Routing,"and Chapter 5, "Configuring Fallback Bridging."

Note The IP base image, formerly known as the standard multilayer software image (SMI), supports static routing and the Routing Information Protocol (RIP). To use SVIs for full Layer 3 routing or for fallback bridging, you must have the IP services image, formerly known as the enhanced multilayer software image (EMI), installed on your switch.

Routed Ports

A routed port is a physical port that acts like a port on a router; it does not have to be connected to a router. A routed port is not associated with a particular VLAN, as is an access port. A routed port behaves like a regular router interface, except that it does not support VLAN subinterfaces. Routed ports can be configured with a Layer 3 routing protocol.

Note The IP base image supports static routing and RIP; for more advanced routing, you must have the IP services image installed on your switch.

Configure routed ports by putting the interface into Layer 3 mode with the **no switchport** interface configuration command. Then assign an IP address to the port, enable routing, and assign routing protocol characteristics by using the **ip routing** and **router** *protocol* global configuration commands.

Caution Entering a **no switchport** interface configuration command shuts the interface down and then re-enables it, which might generate messages on the device to which the interface is connected.

The number of routed ports and SVIs that you can configure is not limited by software; however, the interrelationship between this number and the number of other features being configured might have an impact on CPU utilization because of hardware limitations. For more information about feature combinations, see the "Administering the Switch" section on page 6-1.

For more information about IP unicast and multicast routing and routing protocols, see Chapter 31, "Configuring IP Unicast Routing" and Chapter 3, "Configuring IP Multicast Routing."

EtherChannel Port Groups

EtherChannel port groups provide the ability to treat multiple switch ports as one switch port. These port groups act as a single logical port for high-bandwidth connections between switches or between switches and servers. An EtherChannel balances the traffic load across the links in the channel. If a link within the EtherChannel fails, traffic previously carried over the failed link changes to the remaining links. You can group multiple trunk ports into one logical trunk port, group multiple access ports into one logical access port, group multiple tunnel ports into one logical tunnel port, or group multiple routed ports into one logical routed port. Most protocols operate over either single ports or aggregated switch ports and do not recognize the physical ports within the port group. Exceptions are the DTP, the Cisco Discovery Protocol (CDP), and the Port Aggregation Protocol (PAgP), which operate only on physical ports.

When you configure an EtherChannel, you create a port-channel logical interface and assign an interface to the EtherChannel. For Layer 3 interfaces, you manually create the logical interface by using the **interface port-channel** global configuration command. For Layer 2 interfaces, the logical interface is dynamically created. For both Layer 3 and 2 interfaces, you manually assign an interface to the EtherChannel by using the **channel-group** interface configuration command. This command binds the physical and logical ports together. For more information, see Chapter 30, "Configuring EtherChannels."

Power Over Ethernet Ports

Catalyst 3550 PoE-capable switch ports automatically supply power to these connected devices (if the switch senses that there is no power on the circuit):

- Cisco pre-standard powered devices (such as Cisco IP Phones and Cisco Aironet access points)
- IEEE 802.3af-compliant powered devices

A powered device can receive redundant power when it is connected to a PoE switch port and to an AC power source.

Note PoE ports were previously referred to as inline power ports in earlier versions of the software configuration guide.

Supported Protocols and Standards

The switch uses these protocols and standards to support PoE:

- CDP with power consumption—The powered device notifies the switch of the amount of power it is consuming. The switch does not reply to the power-consumption messages. The switch can only supply power to or remove power from the PoE port.
- Cisco intelligent power management—The powered device and the switch negotiate through power-negotiation CDP messages for an agreed power-consumption level. The negotiation allows a high-power Cisco powered device, which consumes more than 7 W, to operate at its highest power mode. The powered device first boots up in low-power mode, consumes less than 7 W, and negotiates to obtain enough power to operate in high-power mode. The device changes to high-power mode only when it receives confirmation from the switch.

  High-power devices can operate in low-power mode on switches that do not support power-negotiation CDP.

  Before Release 12.1(22)EA2, Catalyst 3550 PoE-capable switches (without intelligent power management support) caused high-power powered devices that supported intelligent power management to operate in low-power mode. Devices in low-power mode are not fully functional.

  Cisco intelligent power management is backward-compatible with CDP with power consumption; the switch responds according to the CDP message that it receives. CDP is not supported on third-party powered devices; therefore, the switch uses the IEEE classification to determine the power usage of the device.
- IEEE 802.3af—The major features of this standard are powered-device discovery, power administration, disconnect detection, and optional powered-device power classification. For more information, see the standard.

Powered-Device Detection and Initial Power Allocation

The switch detects a Cisco pre-standard or an IEEE-compliant powered device when the PoE-capable port is in the no-shutdown state, PoE is enabled (the default), and the connected device is not being powered by an AC adaptor.

After device detection, the switch determines the device power requirements based on its type:

- A Cisco pre-standard powered device does not provide its power requirement when the switch detects it, so the switch allocates 15.4 W as the initial allocation for power budgeting.

  The initial power allocation is the maximum amount of power that a powered device requires. The switch initially allocates this amount of power when it detects and powers the powered device. As the switch receives CDP messages from the powered device and as the powered device negotiates power levels with the switch through CDP power-negotiation messages, the initial power allocation might be adjusted.
- For an IEEE device, the switch always allocates 15.4 W to the port. The switch does not display the IEEE class type in the **show power inline** privileged EXEC command output. Instead, it displays *n/a*.

The switch monitors and tracks requests for power and grants power only when it is available. The switch tracks its power budget (the amount of power available on the switch for PoE). The switch performs power-accounting calculations when a port is granted or denied power to keep the power budget up to date.

After power is applied to the port, the switch uses CDP to determine the *actual* power consumption requirement of the connected Cisco powered devices, and the switch adjusts the power budget accordingly. This does not apply to third-party PoE devices. The switch processes a request and either grants or denies power. If the request is granted, the switch updates the power budget. If the request is denied, the switch ensures that power to the port is turned off, generates a syslog message, and updates the LEDs. Powered devices can also negotiate with the switch for more power.

If the switch detects a fault caused by an undervoltage, overvoltage, overtemperature, oscillator-fault, or short-circuit condition, it turns off power to the port, generates a syslog message, and updates the power budget and LEDs.

Power Management Modes

The switch supports these PoE modes:

- **auto**—The switch automatically detects if the connected device requires power. If the switch discovers a powered device connected to the port and if the switch has enough power, it grants power, updates the power budget, turns on power to the port on a first-come, first-served basis, and updates the LEDs. For LED information, see the hardware installation guide.

  If the switch has enough power for all the powered devices, they all come up. If enough power is available for all powered devices connected to the switch, power is turned on to all devices. If there is not enough available PoE, or if a device is disconnected and reconnected while other devices are waiting for power, it cannot be determined which devices are granted or are denied power.

  If granting power would exceed the system power budget, the switch denies power, ensures that power to the port is turned off, generates a syslog message, and updates the LEDs. After power has been denied, the switch periodically rechecks the power budget and continues to attempt to grant the request for power.

  If a device being powered by the switch is then connected to wall power, the switch might continue to power the device. The switch might continue to report that it is still powering the device whether the device is being powered by the switch or receiving power from an AC power source.

  If a powered device is removed, the switch automatically detects the disconnect and removes power from the port. You can connect a nonpowered device without damaging it.

  Use the **auto** setting on any PoE port. The auto mode is the default setting.

- **never**—The switch disables powered-device detection and never powers the PoE port even if an unpowered device is connected. Use this mode only when you want to make sure power is never applied to a PoE-capable port, making the port a data-only port.

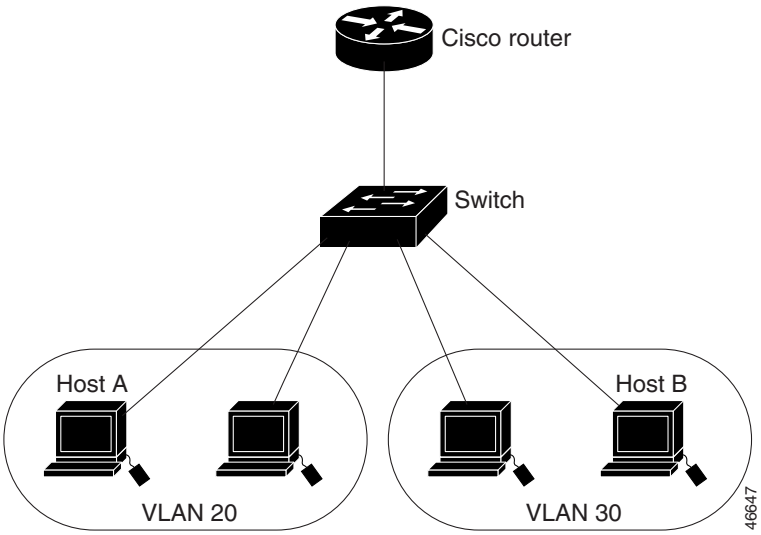
For information on configuring a PoE port, see the "Configuring Power over Ethernet on the Catalyst 3550-24PWR Ports" section on page 9-16.

Connecting Interfaces

Devices within a single VLAN can communicate directly through any switch. Ports in different VLANs cannot exchange data without going through a routing device or routed interface.

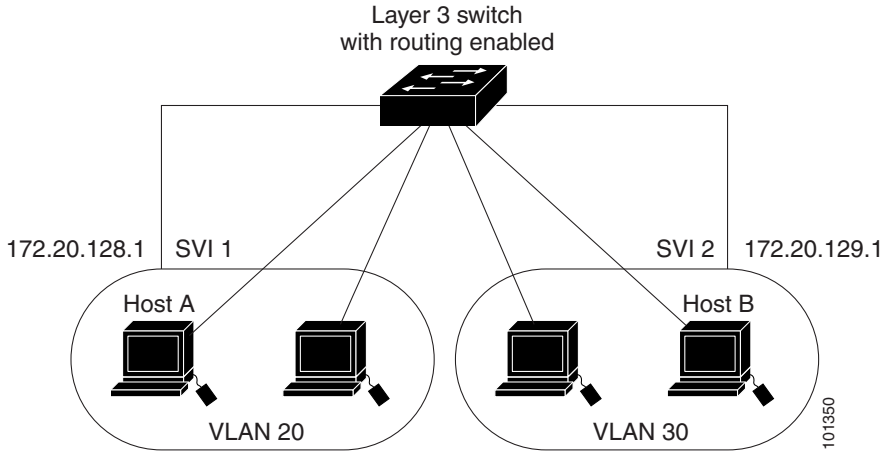
With a standard Layer 2 switch, ports in different VLANs have to exchange information through a router. In the configuration shown in Figure 9-1, when Host A in VLAN 20 sends data to Host B in VLAN 30, it must go from Host A to the switch, to the router, back to the switch, and then to Host B.

Figure 9-1 Connecting VLANs with Layer 2 Switches



By using the Catalyst 3550 with routing enabled (as a Layer 3 switch), when you configure VLAN 20 and VLAN 30 each with an SVI to which an IP address is assigned, packets can be sent from Host A to Host B directly through the Catalyst 3550 switch with no need for an external router (Figure 9-2).

Figure 9-2 Connecting VLANs with the a Layer 3 Switch



The switch with the enhanced multilayer software image supports two methods of forwarding traffic between interfaces: routing and fallback bridging; the standard software image supports only basic routing (static routing and RIP). Whenever possible, to maintain high performance, forwarding is done by switch hardware. However, only IP version 4 packets with Ethernet II encapsulation can be routed in hardware. All other types of traffic can be fallback bridged by hardware.

- The routing function can be enabled on all SVIs and routed ports. The Catalyst 3550 switches route only IP traffic. When IP routing protocol parameters and address configuration are added to an SVI or routed port, any IP traffic received from these ports is routed. For more information, see Chapter 31, "Configuring IP Unicast Routing," Chapter 3, "Configuring IP Multicast Routing," and Chapter 4, "Configuring MSDP."

- Fallback bridging forwards traffic that the switch with the enhanced multilayer software image does not route or traffic belonging to a nonroutable protocol, such as DECnet. Fallback bridging connects multiple VLANs into one bridge domain by bridging between two or more SVIs or routed ports. When configuring fallback bridging, you assign SVIs or routed ports to bridge groups with each SVI or routed port assigned to only one bridge group. All interfaces in the same group belong to the same bridge domain. For more information, see Chapter 5, "Configuring Fallback Bridging."

Using the Interface Command

The switch supports these interface types:

- Physical ports—including switch ports and routed ports
- VLANs—switch virtual interfaces
- Port-channels—EtherChannel of interfaces

You can also configure a range of interfaces (see the "Configuring a Range of Interfaces" section on page 9-10).

To configure a physical interface (port), enter interface configuration mode, and specify the interface type, slot, and number.

- Type—Fast Ethernet (fastethernet or fa) for 10/100 Ethernet or Gigabit Ethernet (gigabitethernet or gi)
- Slot—The slot number on the switch (always 0 on this switch).
- Port number—The interface number on the switch. The port numbers always begin at 1, starting with the leftmost port when facing the front of the switch, for example, fastethernet0/1, fastethernet0/2. If there is more than one interface type (for example, 10/100 ports and Gigabit Ethernet ports), the port number restarts with the second interface type: gigabitethernet0/1, gigabitethernet0/2.

You can identify physical interfaces by physically checking the interface location on the switch. You can also use the Cisco IOS **show** privileged EXEC commands to display information about a specific interface or all the interfaces on the switch. The remainder of this chapter primarily provides physical interface configuration procedures.

Procedures for Configuring Interfaces

These general instructions apply to all interface configuration processes.

Step 1 Enter the **configure terminal** command at the privileged EXEC prompt:

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

Step 2 Enter the **interface** global configuration command. Identify the interface type and the number of the connector. In this example, Gigabit Ethernet interface 0/1 is selected:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)#
```

Note You do not need to add a space between the interface type and interface number. For example, in the preceding line, you can specify either **gigabitethernet 0/1**, **gigabitethernet0/1**, **gi 0/1**, or **gi0/1**.

Step 3 Follow each **interface** command with the interface configuration commands your particular interface requires. The commands you enter define the protocols and applications that will run on the interface. The commands are collected and applied to the interface when you enter another interface command or enter **end** to return to privileged EXEC mode.

You can also configure a range of interfaces by using the **interface range** or **interface range macro** global configuration commands. Interfaces configured in a range must be the same type and must be configured with the same feature options.

Step 4 After you configure an interface, verify its status by using the **show** privileged EXEC commands listed in the "Monitoring and Maintaining the Interfaces" section on page 9-21.

Enter the **show interfaces** privileged EXEC command to see a list of all interfaces on or configured for the switch. A report is provided for each interface that the device supports or for the specified interface:

Configuring a Range of Interfaces

You can use the **interface range** global configuration command to configure multiple interfaces with the same configuration parameters. When you enter the interface range configuration mode, all command parameters that you enter are attributed to all interfaces within that range until you exit this mode.

Beginning in privileged EXEC mode, follow these steps to configure a range of interfaces with the same parameters:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface range** {*port-range* | **macro** *macro_name*}	Specify the range of interfaces (VLANs or physical ports) to be configured, and enter interface range configuration mode. • You can use the **interface range** command to configure up to five port ranges or a previously defined macro. • The **macro** variable is explained in the "Configuring and Using Interface Range Macros" section on page 9-12. • Each comma-separated *port-range* must consist of the same port type. You do not need to enter spaces before or after the comma. • When you define a range, the space between the first port and the hyphen is required.
Step 3		You can now use the normal configuration commands to apply the configuration parameters to all interfaces in the range.
Step 4	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 5	**show interfaces** [*interface-id*]	Verify the configuration of the interfaces in the range.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

When using the **interface range** global configuration command, note these guidelines:

- Valid entries for *port-range*:
  - **vlan** *vlan-ID* - *vlan-ID*, where VLAN ID is from 1 to 4094
  - **fastethernet** slot/{*first port*} - {*last port*}, where slot is **0**
  - **gigabitethernet** slot/{*first port*} - {*last port*}, where slot is **0**
  - **port-channel** *port-channel-number* - *port-channel-number*, where *port-channel-number* is from 1 to 64
- You must add a space between the interface numbers and the hyphen when using the **interface range** command. For example, the command **interface range fastethernet0/1 - 5** is a valid range; the command **interface range fastethernet0/1-5** is not a valid range.
- The **interface range** command works only with VLAN interfaces that have been configured with the **interface vlan** command (the **show running-config** privileged EXEC command displays the configured VLAN interfaces). VLAN interfaces not displayed by the **show running-config** command cannot be used with the **interface range** command.
- All interfaces in a range must be the same type; that is, all Fast Ethernet ports, all Gigabit Ethernet ports, all EtherChannel ports, or all SVIs.

This example shows how to use the **interface range** global configuration command to a range of interfaces:

```
Switch# configure terminal
Switch(config)# interface range fastethernet0/1 - 5
Switch(config-if-range)# no shutdown
Switch(config-if-range)#
*Oct  6 08:24:35: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
*Oct  6 08:24:35: %LINK-3-UPDOWN: Interface FastEthernet0/2, changed state to up
*Oct  6 08:24:35: %LINK-3-UPDOWN: Interface FastEthernet0/3, changed state to up
*Oct  6 08:24:35: %LINK-3-UPDOWN: Interface FastEthernet0/4, changed state to up
*Oct  6 08:24:35: %LINK-3-UPDOWN: Interface FastEthernet0/5, changed state to up
*Oct  6 08:24:36: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/05,
changed state to up
*Oct  6 08:24:36: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed
state to up
*Oct  6 08:24:36: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/4, changed
state to up
```

This example shows how to use a comma to add different interface type strings to the range to enable all Fast Ethernet and Gigabit Ethernet interfaces:

```
Switch# configure terminal
Switch(config)# interface range fastethernet0/1 - 3, gigabitethernet0/1 - 2
Switch(config-if-range)# no shutdown
Switch(config-if-range)#
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface FastEthernet0/2, changed state to up
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface FastEthernet0/3, changed state to up
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface GigabitEthernet0/1, changed state to up
*Oct  6 08:29:28: %LINK-3-UPDOWN: Interface GigabitEthernet0/2, changed state to up
*Oct  6 08:29:29: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/ 1,
changed state to up
```

```
*Oct  6 08:29:29: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/ 2,
changed state to up
*Oct  6 08:29:29: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/ 3,
changed state to up
```

If you enter multiple configuration commands while you are in interface range mode, each command is executed as it is entered. The commands are not batched together and executed after you exit interface range mode. If you exit interface range configuration mode while the commands are being executed, some commands might not be executed on all interfaces in the range. Wait until the command prompt reappears before exiting interface range configuration mode.

Configuring and Using Interface Range Macros

You can create an interface range macro to automatically select a range of interfaces for configuration. Before you can use the **macro** keyword in the **interface range macro** global configuration command string, you must use the **define interface-range** global configuration command to define the macro.

Beginning in privileged EXEC mode, follow these steps to define an interface range macro:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**define interface-range** *macro_name interface-range*	Define the interface-range macro, and save it in NVRAM. • The *macro_name* is a 32-character maximum character string. • A macro can contain up to five comma-separated interface ranges. You do not need to enter spaces before or after the comma. • Each *interface-range* must consist of the same port type.
Step 3	**interface range macro** *macro_name*	Select the interface range to be configured using the values saved in the interface-range macro called *macro_name*. You can now use the normal configuration commands to apply the configuration to all interfaces in the defined macro.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config | include define**	Show the defined interface range macro configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no define interface-range** *macro_name* global configuration command to delete a macro.

When using the **define interface-range** global configuration command, note these guidelines:

- Valid entries for *interface-range*:
  - **vlan** *vlan-ID* - *vlan-ID*, where VLAN ID is from 1 to 4094
  - **fastethernet** slot/{*first port*} - {*last port*}, where slot is **0**
  - **gigabitethernet** slot/{*first port*} - {*last port*}, where slot is **0**
  - **port-channel** *port-channel-number* - *port-channel-number*, where *port-channel-number* is from 1 to 64.

- You must add a space between the interface numbers and the hyphen when entering an *interface-range*. For example, **fastethernet0/1 - 5** is a valid range; **fastethernet0/1-5** is not a valid range.
- The VLAN interfaces (SVIs) must have been configured with the **interface vlan** command. The **show running-config** privileged EXEC command displays the configured VLAN interfaces. VLAN interfaces not displayed by the **show running-config** command cannot be used as *interface-ranges*.
- All interfaces in a range must be the same type; that is, all Fast Ethernet ports, all Gigabit Ethernet ports, all EtherChannel ports, or all VLANs, but you can combine multiple interface types in a macro.

This example shows how to define an interface-range macro named *enet_list* to select Fast Ethernet ports 1 to 4 and to verify the macro configuration:

```
Switch# configure terminal
Switch(config)# define interface-range enet_list fastethernet0/1 - 4
Switch(config)# end
Switch# show running-config | include define
define interface-range enet_list FastEthernet0/1 - 4
```

This example shows how to create a multiple-interface macro named *macro1*:

```
Switch# configure terminal
Switch(config)# define interface-range macro1 gigabitethernet0/1 - 2, fastethernet0/5 - 7
Switch(config)# end
Switch#
```

This example shows how to enter interface range configuration mode for the interface-range macro *enet_list*:

```
Switch# configure terminal
Switch(config)# interface range macro enet_list
Switch(config-if-range)#
```

This example shows how to delete the interface-range macro *enet_list* and to verify that it has been deleted.

```
Switch# configure terminal
Switch(config)# no define interface-range enet_list
Switch# show run | include define
```

Configuring Ethernet Interfaces

These sections describe the default interface configuration and the optional features that you can configure on most physical interfaces:

- Default Ethernet Interface Configuration, page 9-14
- Configuring Interface Speed and Duplex Mode, page 9-15
- Configuring Power over Ethernet on the Catalyst 3550-24PWR Ports, page 9-16
- Configuring IEEE 802.3x Flow Control, page 9-17
- Adding a Description for an Interface, page 9-19

Caution If the interface is in Layer 3 mode, after entering interface configuration mode, you must enter the **switchport** interface configuration command without any parameters to put the interface into Layer 2 mode. This shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. Furthermore, when you use this command to put the interface into Layer 2 mode, you are deleting any Layer 3 characteristics configured on the interface.

Default Ethernet Interface Configuration

Table 9-1 shows the Ethernet interface default configuration. For more details on the VLAN parameters listed in the table, see Chapter 11, "Configuring VLANs." For details on controlling traffic to the port, see Chapter 21, "Configuring Port-Based Traffic Control."

Table 9-1 Default Ethernet Interface Configuration

Feature	Default Setting
Operating mode	Layer 2 or switching mode (**switchport** command).
Allowed VLAN range	VLANs 1 – 4094.
Default VLAN (for access ports)	VLAN 1.
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1.
VLAN trunking	Switchport mode dynamic desirable (supports DTP).
Port enable state	All ports are enabled.
Port description	None defined.
Speed	Autonegotiate.
Duplex mode	Autonegotiate.
Flow control	Flow control is set to *off* for **receive** and *desired* for **send** for Gigabit Ethernet ports. For 10/100 Mb/s ports, **send** is always *off*.
Power over Ethernet (supported only on the Catalyst 3550-24PWR switch)	Enabled (auto).
EtherChannel (PAgP)	Disabled on all Ethernet ports. See Chapter 30, "Configuring EtherChannels."
Port blocking (unknown multicast and unknown unicast traffic)	Disabled (not blocked). See the "Configuring Port Blocking" section on page 21-6.
Broadcast, multicast, and unicast storm control	Disabled. See the "Default Storm Control Configuration" section on page 21-3.
Protected port	Disabled. See the "Configuring Protected Ports" section on page 21-5.
Port security	Disabled. See the "Default Port Security Configuration" section on page 21-9.
Port Fast	Disabled.

Configuring Interface Speed and Duplex Mode

Ethernet interfaces on the switch operate in 10, 100, or 1000 Mbps and in either full- or half-duplex mode. In full-duplex mode, two stations can send and receive at the same time. When packets can flow in both directions simultaneously, effective Ethernet bandwidth doubles to 20 Mbps for 10-Mbps interfaces, to 200 Mbps for Fast Ethernet interfaces, and to 2 Gbps for Gigabit interfaces. Full-duplex communication is often an effective solution to collisions, which are major constrictions in Ethernet networks. Normally, 10-Mbps ports operate in half-duplex mode, which means that stations can either receive or send.

You can configure interface speed on Fast Ethernet (10/100-Mbps) and Gigabit Ethernet (10/100/1000-Mbps) interfaces; you cannot configure speed on Gigabit Interface Converter (GBIC) interfaces. You can configure duplex mode on any Fast Ethernet or Gigabit Ethernet interfaces that are not set to autonegotiate; you cannot configure duplex mode on GBIC interfaces.

Note You cannot configure speed or duplex mode on Gigabit Interface Converter (GBIC) ports, but for certain types of GBICs, you can configure speed to not negotiate (**nonegotiate**) if connected to a device that does not support autonegotiation.

These sections describe how to configure the interface speed and duplex mode:

- Configuration Guidelines, page 9-15
- Setting the Interface Speed and Duplex Parameters, page 9-16

Configuration Guidelines

When configuring an interface speed and duplex mode, note these guidelines:

- If both ends of the line support autonegotiation, we highly recommend the default setting of **auto**negotiation.
- If one interface supports autonegotiation and the other end does not, configure duplex and speed on both interfaces; do not use the **auto** setting on the supported side.
- 100BASE-FX ports operate only at 100 Mbps in either full- or half-duplex mode and do not support autonegotiation.
- GigaStack-to-GigaStack cascade connections operate in half-duplex mode, and GigaStack-to-GigaStack point-to-point connections operate in full-duplex mode.
- When STP is enabled and a port is reconfigured, the switch can take up to 30 seconds to check for loops. The port LED is amber while STP reconfigures.

Caution Changing the interface speed and duplex mode configuration might shut down and re-enable the interface during the reconfiguration.

Setting the Interface Speed and Duplex Parameters

Beginning in privileged EXEC mode, follow these steps to set the speed and duplex mode for a physical interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the physical interface to be configured, and enter interface configuration mode.
Step 3	**speed** {**10** | **100** | **1000** | **auto** [**10** | **100** | **1000**] | **nonegotiate**}	Enter the appropriate speed parameter for the interface, or enter **auto** or **nonegotiate**. **Note** The **1000** keyword is available only for 10/100/1000 Mbps ports. 100BASE-FX ports operate only at 100 Mbps. GBIC module ports operate only at 1000 Mbps. The **nonegotiate** keyword is available only for 1000BASE-SX, -LX, and -ZX GBIC ports. If you use the **10**, **100**, or **1000** keywords with the **auto** keyword, the port only autonegotiates at the specified speeds.
Step 4	**duplex** {**auto** | **full** | **half**}	Enter the duplex parameter for the interface. **Note** 100BASE-FX ports operate only in full-duplex mode. This keyword is not available on GBIC ports.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interfaces** *interface-id*	Display the interface speed and duplex mode configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no speed** and **no duplex** interface configuration commands to return the interface to the default speed and duplex settings (autonegotiate). To return all interface settings to the defaults, use the **default interface** *interface-id* interface configuration command.

This example shows how to set the interface speed to 10 Mbps and the duplex mode to half:

```
Switch# configure terminal
Switch(config)# interface fastethernet0/3
Switch(config-if)# speed 10
Switch(config-if)# duplex half
```

Configuring Power over Ethernet on the Catalyst 3550-24PWR Ports

The Catalyst 3550-24PWR switch automatically supplies Power over Ethernet (PoE) to connected Cisco IP Phones, Cisco Aironet Access Points, and IEEE-compliant powered devices if it senses *no* power on the circuit. If there is power on the circuit, the switch does not supply it.

Note PoE ports were previously referred to as inline power ports in earlier versions of the software configuration guide.

For information about configuring a switch port to forward IP voice traffic to and from connected Cisco IP Phones, see the "Configuring a Port to Connect to a Cisco 7960 IP Phone" section on page 13-3.

For information about configuring the switch for certain IEEE-compliant powered devices that require multiple reloads during initialization, see the **power inline** interface configuration command in the command reference for this release.

Beginning in privileged EXEC mode, follow these steps to enable PoE on a PoE-capable port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface*	Specify the port to be configured, and enter interface configuration mode.
Step 3	**power inline auto**	Enable powered-device detection. If enough power is available, automatically allocate power to the PoE port after device detection. This is the default.
Step 4	end	Return to privileged EXEC mode.
Step 5	**show power inline** *interface*	Verify the change.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable PoE on a port, use the **power inline never** interface configuration command.

Note If a port has a Cisco powered device connected to it, you should not use the **power inline never** command to configure the port. A false link-up can occur on the port, placing the port into an error-disabled state.

Configuring IEEE 802.3x Flow Control

Flow control enables connected Ethernet ports to control traffic rates during congestion by allowing congested nodes to pause link operation at the other end. If one port experiences congestion and cannot receive any more traffic, it notifies the other port to stop sending until the condition clears. When the local device detects any congestion at its end, it can notify the link partner or the remote device of the congestion by sending a pause frame. Upon receipt of a pause frame, the remote device stops sending any data packets, which prevents any loss of data packets during the congestion period.

Note You must not configure both IEEE 802.3z flowcontrol and quality of service (QoS) on a switch. Before configuring flowcontrol on an interface, use the **no mls qos** global configuration command to disable QoS on the switch.

Flow control can be implemented in two forms, symmetric and asymmetric. The symmetric implementation is suitable for point-to-point links, and asymmetric is suitable for hub-to-end node connections, where it is desirable for the hub to pause the end system, but not vice-versa. You use the **flowcontrol** interface configuration command to set the interface's ability to **receive** and **send** pause frames to **on**, **off**, or **desired**. The default state for Gigabit Ethernet ports is **receive off** and **send desired**. The default state for Fast Ethernet ports is **receive off** and **send off**.

Note On Catalyst 3550 switches, Gigabit Ethernet ports are capable of receiving and sending pause frames; Fast Ethernet ports can only receive pause frames. Therefore, for Fast Ethernet ports, only the conditions described with **send off** are applicable.

These rules apply to flow control settings on the device:

- **receive on** (or **desired**) and **send on**: Flow control operates in both directions; both the local and the remote devices can send pause frames to show link congestion.
- **receive on** (or **desired**) and **send desired**: The port can receive pause frames and can send pause frames if the attached device supports flow control.
- **receive on** (or **desired**) and **send off**: The port cannot send pause frames but can operate with an attached device that is required to or can send pause frames; the port can receive pause frames.
- **receive off** and **send on**: The port sends pause frames if the remote device supports flow control but cannot receive pause frames from the remote device.
- **receive off** and **send desired**: The port cannot receive pause frames but can send pause frames if the attached device supports flow control.
- **receive off** and **send off**: Flow control does not operate in either direction. In case of congestion, no indication is given to the link partner, and no pause frames are sent or received by either device.

Note For details on the command settings and the resulting flow control resolution on local and remote ports, see the **flowcontrol** interface configuration command in the command reference for this release.

Beginning in privileged EXEC mode, follow these steps to configure flow control on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**no mls qos**	Disable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the physical port to be configured, and enter interface configuration mode.
Step 4	**flowcontrol** {**receive** | **send**} {**on** | **off** | **desired**}	Configure the flow control mode for the port. **Note** The **send** keyword is not available for 10/100 Mbps ports.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interfaces** *interface-id*	Verify the interface flow control settings.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable flow control, use the **flowcontrol receive off** and **flowcontrol send off** interface configuration commands.

This example shows how to turn off all flow control on an interface:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# flowcontrol receive off
Switch(config-if)# flowcontrol send off
Switch(config-if)# end
```

Adding a Description for an Interface

You can add a description about an interface to help you remember its function. The description appears in the output of these commands: **show configuration**, **show running-config**, and **show interfaces**.

Beginning in privileged EXEC mode, follow these steps to add a description for an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**interface** *interface-id*	Specify the interface for which you are adding a description, and enter interface configuration mode.
Step 3	**description** *string*	Add a description (up to 240 characters) for an interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **description** or **show running-config**	Verify your entry.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no description** interface configuration command to delete the description.

This example shows how to add a description on an interface and to verify the description:

```
Switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface fastethernet0/4
Switch(config-if)# description Connects to Marketing
Switch(config-if)# end
Switch# show interfaces fastethernet0/4 description
Interface Status         Protocol Description
Fa0/4     up             down     Connects to Marketing
```

Configuring Layer 3 Interfaces

The switch supports three types of Layer 3 interfaces:

- SVIs: You should configure SVIs for any VLANs for which you want to route traffic. SVIs are created when you enter a VLAN ID following the **interface vlan** global configuration command. To delete an SVI, use the **no interface vlan** global configuration command. You cannot delete interface VLAN 1.

  Note When you create an SVI, it does not become active until it is associated with a physical port. For information about assigning Layer 2 ports to VLANs, see Chapter 11, "Configuring VLANs."

- Layer 3 EtherChannel ports: EtherChannel interfaces made up of routed ports.

  EtherChannel port interfaces are described in Chapter 30, "Configuring EtherChannels."

- Routed ports: Routed ports are physical ports configured to be in Layer 3 mode by using the **no switchport** interface configuration command.

Note A Layer 3 switch can have an IP address assigned to each routed port and SVI. The number of routed ports and SVIs that you can configure is not limited by software; however, the interrelationship between this number and the number of other features being configured might have an impact on CPU utilization because of hardware limitations. For more information about feature combinations, see the "Optimizing System Resources for User-Selected Features" section on page 6-26.

All Layer 3 interfaces require an IP address to route traffic. The following procedure shows how to configure an interface as a Layer 3 interface and how to assign an IP address to an interface.

Note If the physical port is in Layer 2 mode (the default), you must enter the **no switchport** interface configuration command to put the interface into Layer 3 mode. Entering a **no switchport** command disables and then re-enables the interface, which might generate messages on the device to which the interface is connected.

Beginning in privileged EXEC mode, follow these steps to configure a Layer 3 interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** {{**fastethernet** | **gigabitethernet**} *interface-id*} | {**vlan** *vlan-id*} | {**port-channel** *port-channel-number*}	Specify the interface to be configured as a Layer 3 interface, and enter interface configuration mode.
Step 3	**no switchport**	For physical ports only, enter Layer 3 mode.
Step 4	**ip address** *ip_address subnet_mask*	Configure the IP address and IP subnet.
Step 5	**no shutdown**	Enable the interface.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces** [*interface-id*] **show ip interface** [*interface-id*] **show running-config interface** [*interface-id*]	Verify the configuration.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an IP address from an interface, use the **no ip address** interface configuration command.

This example shows how to configure an interface as a routed port and to assign it an IP address:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 192.20.135.21 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# end
```

Monitoring and Maintaining the Interfaces

You can perform the tasks in these sections to monitor and maintain interfaces:

- Monitoring Interface and Controller Status, page 9-21
- Clearing and Resetting Interfaces and Counters, page 9-22
- Shutting Down and Restarting the Interface, page 9-22

Monitoring Interface and Controller Status

Commands entered at the privileged EXEC prompt display information about the interface, including the version of the software and the hardware, the controller status, and statistics about the interfaces. Table 9-2 lists some of these interface monitoring commands. (You can display the full list of **show** commands by using the **show ?** command at the privileged EXEC prompt.) These commands are fully described in the *Cisco IOS Interface Command Reference, Release 12.2*.

Table 9-2 show Commands for Interfaces

Command	Purpose
show interfaces [*interface-id*]	Display the status and configuration of all interfaces or a specific interface.
show interfaces [*interface-id*] capabilities [module {*module-number*}]	Display the capabilities of an interface. The module number is always 0. If you enter an interface ID, the **module** keyword is not visible.
show interfaces *interface-id* **status** [**err-disabled**]	Display interface status or a list of interfaces in error-disabled state.
show interfaces [*interface-id*] **switchport**	Display administrative and operational status of switching (nonrouting) ports. You can use this command to determine if a port is in routing or switching mode.
show interfaces [*interface-id*] **description**	Display the description configured on an interface or all interfaces and the interface status.
show ip interface [*interface-id*]	Display the usability status of all interfaces configured for IP or the specified interface.
show interfaces transceiver properties	(Optional) Display speed, duplex, and inline power settings on the interface.
show running-config interface [*interface-id*]	Display the running configuration in RAM for the interface.
show version	Display the hardware configuration, software version, the names and sources of configuration files, and the boot images.

For examples of output displays and definitions of output fields for the **show interfaces** privileged EXEC command, see the command reference for this release.

Clearing and Resetting Interfaces and Counters

Table 9-3 lists the privileged EXEC mode **clear** commands that you can use to clear counters and reset interfaces.

Table 9-3 Clear Commands for Interfaces

Command	Purpose
clear counters [*interface-id*]	Clear interface counters.
clear interface *interface-id*	Reset the hardware logic on an interface.
clear line [*number* | **console 0** | **vty** *number*]	Reset the hardware logic on an asynchronous serial line.

To clear the interface counters shown by the **show interfaces** privileged EXEC command, use the **clear counters** privileged EXEC command. The **clear counters** command clears all current interface counters from the interface unless optional arguments are specified to clear only a specific interface type from a specific interface number.

Note The **clear counters** privileged EXEC command does not clear counters retrieved by using Simple Network Management Protocol (SNMP), but only those seen with the **show interface** privileged EXEC command.

This example shows how to clear and reset the counters on an interface:

```
Switch# clear counters fastethernet0/5
Clear "show interface" counters on this interface [confirm] y
Switch#
*Sep 30 08:42:55: %CLEAR-5-COUNTERS: Clear counter on interface FastEthernet0/5
by vty1 (171.69.115.10)
```

Use the **clear interface** or **clear line** privileged EXEC command to clear and reset an interface or serial line. Under most circumstances, you do not need to clear the hardware logic on interfaces or serial lines.

This example shows how to clear and reset an interface:

```
Switch# clear interface fastethernet0/5
```

Shutting Down and Restarting the Interface

Shutting down an interface disables all functions on the specified interface and marks the interface as unavailable on all monitoring command displays. This information is communicated to other network servers through all dynamic routing protocols. The interface is not mentioned in any routing updates.

Beginning in privileged EXEC mode, follow these steps to shut down an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** {**vlan** *vlan-id*} | {{**fastethernet** | **gigabitethernet**} *interface-id*} | {**port-channel** *port-channel-number*}	Select the interface to be configured.
Step 3	**shutdown**	Shut down an interface.

	Command	Purpose
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entry.

Use the **no shutdown** interface configuration command to restart the interface.

This example shows how to shut down an interface:

```
Switch# configure terminal
Switch(config)# interface fastethernet0/5
Switch(config-if)# shutdown
Switch(config-if)#
*Sep 30 08:33:47: %LINK-5-CHANGED: Interface FastEthernet0/5, changed state to a
administratively down
```

This example shows how to re-enable an interface:

```
Switch# configure terminal
Switch(config)# interface fastethernet0/5
Switch(config-if)# no shutdown
Switch(config-if)#
*Sep 30 08:36:00: %LINK-3-UPDOWN: Interface FastEthernet0/5, changed state to up
```

To verify that an interface is disabled, enter the **show interfaces** privileged EXEC command. A disabled interface is shown as *administratively down* in the **show interface** command display.

CHAPTER 10

Configuring Smartports Macros

This chapter describes how to configure and apply Smartports macros on the Catalyst 3550 switch.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding Smartports Macros, page 10-1
- Configuring Smartports Macros, page 10-2
- Displaying Smartports Macros, page 10-8

Understanding Smartports Macros

Smartports macros provide a convenient way to save and share common configurations. You can use Smartports macros to enable features and settings based on the location of a switch in the network and for mass configuration deployments across the network.

Each Smartports macro is a set of CLI commands that you define. Smartports macros do not contain new CLI commands; they are simply a group of existing CLI commands.

When you apply a Smartports macro on an interface, the CLI commands within the macro are configured on the interface. When the macro is applied to an interface, the existing interface configurations are not lost. The new commands are added to the interface and are saved in the running configuration file.

There are Cisco-default Smartports macros embedded in the switch software (see Table 10-1). You can display these macros and the commands they contain by using the **show parser macro** user EXEC command.

Table 10-1 Cisco-Default Smartports Macros

Macro Name[1]	Description
cisco-global	Use this global configuration macro to enable load balancing across VLANs, provide rapid convergence of spanning-tree instances and to enable port error recovery.
cisco-desktop	Use this interface configuration macro for increased network security and reliability when connecting a desktop device, such as a PC, to a switch port.

Table 10-1 Cisco-Default Smartports Macros (continued)

Macro Name[1]	Description
cisco-phone	Use this interface configuration macro when connecting a desktop device such as a PC with a Cisco IP Phone to a switch port. This macro is an extension of the **cisco-desktop** macro and provides the same security and resiliency features, but with the addition of dedicated voice VLANs to ensure proper treatment of delay-sensitive voice traffic.
cisco-switch	Use this interface configuration macro when connecting an access switch and a distribution switch or between access switches connected using GigaStack modules or GBICs.
cisco-router	Use this interface configuration macro when connecting the switch and a WAN router.
cisco-wireless	Use this interface configuration macro when connecting the switch and a wireless access point.

1. Cisco-default Smartports macros vary depending on the software version running on your switch.

Cisco also provides a collection of pretested, Cisco-recommended baseline configuration templates for Catalyst switches. The online reference guide templates provide the CLI commands that you can use to create Smartports macros based on the usage of the port. You can use the configuration templates to create Smartports macros to build and deploy Cisco-recommended network designs and configurations. For more information about Cisco-recommended configuration templates, see this Smartports website:

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

Configuring Smartports Macros

You can create a new Smartports macro or use an existing macro as a template to create a new macro that is specific to your application. After you create the macro, you can apply it globally to a switch or to a switch interface or range of interfaces.

This section includes information about:

- Default Smartports Macro Configuration, page 10-2
- Smartports Macro Configuration Guidelines, page 10-3
- Creating Smartports Macros, page 10-4
- Applying Smartports Macros, page 10-4
- Applying Cisco-Default Smartports Macros, page 10-6

Default Smartports Macro Configuration

There are no Smartports macros enabled.

Smartports Macro Configuration Guidelines

Follow these guidelines when configuring macros on your switch:

- When creating a macro, do not use the **exit** or **end** commands or change the command mode by using **interface** *interface-id*. This could cause commands that follow **exit**, **end**, or **interface** *interface-id* to execute in a different command mode.
- When creating a macro, all CLI commands should be in the same configuration mode.
- When creating a macro that requires the assignment of unique values, use the **parameter** *value* keywords to designate values specific to the interface. Keyword matching is case sensitive. All matching occurrences of the keyword are replaced with the corresponding value. Any full match of a keyword, even if it is part of a larger string, is considered a match and is replaced by the corresponding value.
- Macro names are case sensitive. For example, the commands **macro name Sample-Macro** and **macro name sample-macro** will result in two separate macros.
- Some macros might contain keywords that require a parameter value. You can use the **macro global apply** *macro-name* **?** global configuration command or the **macro apply** *macro-name* **?** interface configuration command to display a list of any required values in the macro. If you apply a macro without entering the keyword values, the commands are invalid and are not applied.
- When a macro is applied globally to a switch or to a switch interface, all existing configuration on the interface is retained. This is helpful when applying an incremental configuration.
- If you modify a macro definition by adding or deleting commands, the changes are not reflected on the interface where the original macro was applied. You need to reapply the updated macro on the interface to apply the new or changed commands.
- You can use the **macro global trace** *macro-name* global configuration command or the **macro trace** *macro-name* interface configuration command to apply and debug a macro to find any syntax or configuration errors. If a command fails because of a syntax error or a configuration error, the macro continues to apply the remaining commands.
- Some CLI commands are specific to certain interface types. If a macro is applied to an interface that does not accept the configuration, the macro will fail the syntax check or the configuration check, and the switch will return an error message.
- Applying a macro to an interface range is the same as applying a macro to a single interface. When you use an interface range, the macro is applied sequentially to each interface within the range. If a macro command fails on one interface, it is still applied to the remaining interfaces.
- When you apply a macro to a switch or a switch interface, the macro name is automatically added to the switch or interface. You can display the applied commands and macro names by using the **show running-config** user EXEC command.

There are Cisco-default Smartports macros embedded in the switch software. You can display these macros and the commands they contain by using the **show parser macro** user EXEC command.

Follow these guidelines when you apply a Cisco-default Smartports macro on an interface:

- Display all macros on the switch by using the **show parser macro** user EXEC command. Display the contents of a specific macro by using the **show parser macro name** *macro-name* user EXEC command.
- Keywords that begin with **$** mean that a unique parameter value is required. Append the Cisco-default macro with the required values by using the **parameter** *value* keywords.

  The Cisco-default macros use the **$** character to help identify required keywords. There is no restriction on using the **$** character to define keywords when you create a macro.

Creating Smartports Macros

Beginning in privileged EXEC mode, follow these steps to create a Smartports macro:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	macro name *macro-name*	Create a macro definition, and enter a macro name. A macro definition can contain up to 3000 characters. Enter the macro commands with one command per line. Use the @ character to end the macro. Use the **#** character at the beginning of a line to enter comment text within the macro. (Optional) You can define keywords within a macro by using a help string to specify the keywords. Enter **# macro keywords** *word* to define the keywords that are available for use with the macro. Separated by a space, you can enter up to three help string keywords in a macro. Macro names are case sensitive. For example, the commands **macro name Sample-Macro** and **macro name sample-macro** will result in two separate macros. We recommend that you do not use the **exit** or **end** commands or change the command mode by using **interface** *interface-id* in a macro. This could cause any commands following **exit**, **end**, or **interface** *interface-id* to execute in a different command mode. For best results, all commands in a macro should be in the same configuration mode.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show parser macro** name *macro-name*	Verify that the macro was created.

The **no** form of the **macro name** global configuration command only deletes the macro definition. It does not affect the configuration of those interfaces on which the macro is already applied.

This example shows how to create a macro that defines the switchport access VLAN and the number of secure MAC addresses and also includes two help string keywords by using **# macro keywords**:

```
Switch(config)# macro name test
switchport access vlan $VLANID
switchport port-security maximum $MAX
#macro keywords $VLANID $MAX
@
```

Applying Smartports Macros

Beginning in privileged EXEC mode, follow these steps to apply a Smartports macro:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**macro global** {**apply** | **trace**} *macro-name* [**parameter** {*value*}] [**parameter** {*value*}] [**parameter** {*value*}]	Apply each individual command defined in the macro to the switch by entering **macro global apply** *macro-name*. Specify **macro global trace** *macro-name* to apply and debug a macro to find any syntax or configuration errors. (Optional) Specify unique parameter values that are specific to the switch. You can enter up to three keyword-value pairs. Parameter keyword matching is case sensitive. All matching occurrences of the keyword are replaced with the corresponding value. Some macros might contain keywords that require a parameter value. You can use the **macro global apply** *macro-name* **?** command to display a list of any required values in the macro. If you apply a macro without entering the keyword values, the commands are invalid and are not applied.
Step 3	macro global description *text*	(Optional) Enter a description about the macro that is applied to the switch.
Step 4	**interface** *interface-id*	(Optional) Enter interface configuration mode, and specify the interface on which to apply the macro.
Step 5	**default interface** *interface-id*	(Optional) Clear all configuration from the specified interface.
Step 6	**macro** {**apply** | **trace**} *macro-name* [**parameter** {*value*}] [**parameter** {*value*}] [**parameter** {*value*}]	Apply each individual command defined in the macro to the interface by entering **macro apply** *macro-name*. Specify **macro trace** *macro-name* to apply and debug a macro to find any syntax or configuration errors. (Optional) Specify unique parameter values that are specific to the interface. You can enter up to three keyword-value pairs. Parameter keyword matching is case sensitive. All matching occurrences of the keyword are replaced with the corresponding value. Some macros might contain keywords that require a parameter value. You can use the **macro apply** *macro-name* **?** command to display a list of any required values in the macro. If you apply a macro without entering the keyword values, the commands are invalid and are not applied.
Step 7	macro description *text*	(Optional) Enter a description about the macro that is applied to the interface.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	show parser macro description [**interface** *interface-id*]	Verify that the macro is applied to the interface.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

You can delete a global macro-applied configuration on a switch only by entering the **no** version of each command that is in the macro. You can delete a macro-applied configuration on an interface by entering the **default interface** *interface-id* interface configuration command.

This example shows how to apply the user-created macro called **snmp**, to set the host name address to **test-server** and to set the IP precedence value to **7**:

```
Switch(config)# macro global apply snmp ADDRESS test-server VALUE 7
```

This example shows how to debug the user-created macro called **snmp** by using the **macro global trace** global configuration command to find any syntax or configuration errors in the macro when it is applied to the switch.

```
Switch(config)# macro global trace snmp VALUE 7
Applying command...'snmp-server enable traps port-security'
Applying command...'snmp-server enable traps linkup'
Applying command...'snmp-server enable traps linkdown'
Applying command...'snmp-server host'
%Error Unknown error.
Applying command...'snmp-server ip precedence 7'
```

This example shows how to apply the user-created macro called **desktop-config** and to verify the configuration.

```
Switch(config)# interface fastethernet0/2
Switch(config-if)# macro apply desktop-config
Switch(config-if)# end
Switch# show parser macro description
Interface    Macro Description
---------------------------------------------------------------
Gi0/2      desktop-config
---------------------------------------------------------------
```

This example shows how to apply the user-created macro called **desktop-config** and to replace all occurrences of VLAN 1 with VLAN 25:

```
Switch(config-if)# macro apply desktop-config vlan 25
```

Applying Cisco-Default Smartports Macros

Beginning in privileged EXEC mode, follow these steps to apply a Smartports macro:

	Command	Purpose
Step 1	show parser macro	Display the Cisco-default Smartports macros embedded in the switch software.
Step 2	show parser macro name *macro-name*	Display the specific macro that you want to apply.
Step 3	**configure terminal**	Enter global configuration mode.
Step 4	**macro global** {**apply** | **trace**} *macro-name* [**parameter** {*value*}] [**parameter** {*value*}] [**parameter** {*value*}]	Append the Cisco-default macro with the required values by using the **parameter** *value* keywords and apply the macro to the switch. Keywords that begin with **$** mean that a unique parameter value is required. You can use the **macro global apply** *macro-name* **?** command to display a list of any required values in the macro. If you apply a macro without entering the keyword values, the commands are invalid and are not applied.
Step 5	**interface** *interface-id*	(Optional) Enter interface configuration mode, and specify the interface on which to apply the macro.
Step 6	**default interface** *interface-id*	(Optional) Clear all configuration from the specified interface.

	Command	Purpose
Step 7	**macro** {**apply** | **trace**} *macro-name* [**parameter** {*value*}] [**parameter** {*value*}] [**parameter** {*value*}]	Append the Cisco-default macro with the required values by using the **parameter** *value* keywords, and apply the macro to the interface. Keywords that begin with **$** mean that a unique parameter value is required. You can use the **macro apply** *macro-name* **?** command to display a list of any required values in the macro. If you apply a macro without entering the keyword values, the commands are invalid and are not applied.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	show running-config **interface** *interface-id*	Verify that the macro is applied to an interface.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

You can delete a global macro-applied configuration on a switch only by entering the **no** version of each command that is in the macro. You can delete a macro-applied configuration on an interface by entering the **default interface** *interface-id* interface configuration command.

This example shows how to display the **cisco-desktop** macro, how to apply the macro, and to set the access VLAN ID to 25 on an interface:

```
Switch# show parser macro cisco-desktop
------------------------------------------------------------
Macro name : cisco-desktop
Macro type : default

# Basic interface - Enable data VLAN only
# Recommended value for access vlan (AVID) should not be 1
switchport access vlan $AVID
switchport mode access

# Enable port security limiting port to a single
# MAC address -- that of desktop
switchport port-security
switchport port-security maximum 1

# Ensure port-security age is greater than one minute
# and use inactivity timer
switchport port-security violation restrict
switchport port-security aging time 2
switchport port-security aging type inactivity

# Configure port as an edge network port
spanning-tree portfast
spanning-tree bpduguard enable
------------------------------------------------------------
Switch#
Switch# configure terminal
Switch(config)# fastethernet0/4
Switch(config-if)# macro apply cisco-desktop $AVID 25
```

Displaying Smartports Macros

To display the Smartports macros, use one or more of the privileged EXEC commands in Table 10-2.

Table 10-2 Commands for Displaying Smartports Macros

Command	Purpose
show parser macro	Displays all configured macros.
show parser macro name *macro-name*	Displays a specific macro.
show parser macro brief	Displays the configured macro names.
show parser macro description [**interface** *interface-id*]	Displays the macro description for all interfaces or for a specified interface.

CHAPTER 11

Configuring VLANs

This chapter describes how to configure normal-range VLANs (VLAN IDs 1 to 1005) and extended-range VLANs (VLAN IDs 1006 to 4094) on your Catalyst 3550 switch. It includes information about VLAN modes and the VLAN Membership Policy Server (VMPS).

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

The chapter includes these sections:

- Understanding VLANs, page 11-1
- Configuring Normal-Range VLANs, page 11-4
- Configuring Extended-Range VLANs, page 11-11
- Displaying VLANs, page 11-14
- Configuring VLAN Trunks, page 11-15
- Configuring VMPS, page 11-26

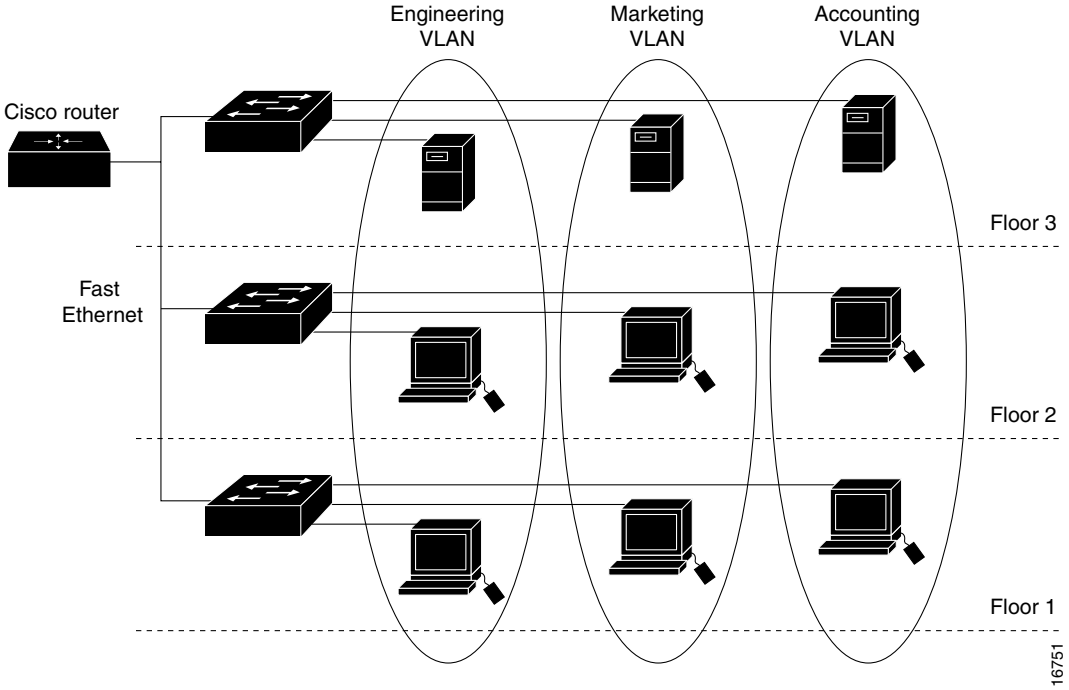
Understanding VLANs

A VLAN is a switched network that is logically segmented by function, project team, or application, without regard to the physical locations of the users. VLANs have the same attributes as physical LANs, but you can group end stations even if they are not physically located on the same LAN segment. Any switch port can belong to a VLAN, and unicast, broadcast, and multicast packets are forwarded and flooded only to end stations in the VLAN. Each VLAN is considered a logical network, and packets destined for stations that do not belong to the VLAN must be forwarded through a router or bridge as shown in Figure 11-1. Because a VLAN is considered a separate logical network, it contains its own MIB information and can support its own implementation of spanning tree. See Chapter 15, "Configuring STP" and Chapter 16, "Configuring MSTP."

Note Before you create VLANs, you must decide whether to use VLAN Trunking Protocol (VTP) to maintain global VLAN configuration for your network. For more information on VTP, see Chapter 12, "Configuring VTP."

Figure 11-1 shows an example of VLANs segmented into logically defined networks.

Figure 11-1 VLANs as Logically Defined Networks



VLANs are often associated with IP subnetworks. For example, all the end stations in a particular IP subnet belong to the same VLAN. Interface VLAN membership on the switch is assigned manually on an interface-by-interface basis. When you assign switch interfaces to VLANs by using this method, it is known as interface-based, or static, VLAN membership.

Traffic between VLANs must be routed or fallback bridged. A Catalyst 3550 switch can route traffic between VLANs by using switch virtual interfaces (SVIs). An SVI must be explicitly configured and assigned an IP address to route traffic between VLANs. For more information, see the "Switch Virtual Interfaces" section on page 9-4 and the "Configuring Layer 3 Interfaces" section on page 9-19.

Supported VLANs

The Catalyst 3550 switch supports 1005 VLANs in VTP client, server, and transparent modes. VLANs are identified with a number from 1 to 4094. VLAN IDs 1002 through 1005 are reserved for Token Ring and FDDI VLANs. VTP only learns normal-range VLANs, with VLAN IDs 1 to 1005; VLAN IDs greater than 1005 are extended-range VLANs and are not stored in the VLAN database. The switch must be in VTP transparent mode when you create VLAN IDs from 1006 to 4094.

The switch supports per-VLAN spanning-tree plus (PVST+) and rapid PVST+ with a maximum of 128 spanning-tree instances. One spanning-tree instance is allowed per VLAN. See the "Normal-Range VLAN Configuration Guidelines" section on page 11-5 for more information about the number of spanning-tree instances and the number of VLANs. The switch supports both Inter-Switch Link (ISL) and IEEE 802.1Q trunking methods for sending VLAN traffic over Ethernet ports.

VLAN Port Membership Modes

You configure a port to belong to a VLAN by assigning a membership mode that determines the kind of traffic the port carries and the number of VLANs to which it can belong. Table 11-1 lists the membership modes and membership and VTP characteristics.

Table 11-1 Port Membership Modes

Membership Mode	VLAN Membership Characteristics	VTP Characteristics
Static-access	A static-access port can belong to one VLAN and is manually assigned to that VLAN. For more information, see the "Assigning Static-Access Ports to a VLAN" section on page 11-10.	VTP is not required. If you do not want VTP to globally propagate information, set the VTP mode to transparent to disable VTP. To participate in VTP, there must be at least one trunk port on the switch connected to a trunk port of a second switch.
Trunk (ISL or IEEE 802.1Q)	A trunk port is a member of all VLANs by default, including extended-range VLANs, but membership can be limited by configuring the allowed-VLAN list. You can also modify the pruning-eligible list to block flooded traffic to VLANs on trunk ports that are included in the list. For information about configuring trunk ports, see the "Configuring an Ethernet Interface as a Trunk Port" section on page 11-18.	VTP is recommended but not required. VTP maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs on a network-wide basis. VTP exchanges VLAN configuration messages with other switches over trunk links.
Dynamic access	A dynamic-access port can belong to one VLAN (VLAN ID 1 to 4094), and is dynamically assigned by a VMPS. The VMPS can be a Catalyst 5000 or Catalyst 6500 series switch, for example, but never a Catalyst 3550 switch. You can have dynamic-access ports and trunk ports on the same switch, but you must connect the dynamic-access port to an end station and not to another switch. For configuration information, see the "Configuring Dynamic Access Ports on VMPS Clients" section on page 11-30.	VTP is required. Configure the VMPS and the client with the same VTP domain name. You can change the reconfirmation interval and retry count on the VMPS client switch.
Voice VLAN	A voice VLAN port is an access port attached to a Cisco IP Phone, configured to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone. For more information about voice VLAN ports, see Chapter 13, "Configuring Voice VLAN."	VTP is not required; it has no effect on voice VLAN.
Tunnel (**dot1q-tunnel**)	Tunnel ports are used for 802.1Q tunneling to maintain customer VLAN integrity across a service provider network. You configure a tunnel port on an edge switch in the service provider network and connect it to an 802.1Q trunk port on a customer interface, creating an asymmetric link. A tunnel port belongs to a single VLAN that is dedicated to tunneling. For more information about tunnel ports, see Chapter 14, "Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling."	VTP is not required. You manually assign the tunnel port to a VLAN by using the **switchport access vlan** interface configuration command.

For more detailed definitions of the modes and their functions, see Table 11-4 on page 11-17.

When a port belongs to a VLAN, the switch learns and manages the addresses associated with the port on a per-VLAN basis. For more information, see the "Managing the MAC Address Table" section on page 6-19.

Configuring Normal-Range VLANs

Normal-range VLANs are VLANs with VLAN IDs 1 to 1005. If the switch is in VTP server or transparent mode, you can add, modify or remove configurations for VLANs 2 to 1001 in the VLAN database. (VLAN IDs 1 and 1002 to 1005 are automatically created and cannot be removed.)

Note When the switch is in VTP transparent mode, you can also create extended-range VLANs (VLANs with IDs from 1006 to 4094), but these VLANs are not saved in the VLAN database. See the "Configuring Extended-Range VLANs" section on page 11-11.

Configurations for VLAN IDs 1 to 1005 are written to the file *vlan.dat* (VLAN database), and you can display them by entering the **show vlan** privileged EXEC command. The *vlan.dat* file is stored in Flash memory.

Caution You can cause inconsistency in the VLAN database if you attempt to manually delete the *vlan.dat* file. If you want to modify the VLAN configuration, use the commands described in these sections and in the command reference for this release. To change the VTP configuration, see Chapter 12, "Configuring VTP."

You use the interface configuration mode to define the port membership mode and to add and remove ports from VLANs. The results of these commands are written to the running-configuration file, and you can display the file by entering the **show running-config** privileged EXEC command.

You can set these parameters when you create a new normal-range VLAN or modify an existing VLAN in the VLAN database:

- VLAN ID
- VLAN name
- VLAN type (Ethernet, Fiber Distributed Data Interface [FDDI], FDDI network entity title [NET], TrBRF, or TrCRF, Token Ring, Token Ring-Net)
- VLAN state (active or suspended)
- Maximum transmission unit (MTU) for the VLAN
- Security Association Identifier (SAID)
- Bridge identification number for TrBRF VLANs
- Ring number for FDDI and TrCRF VLANs
- Parent VLAN number for TrCRF VLANs
- Spanning Tree Protocol (STP) type for TrCRF VLANs
- VLAN number to use when translating from one VLAN type to another

Note This section does not provide configuration details for most of these parameters. For complete information on the commands and parameters that control VLAN configuration, see the command reference for this release.

This section includes information about these topics about normal-range VLANs:

- Token Ring VLANs, page 11-5
- Normal-Range VLAN Configuration Guidelines, page 11-5
- VLAN Configuration Mode Options, page 11-6
- Saving VLAN Configuration, page 11-7
- Default Ethernet VLAN Configuration, page 11-7
- Creating or Modifying an Ethernet VLAN, page 11-8
- Deleting a VLAN, page 11-10
- Assigning Static-Access Ports to a VLAN, page 11-10

Token Ring VLANs

Although the switch does not support Token Ring connections, a remote device such as a Catalyst 5000 series switch with Token Ring connections could be managed from one of the supported switches. Switches running VTP version 2 advertise information about these Token Ring VLANs:

- Token Ring TrBRF VLANs
- Token Ring TrCRF VLANs

For more information on configuring Token Ring VLANs, see the *Catalyst 5000 Series Software Configuration Guide*.

Normal-Range VLAN Configuration Guidelines

Follow these guidelines when creating and modifying normal-range VLANs in your network:

- The switch supports 1005 VLANs in VTP client, server, and transparent modes.
- Normal-range VLANs are identified with a number between 1 and 1001. VLAN numbers 1002 through 1005 are reserved for Token Ring and FDDI VLANs.
- VLAN configuration for VLANs 1 to 1005 are always saved in the VLAN database. If VTP mode is transparent, VTP and VLAN configuration are also saved in the switch running configuration file.
- The switch also supports VLAN IDs 1006 through 4094 in VTP transparent mode (VTP disabled). These are extended-range VLANs, and configuration options are limited. Extended-range VLANs are not saved in the VLAN database. See the "Configuring Extended-Range VLANs" section on page 11-11.
- Before you can create a VLAN, the switch must be in VTP server mode or VTP transparent mode. If the switch is a VTP server, you must define a VTP domain, or VTP will not function.
- The switch does not support Token Ring or FDDI media. The switch does not forward FDDI, FDDI-Net, TrCRF, or TrBRF traffic, but it does propagate the VLAN configuration through VTP.

- The switch supports 128 spanning-tree instances. If a switch has more active VLANs than supported spanning-tree instances, spanning tree can be enabled on 128 VLANs and is disabled on the remaining VLANs. If you have already used all available spanning-tree instances on a switch, adding another VLAN anywhere in the VTP domain creates a VLAN on that switch that is not running spanning tree. If you have the default allowed list on the trunk ports of that switch (which is to allow all VLANs), the new VLAN is carried on all trunk ports. Depending on the topology of the network, this could create a loop in the new VLAN that would not be broken, particularly if there are several adjacent switches that all have run out of spanning-tree instances. You can prevent this possibility by setting allowed lists on the trunk ports of switches that have used up their allocation of spanning-tree instances.

  If the number of VLANs on the switch exceeds the number of supported spanning tree instances, we recommend that you configure the IEEE 802.1S Multiple STP (MSTP) on your switch to map multiple VLANs to a single STP instance. For more information about MSTP, see Chapter 16, "Configuring MSTP."

VLAN Configuration Mode Options

You can configure normal-range VLANs (with VLAN IDs 1 to 1005) by using these two configuration modes:

- VLAN Configuration in config-vlan Mode, page 11-6

  You access config-vlan mode by entering the **vlan** *vlan-id* global configuration command.

- VLAN Configuration in VLAN Configuration Mode, page 11-6

  You access VLAN database configuration mode by entering the **vlan database** privileged EXEC command.

VLAN Configuration in config-vlan Mode

To access config-vlan mode, enter the **vlan** global configuration command with a VLAN ID. Enter a new VLAN ID to create a VLAN or with an existing VLAN ID to modify the VLAN. You can use the default VLAN configuration (Table 11-2) or enter multiple commands to configure the VLAN. For more information about commands available in this mode, see the **vlan** global configuration command description in the command reference for this release. When you have finished the configuration, you must exit config-vlan mode for the configuration to take effect. To display the VLAN configuration, enter the **show vlan** privileged EXEC command.

You must use this config-vlan mode when creating extended-range VLANs (VLAN IDs greater than 1005). See the "Configuring Extended-Range VLANs" section on page 11-11.

VLAN Configuration in VLAN Configuration Mode

To access VLAN configuration mode, enter the **vlan database** privileged EXEC command. Then enter the **vlan** command with a new VLAN ID to create a VLAN or with an existing VLAN ID to modify the VLAN. You can use the default VLAN configuration (Table 11-2) or enter multiple commands to configure the VLAN. For more information about keywords available in this mode, see the **vlan** VLAN configuration command description in the command reference for this release. When you have finished the configuration, you must enter **apply** or **exit** for the configuration to take effect. When you enter the **exit** command, it applies all commands and updates the VLAN database. VTP messages are sent to other switches in the VTP domain, and the privileged EXEC mode prompt appears.

Saving VLAN Configuration

The configurations of VLAN IDs 1 to 1005 are always saved in the VLAN database (vlan.dat file). If VTP mode is transparent, they are also saved in the switch running configuration file, and you can enter the **copy running-config startup-config** privileged EXEC command to save the configuration in the startup configuration file. You can use the **show running-config vlan** privileged EXEC command to display the switch running configuration file. To display the VLAN configuration, enter the **show vlan** privileged EXEC command.

When you save VLAN and VTP information (including extended-range VLAN configuration information) in the startup configuration file and reboot the switch, the switch configuration is determined as follows:

- If the VTP mode is transparent in the startup configuration, and the VLAN database and the VTP domain name from the VLAN database matches that in the startup configuration file, the VLAN database is ignored (cleared). The VTP and VLAN configurations in the startup configuration file are used. The VLAN database revision number remains unchanged in the VLAN database.
- If the VTP mode or domain name in the startup configuration does not match the VLAN database, the domain name and VTP mode and configuration for the first 1005 VLAN IDs use the VLAN database information.
- If the VTP mode is server, the domain name and VLAN configuration for the first 1005 VLAN IDs use the VLAN database information.
- If the switch is running Cisco IOS Release 12.1(9)EA1 or later and you use an older startup configuration file to boot up the switch, the configuration file does not contain VTP or VLAN information, and the switch uses the VLAN database configurations.
- If the switch is running a Cisco IOS release earlier than 12.1(9)EA1 and you use a startup configuration file from Cisco IOS Release 12.1(9)EA1 or later to boot up the switch, the image on the switch does not recognize the VLAN and VTP configurations in the startup configuration file, so the switch uses the VLAN database configuration.

Caution If the VLAN database configuration is used at startup and the startup configuration file contains extended-range VLAN configuration, this information is lost when the system boots up.

Default Ethernet VLAN Configuration

Table 11-2 shows the default configuration for Ethernet VLANs.

Note The switch supports Ethernet interfaces exclusively. Because FDDI and Token Ring VLANs are not locally supported, you only configure FDDI and Token Ring media-specific characteristics for VTP global advertisements to other switches.

Table 11-2 Ethernet VLAN Defaults and Ranges

Parameter	Default	Range
VLAN name	*VLANxxxx*, where *xxxx* represents four numeric digits (including leading zeros) equal to the VLAN ID number	No range
IEEE 802.10 SAID	100001 (100000 plus the VLAN ID)	1 to 4294967294
MTU size	1500	1500 to 18190
Translational bridge 1	0	0–1005
Translational bridge 2	0	0–1005
VLAN state	active	active, suspend
Remote SPAN	disabled	enabled, disabled

Creating or Modifying an Ethernet VLAN

Each Ethernet VLAN in the VLAN database has a unique, 4-digit ID that can be a number from 1 to 1001. VLAN IDs 1002 to 1005 are reserved for Token Ring and FDDI VLANs. To create a normal-range VLAN to be added to the VLAN database, assign a number and name to the VLAN.

Note When the switch is in VTP transparent mode, you can assign VLAN IDs greater than 1006, but they are not added to the VLAN database. See the "Configuring Extended-Range VLANs" section on page 11-11.

For the list of default parameters that are assigned when you add a VLAN, see the "Configuring Normal-Range VLANs" section on page 11-4.

Beginning in privileged EXEC mode, follow these steps to use config-vlan mode to create or modify an Ethernet VLAN:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vlan** *vlan-id*	Enter a VLAN ID, and enter config-vlan mode. Enter a new VLAN ID to create a VLAN, or enter an existing VLAN ID to modify a VLAN. **Note** The available VLAN ID range for this command is 1 to 4094. For information about adding VLAN IDs greater than 1005 (extended-range VLANs), see the "Configuring Extended-Range VLANs" section on page 11-11.
Step 3	**name** *vlan-name*	(Optional) Enter a name for the VLAN. If no name is entered, the default is to append the *vlan-id* with leading zeros to the word VLAN. For example, VLAN0004 is a default VLAN name for VLAN 4.
Step 4	**mtu** *mtu-size*	(Optional) Change the MTU size (or other VLAN characteristic).

	Command	Purpose
Step 5	remote-span	(Optional) Configure the VLAN as the RSPAN VLAN for a remote SPAN session. For more information on remote SPAN, see Chapter 24, "Configuring SPAN and RSPAN."
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show vlan** {**name** *vlan-name* | **id** *vlan-id*}	Verify your entries.
Step 8	**copy running-config startup config**	(Optional) If the switch is in VTP transparent mode, the VLAN configuration is saved in the running configuration file as well as in the VLAN database. This saves the configuration in the switch startup configuration file.

To return the VLAN name to the default settings, use the **no vlan name**, **no vlan mtu**, or **no remote span** config-vlan commands.

This example shows how to use config-vlan mode to create Ethernet VLAN 20, name it *test20,* and add it to the VLAN database:

```
Switch# configure terminal
Switch(config)# vlan 20
Switch(config-vlan)# name test20
Switch(config-vlan)# end
```

Beginning in privileged EXEC mode, follow these steps to use VLAN configuration mode to create or modify an Ethernet VLAN:

	Command	Purpose
Step 1	**vlan database**	Enter VLAN database configuration mode.
Step 2	**vlan** *vlan-id* **name** *vlan-name*	Add an Ethernet VLAN by assigning a number to it. The range is 1 to 1001; do not enter leading zeros. If no name is entered, the default is to append the *vlan-id* with leading zeros to the word VLAN. For example, VLAN0004 is a default VLAN name for VLAN 4.
Step 3	**vlan** *vlan-id* **mtu** *mtu-size*	(Optional) To modify a VLAN, identify the VLAN and change a characteristic, such as the MTU size.
Step 4	**exit**	Update the VLAN database, propagate it throughout the administrative domain, and return to privileged EXEC mode.
Step 5	**show vlan** {**name** *vlan-name* | **id** *vlan-id*}	Verify your entries.
Step 6	**copy running-config startup config**	(Optional) If the switch is in VTP transparent mode, the VLAN configuration is saved in the running configuration file as well as in the VLAN database. This saves the configuration in the switch startup configuration file.

Note You cannot configure an RSPAN VLAN in VLAN database configuration mode.

To return the VLAN name to the default settings, use the **no vlan** *vlan-id* **name** VLAN configuration command.

This example shows how to use VLAN database configuration mode to create Ethernet VLAN 20, name it *test20*, and add it to the VLAN database:

```
Switch# vlan database
Switch(vlan)# vlan 20 name test20
Switch(vlan)# exit
APPLY completed.
Exiting....
Switch#
```

Deleting a VLAN

When you delete a VLAN from a switch that is in VTP server mode, the VLAN is removed from the VLAN database for all switches in the VTP domain. When you delete a VLAN from a switch that is in VTP transparent mode, the VLAN is deleted only on that specific switch.

You cannot delete the default VLANs for the different media types: Ethernet VLAN 1 and FDDI or Token Ring VLANs 1002 to 1005.

Caution When you delete a VLAN, any ports assigned to that VLAN become inactive. They remain associated with the VLAN (and thus inactive) until you assign them to a new VLAN.

Beginning in privileged EXEC mode, follow these steps to delete a VLAN on the switch by using global configuration mode:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no vlan** *vlan-id*	Remove the VLAN by entering the VLAN ID.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show vlan brief**	Verify the VLAN removal.
Step 5	**copy running-config startup config**	(Optional) If the switch is in VTP transparent mode, the VLAN configuration is saved in the running configuration file as well as in the VLAN database. This saves the configuration in the switch startup configuration file.

To delete a VLAN in VLAN database configuration mode, use the **vlan database** privileged EXEC command to enter VLAN database configuration mode and the **no vlan** *vlan-id* VLAN configuration command.

Assigning Static-Access Ports to a VLAN

You can assign a static-access port to a VLAN without having VTP globally propagate VLAN configuration information by disabling VTP (VTP transparent mode). If you are assigning a port on a cluster member switch to a VLAN, first use the **rcommand** privileged EXEC command to log in to the member switch.

Note If you assign an interface to a VLAN that does not exist, the new VLAN is created. (See the "Creating or Modifying an Ethernet VLAN" section on page 11-8.)

Beginning in privileged EXEC mode, follow these steps to assign a port to a VLAN in the VLAN database:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**interface** *interface-id*	Enter the interface to be added to the VLAN.
Step 3	**switchport mode access**	Define the VLAN membership mode for the port (Layer 2 access port).
Step 4	**switchport access vlan** *vlan-id*	Assign the port to a VLAN. Valid VLAN IDs are 1 to 4094.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config interface** *interface-id*	Verify the VLAN membership mode of the interface.
Step 7	**show interfaces** *interface-id* **switchport**	Verify your entries in the *Administrative Mode* and the *Access Mode VLAN* fields of the display.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return an interface to its default configuration, use the **default interface** *interface-id* interface configuration command.

This example shows how to configure a port as an access port in VLAN 2:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface fastethernet0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 2
Switch(config-if)# end
Switch#
```

Configuring Extended-Range VLANs

When the switch is in VTP transparent mode (VTP disabled), you can create extended-range VLANs (in the range 1006 to 4094 for any switch port commands that allow VLAN IDs). Enter the **vlan** *vlan-id* global configuration command to access config-vlan mode and to configure extended-range VLANs. The VLAN database configuration mode (that you access by entering the **vlan database** privileged EXEC command) does not support the extended range.

Extended-range VLAN configurations are not stored in the VLAN database. Because VTP mode is transparent, they are stored in the switch running configuration file. You can save the configuration in the startup configuration file by using the **copy running-config startup-config** privileged EXEC command.

Note Although the switch supports 4094 VLAN IDs, see the "Supported VLANs" section on page 11-2 for the actual number of VLANs supported.

This section includes this information about extended-range VLANs:

- Default VLAN Configuration, page 11-12
- Extended-Range VLAN Configuration Guidelines, page 11-12
- Creating an Extended-Range VLAN, page 11-13
- Creating an Extended-Range VLAN with an Internal VLAN ID, page 11-14

Default VLAN Configuration

See Table 11-2 on page 11-8 for the default configuration for Ethernet VLANs. You can change only the MTU size on extended-range VLANs; all other characteristics must remain at the default state.

Extended-Range VLAN Configuration Guidelines

Follow these guidelines when creating extended-range VLANs:

- To add an extended-range VLAN, you must use the **vlan** *vlan-id* global configuration command and access config-vlan mode. You cannot add extended-range VLANs in VLAN database configuration mode (accessed by entering the **vlan database** privileged EXEC command).
- VLAN IDs in the extended range are not saved in the VLAN database and are not recognized by VTP.
- You cannot include extended-range VLANs in the pruning eligible range.
- The switch must be in VTP transparent mode when you create extended-range VLANs. If VTP mode is server or client, an error message is generated, and the extended-range VLAN is rejected.
- You can set the VTP mode to transparent in global configuration mode or in VLAN database configuration mode. See the "Disabling VTP (VTP Transparent Mode)" section on page 12-12. You should save this configuration to the startup configuration so that the switch will boot up in VTP transparent mode. Otherwise, you will lose extended-range VLAN configuration if the switch resets.
- VLANs in the extended range are not supported by VQP. They cannot be configured by VMPS.
- STP is enabled by default on extended-range VLANs, but you can disable it by using the **no spanning-tree vlan** *vlan-id* global configuration command. When the maximum number of spanning-tree instances (128) are on the switch, spanning tree is disabled on any newly created VLANs. If the number of VLANs on the switch exceeds the maximum number of spanning tree instances, we recommend that you configure the IEEE 802.1S Multiple STP (MSTP) on your switch to map multiple VLANs to a single STP instance. For more information about MSTP, see Chapter 16, "Configuring MSTP."
- Each routed port on a Catalyst 3550 switch creates an internal VLAN for its use. These internal VLANs use extended-range VLAN numbers, and the internal VLAN ID cannot be used for an extended-range VLAN. If you try to create an extended-range VLAN with a VLAN ID that is already allocated as an internal VLAN, an error message is generated, and the command is rejected.
  - Because internal VLAN IDs are in the lower part of the extended range, we recommend that you create extended-range VLANs beginning from the highest number (4094) and moving to the lowest (1006) to reduce the possibility of using an internal VLAN ID.
  - Before configuring extended-range VLANs, enter the **show vlan internal usage** privileged EXEC command to see which VLANs have been allocated as internal VLANs.

– If necessary, you can shut down the routed port assigned to the internal VLAN, which frees up the internal VLAN, and then create the extended-range VLAN and re-enable the port, which then uses another VLAN as its internal VLAN. See the "Creating an Extended-Range VLAN with an Internal VLAN ID" section on page 11-14.

Creating an Extended-Range VLAN

You create an extended-range VLAN in global configuration mode by entering the **vlan** global configuration command with a VLAN ID from 1006 to 4094. This command accesses the config-vlan mode. The extended-range VLAN has the default Ethernet VLAN characteristics (see Table 11-2), and the MTU size is the only parameter you can change. See the description of the **vlan** global configuration command in the command reference for defaults of all parameters. If you enter an extended-range VLAN ID when the switch is not in VTP transparent mode, an error message is generated when you exit from config-vlan mode, and the extended-range VLAN is not created.

Extended-range VLANs are not saved in the VLAN database; they are saved in the switch running configuration file. You can save the extended-range VLAN configuration in the switch startup configuration file by using the **copy running-config startup-config** privileged EXEC command.

Note Before you create an extended-range VLAN, you can verify that the VLAN ID is not used internally by entering the **show vlan internal usage** privileged EXEC command. If the VLAN ID is used internally and you want to free it up, go to the"Creating an Extended-Range VLAN with an Internal VLAN ID" section on page 11-14 before creating the extended-range VLAN.

Beginning in privileged EXEC mode, follow these steps to create an extended-range VLAN:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vtp mode transparent**	Configure the switch for VTP transparent mode, disabling VTP.
Step 3	**vlan** *vlan-id*	Enter an extended-range VLAN ID and enter config-vlan mode. The range is 1006 to 4094.
Step 4	**mtu** *mtu-size*	(Optional) Modify the VLAN by changing the MTU size. **Note** Although all commands appear in the CLI help in config-vlan mode, only the **mtu** *mtu-size* command is supported for extended-range VLANs.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show vlan id** *vlan-id*	Verify that the VLAN has been created.
Step 7	**copy running-config startup config**	Save your entries in the switch startup configuration file. To save extended-range VLAN configurations, you need to save the VTP transparent mode configuration and the extended-range VLAN configuration in the switch startup configuration file. Otherwise, if the switch resets, it will default to VTP server mode, and the extended-range VLAN IDs will not be saved.

To delete an extended-range VLAN, use the **no vlan** *vlan-id* global configuration command.

The procedure for assigning static-access ports to an extended-range VLAN is the same as for normal-range VLANs. See the "Assigning Static-Access Ports to a VLAN" section on page 11-10.

This example shows how to create a new extended-range VLAN with all default characteristics, enter config-vlan mode, and save the new VLAN in the switch startup configuration file:

```
Switch(config)# vtp mode transparent
Switch(config)# vlan 2000
Switch(config-vlan)# end
Switch# copy running-config startup config
```

Creating an Extended-Range VLAN with an Internal VLAN ID

If you enter an extended-range VLAN ID that is already assigned to an internal VLAN, an error message is generated, and the extended-range VLAN is rejected. To manually free an internal VLAN ID, you must temporarily shut down the routed port that is using the internal VLAN ID.

Beginning in privileged EXEC mode, follow these steps to release a VLAN ID that is assigned to an internal VLAN and to create an extended-range VLAN with that ID:

	Command	Purpose
Step 1	**show vlan internal usage**	Display the VLAN IDs being used internally by the switch. If the VLAN ID that you want to use is an internal VLAN, the display shows the routed port that is using the VLAN ID. Enter that port number in Step 3.
Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**interface** *interface-id*	Enter the interface ID for the routed port that is using the VLAN ID.
Step 4	**shutdown**	Shut down the port to free the internal VLAN ID.
Step 5	**exit**	Return to global configuration mode.
Step 6	**vtp mode transparent**	Set the VTP mode to transparent for creating extended-range VLANs.
Step 7	**vlan** *vlan-id*	Enter the new extended-range VLAN ID, and enter config-vlan mode.
Step 8	**exit**	Exit from config-vlan mode, and return to global configuration mode.
Step 9	**interface** *interface-id*	Enter the interface ID for the routed port that you shut down in Step 4.
Step 10	**no shutdown**	Re-enable the routed port. It will be assigned a new internal VLAN ID.
Step 11	**end**	Return to privileged EXEC mode.
Step 12	**copy running-config startup config**	Save your entries in the switch startup configuration file. To save an extended-range VLAN configuration, you need to save the VTP transparent mode configuration and the extended-range VLAN configuration in the switch startup configuration file. Otherwise, if the switch resets, it will default to VTP server mode, and the extended-range VLAN IDs will not be saved.

Displaying VLANs

Use the **show vlan** privileged EXEC command to display a list of all VLANs on the switch, including extended-range VLANs. The display includes VLAN status, ports, and configuration information. To view normal-range VLANs in the VLAN database (1 to 1005) use the **show** VLAN configuration command (accessed by entering the **vlan database** privileged EXEC command). For a list of the VLAN IDs on the switch, use the **show running-config vlan** privileged EXEC command, optionally entering a VLAN ID range.

Table 11-3 lists the commands for monitoring VLANs.

Table 11-3 VLAN Monitoring Commands

Command	Command Mode	Purpose
show	VLAN configuration	Display status of VLANs in the VLAN database.
show current [*vlan-id*]	VLAN configuration	Display status of all or the specified VLAN in the VLAN database.
show interfaces [**vlan** *vlan-id*]	Privileged EXEC	Display characteristics for all interfaces or for the specified VLAN configured on the switch.
show running-config vlan	Privileged EXEC	Display all or a range of VLANs on the switch.
show vlan [**id** *vlan-id*]	Privileged EXEC	Display parameters for all VLANs or the specified VLAN on the switch.

For more details about the show command options and explanations of output fields, see the command reference for this release.

Configuring VLAN Trunks

These sections describe how VLAN trunks function on the switch:

- Trunking Overview, page 11-15
- Encapsulation Types, page 11-17
- Default Layer 2 Ethernet Interface VLAN Configuration, page 11-18

Trunking Overview

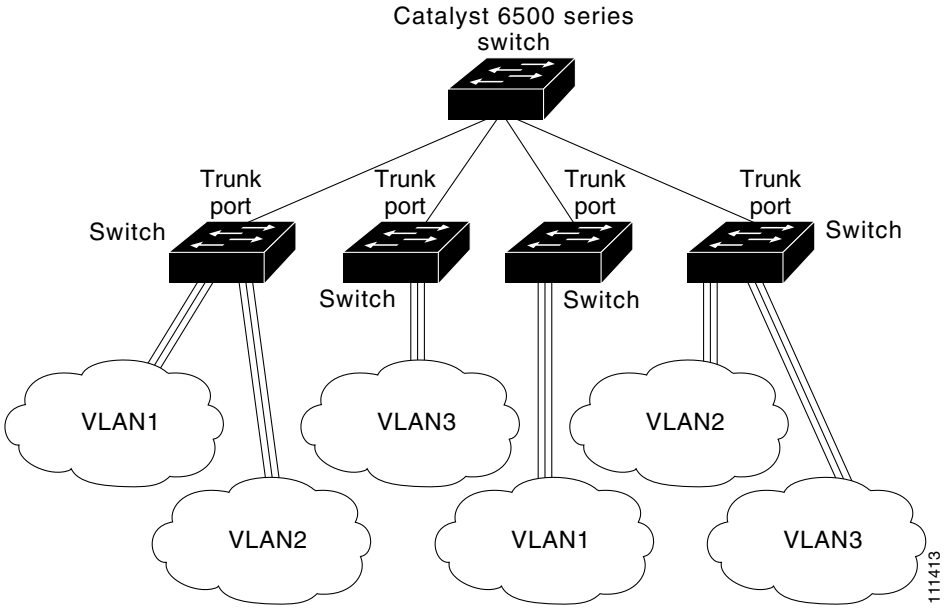
A trunk is a point-to-point link between one or more Ethernet switch interfaces and another networking device such as a router or a switch. Fast Ethernet and Gigabit Ethernet trunks carry the traffic of multiple VLANs over a single link, and you can extend the VLANs across an entire network.

Two trunking encapsulations are available on all Ethernet interfaces:

- Inter-Switch Link (ISL)—ISL is Cisco-proprietary trunking encapsulation.
- 802.1Q—802.1Q is industry-standard trunking encapsulation.

Figure 11-2 shows a network of switches that are connected by IEEE 802.1Q or ISL trunks.

Figure 11-2 Switches in an IEEE 802.1Q or ISL Trunking Environment



You can configure a trunk on a single Ethernet interface or on an EtherChannel bundle. For more information about EtherChannel, see Chapter 30, "Configuring EtherChannels."

Ethernet trunk interfaces support different trunking modes (see Table 11-4). You can set an interface as trunking or nontrunking or to negotiate trunking with the neighboring interface. To autonegotiate trunking, the interfaces must be in the same VTP domain.

Trunk negotiation is managed by the Dynamic Trunking Protocol (DTP), which is a Point-to-Point Protocol. However, some internetworking devices might forward DTP frames improperly, which could cause misconfigurations.

To avoid this, you should configure interfaces connected to devices that do not support DTP to not forward DTP frames, that is, to turn off DTP.

- If you do not intend to trunk across those links, use the **switchport mode access** interface configuration command to disable trunking.
- To enable trunking to a device that does not support DTP, use the **switchport mode trunk** and **switchport nonegotiate** interface configuration commands to cause the interface to become a trunk but to not generate DTP frames.
- With GigaStack GBICs, dynamic trunking is only supported when two switches are connected by a single GigaStack GBIC link. If trunking is required when more than two switches in a stack are connected by GigaStack GBIC links, you must manually configure trunking in this manner:
  - Manually shut down the GigaStack port by using the **shutdown** interface configuration command.
  - Manually configure trunk mode on the GigaStack port by using the **switchport mode trunk** interface configuration command on both GBIC interfaces to cause the interfaces to become trunks.
  - Use the **no shutdown** interface configuration command to bring up the GigaStack port.

You can also specify whether the trunk uses ISL or 802.1Q encapsulation or if the encapsulation type is autonegotiated. The DTP supports autonegotiation of both ISL and 802.1Q trunks.

Note Tunnel ports do not support DTP. See Chapter 14, "Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling," for more information on tunnel ports.

Table 11-4 Layer 2 Interface Modes

Mode	Function
switchport mode access	Puts the interface (access port) into permanent nontrunking mode. The interface becomes a nontrunk interface even if the neighboring interface is a trunk interface.
switchport mode dynamic desirable	Makes the interface actively attempt to convert the link to a trunk link. The interface becomes a trunk interface if the neighboring interface is set to *trunk*, *desirable*, or *auto* mode. The default switch-port mode for all Ethernet interfaces is **dynamic desirable**.
switchport mode dynamic auto	Makes the interface able to convert the link to a trunk link. The interface becomes a trunk interface if the neighboring interface is set to *trunk* or *desirable* mode.
switchport mode trunk	Puts the interface into permanent trunking mode and negotiates to convert the link into a trunk link. The interface becomes a trunk interface even if the neighboring interface is not a trunk interface.
switchport nonegotiate	Prevents the interface from generating DTP frames. You can use this command only when the interface switchport mode is **access** or **trunk**. You must manually configure the neighboring interface as a trunk interface to establish a trunk link.
switchport mode dot1q-tunnel	Configures the interface as a tunnel (nontrunking) port to be connected in an asymmetric link with an 802.1Q trunk port. 802.1Q tunneling is used to maintain customer VLAN integrity across a service provider network. See Chapter 14, "Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling," for more information on tunnel ports.

Encapsulation Types

Table 11-5 lists the Ethernet trunk encapsulation types and keywords.

Table 11-5 Ethernet Trunk Encapsulation Types

Encapsulation	Function
switchport trunk encapsulation isl	Specifies ISL encapsulation on the trunk link.
switchport trunk encapsulation dot1q	Specifies 802.1Q encapsulation on the trunk link.
switchport trunk encapsulation negotiate	Specifies that the interface negotiate with the neighboring interface to become an ISL (preferred) or 802.1Q trunk, depending on the configuration and capabilities of the neighboring interface.

Note The switch does not support Layer 3 trunks; you cannot configure subinterfaces or use the **encapsulation** keyword on Layer 3 interfaces. The switch does support Layer 2 trunks and Layer 3 VLAN interfaces, which provide equivalent capabilities.

The trunking mode, the trunk encapsulation type, and the hardware capabilities of the two connected interfaces determine whether a link becomes an ISL or 802.1Q trunk.

IEEE 802.1Q Configuration Considerations

IEEE 802.1Q trunks impose these limitations on a network:

- In a network of Cisco switches connected through IEEE 802.1Q trunks, the switches maintain one instance of spanning tree for each VLAN allowed on the trunks. Non-Cisco devices might support one spanning-tree instance for all VLANs.

  When you connect a Cisco switch to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco switch combines the spanning-tree instance of the VLAN of the trunk with the spanning-tree instance of the non-Cisco IEEE 802.1Q switch. However, spanning-tree information for each VLAN is maintained by Cisco switches separated by a cloud of non-Cisco IEEE 802.1Q switches. The non-Cisco IEEE 802.1Q cloud separating the Cisco switches is treated as a single trunk link between the switches.
- Make sure the native VLAN for an IEEE 802.1Q trunk is the same on both ends of the trunk link. If the native VLAN on one end of the trunk is different from the native VLAN on the other end, spanning-tree loops might result.
- Disabling spanning tree on the native VLAN of an IEEE 802.1Q trunk without disabling spanning tree on every VLAN in the network can potentially cause spanning-tree loops. We recommend that you leave spanning tree enabled on the native VLAN of an IEEE 802.1Q trunk or disable spanning tree on every VLAN in the network. Make sure your network is loop-free before disabling spanning tree.

Default Layer 2 Ethernet Interface VLAN Configuration

Table 11-6 shows the default Layer 2 Ethernet interface VLAN configuration.

Table 11-6 Default Layer 2 Ethernet Interface VLAN Configuration

Feature	Default Setting
Interface mode	**switchport mode dynamic desirable**
Trunk encapsulation	**switchport trunk encapsulation negotiate**
Allowed VLAN range	VLANs 1 to 4094.
VLAN range eligible for pruning	VLANs 2 to 1001
Default VLAN (for access ports)	VLAN 1
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1

Configuring an Ethernet Interface as a Trunk Port

Because trunk ports send and receive VTP advertisements, to use VTP you must ensure that at least one trunk port is configured on the switch and is connected to the trunk port of a second switch. Otherwise, the switch cannot receive any VTP advertisements.

This section includes these procedures for configuring an Ethernet interface as a trunk port on the switch:

- Interaction with Other Features, page 11-19
- Defining the Allowed VLANs on a Trunk, page 11-21
- Changing the Pruning-Eligible List, page 11-22
- Configuring the Native VLAN for Untagged Traffic, page 11-22

Note By default, an interface is in Layer 2 mode. The default mode for Layer 2 interfaces is **switchport mode dynamic desirable** interface configuration mode. If the neighboring interface supports trunking and is configured to allow trunking, the link is a Layer 2 trunk or, if the interface is in Layer 3 mode, it becomes a Layer 2 trunk when you enter the **switchport** interface configuration command. By default, trunks negotiate encapsulation. If the neighboring interface supports ISL and 802.1Q encapsulation and both interfaces are set to negotiate the encapsulation type, the trunk uses ISL encapsulation.

Interaction with Other Features

Trunking interacts with other features in these ways:

- A trunk port cannot be a tunnel port.
- Trunk ports can be grouped into EtherChannel port groups, but all trunks in the group must have the same configuration. When a group is first created, all ports follow the parameters set for the first port to be added to the group. If you change the configuration of one of these parameters, the switch propagates that setting to all ports in the group:
  - allowed-VLAN list
  - STP port priority for each VLAN
  - STP Port Fast setting
  - trunk status (If one port in a port group ceases to be a trunk, all ports cease to be trunks.)
- We recommend that you configure no more than 24 trunk ports in PVST+ mode and no more than 40 trunk ports in MST mode.
- If you try to enable IEEE 802.1X on a trunk port, an error message appears, and IEEE 802.1X is not enabled. If you try to change the mode of an IEEE 802.1X-enabled port to trunk, the port mode is not changed.
- A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable IEEE 802.1X on a dynamic port, an error message appears, and IEEE 802.1X is not enabled. If you try to change the mode of an IEEE 802.1X-enabled port to dynamic, the port mode is not changed.
- Protected ports are supported on IEEE 802.1Q trunks.

Configuring a Trunk Port

Beginning in privileged EXEC mode, follow these steps to configure a port as an ISL or IEEE 802.1Q trunk port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter the interface configuration mode and the port to be configured for trunking.
Step 3	**switchport trunk encapsulation** {**isl** | **dot1q** | **negotiate**}	Configure the port to support ISL or 802.1Q encapsulation or to negotiate (the default) with the neighboring interface for encapsulation type. You must configure each end of the link with the same encapsulation type.
Step 4	**switchport mode** {**dynamic** {**auto** | **desirable**} | **trunk**}	Configure the interface as a Layer 2 trunk (required only if the interface is a Layer 2 access port or tunnel port, or to specify the trunking mode). • **dynamic auto**—Set the interface to a trunk link if the neighboring interface is set to trunk or desirable mode. • **dynamic desirable**—Set the interface to a trunk link if the neighboring interface is set to trunk, desirable, or auto mode. • **trunk**—Set the interface in permanent trunking mode and negotiate to convert the link to a trunk link even if the neighboring interface is not a trunk interface.
Step 5	**switchport access vlan** *vlan-id*	(Optional) Specify the default VLAN, which is used if the interface stops trunking.
Step 6	**switchport trunk native vlan** *vlan-id*	Specify the native VLAN for 802.1Q trunks.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show interfaces** *interface-id* **switchport**	Display the switchport configuration of the interface in the *Administrative Mode* and the *Administrative Trunking Encapsulation* fields of the display.
Step 9	**show interfaces** *interface-id* **trunk**	Display the trunk configuration of the interface.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return an interface to its default configuration, use the **default interface** *interface-id* interface configuration command. To reset all trunking characteristics of a trunking interface to the defaults, use the **no switchport trunk** interface configuration command. To disable trunking, use the **switchport mode access** interface configuration command to configure the port as a static-access port.

This example shows how to configure a port as an IEEE 802.1Q trunk. The example assumes that the neighbor interface is configured to support IEEE 802.1Q trunking.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface fastethernet0/4
Switch(config-if)# switchport mode dynamic desirable
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# end
```

Defining the Allowed VLANs on a Trunk

By default, a trunk port sends traffic to and receives traffic from all VLANs. All VLAN IDs are allowed on each trunk. However, you can remove VLANs from the allowed list, preventing traffic from those VLANs from passing over the trunk. To restrict the traffic a trunk carries, use the **switchport trunk allowed vlan remove** *vlan-list* interface configuration command to remove specific VLANs from the allowed list.

To reduce the risk of spanning-tree loops or storms, you can disable VLAN 1 on any individual VLAN trunk port by removing VLAN 1 from the allowed list. This is known as VLAN 1 minimization. VLAN 1 minimization disables VLAN 1 (the default VLAN on all Cisco switch trunk ports) on an individual VLAN trunk link. As a result, no user traffic, including spanning-tree advertisements, is sent or received on VLAN 1.

When you remove VLAN 1 from a trunk port, the interface continues to send and receive management traffic, for example, Cisco Discovery Protocol (CDP), Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), Dynamic Trunking Protocol (DTP), and VLAN Trunking Protocol (VTP) in VLAN 1.

If a trunk port with VLAN 1 disabled is converted to a nontrunk port, it is added to the access VLAN. If the access VLAN is set to 1, the port is added to VLAN 1, regardless of the **switchport trunk allowed** setting. The same is true for any VLAN that has been disabled on the port.

A trunk port can become a member of a VLAN if the VLAN is enabled, if VTP knows of the VLAN, and if the VLAN is in the allowed list for the port. When VTP detects a newly enabled VLAN and the VLAN is in the allowed list for a trunk port, the trunk port automatically becomes a member of the enabled VLAN. When VTP detects a new VLAN and the VLAN is not in the allowed list for a trunk port, the trunk port does not become a member of the new VLAN.

Beginning in privileged EXEC mode, follow these steps to modify the allowed list of an ISL or IEEE 802.1Q trunk:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode and the port to be configured.
Step 3	**switchport mode trunk**	Configure the interface as a VLAN trunk port.
Step 4	**switchport trunk allowed vlan** {**add** | **all** | **except** | **remove**} *vlan-list*	(Optional) Configure the list of VLANs allowed on the trunk. For explanations about using the **add**, **all**, **except**, and **remove** keywords, see the command reference for this release. The *vlan-list* parameter is either a single VLAN number from 1 to 4094 or a range of VLANs described by two VLAN numbers, the lower one first, separated by a hyphen. Do not enter any spaces between comma-separated VLAN parameters or in hyphen-specified ranges. All VLANs are allowed by default.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport**	Verify your entries in the *Trunking VLANs Enabled* field of the display.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default allowed VLAN list of all VLANs, use the **no switchport trunk allowed vlan** interface configuration command.

This example shows how to remove VLAN 2 from the allowed VLAN list:

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# switchport trunk allowed vlan remove 2
Switch(config-if)# end
Switch#
```

Changing the Pruning-Eligible List

The pruning-eligible list applies only to trunk ports. Each trunk port has its own eligibility list. VTP pruning must be enabled for this procedure to take effect. The "Enabling VTP Pruning" section on page 12-13 describes how to enable VTP pruning.

Beginning in privileged EXEC mode, follow these steps to remove VLANs from the pruning-eligible list on a trunk port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and select the trunk port for which VLANs should be pruned.
Step 3	**switchport trunk pruning vlan** {**add** | **except** | **none** | **remove**} *vlan-list* [,*vlan*[,*vlan*[,,,]]	Configure the list of VLANs allowed to be pruned from the trunk. (See the "VTP Pruning" section on page 12-4). For explanations about using the **add**, **except**, **none**, and **remove** keywords, see the command reference for this release. Separate nonconsecutive VLAN IDs with a comma and no spaces; use a hyphen to designate a range of IDs. Valid IDs are from 2 to 1001**. Extended-range VLANs (VLAN IDs 1006 to 4094) cannot be pruned.** VLANs that are pruning-ineligible receive flooded traffic. The default list of VLANs allowed to be pruned contains VLANs 2 to 1001.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **switchport**	Verify your entries in the *Pruning VLANs Enabled* field of the display.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default pruning-eligible list of all VLANs, use the **no switchport trunk pruning vlan** interface configuration command.

Configuring the Native VLAN for Untagged Traffic

A trunk port configured with IEEE 802.1Q tagging can receive both tagged and untagged traffic. By default, the switch forwards untagged traffic in the native VLAN configured for the port. The native VLAN is VLAN 1 by default.

Note The native VLAN can be assigned any VLAN ID.

For information about IEEE 802.1Q configuration issues, see the "Encapsulation Types" section on page 11-17.

Beginning in privileged EXEC mode, follow these steps to configure the native VLAN on an IEEE 802.1Q trunk:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and define the interface that is configured as the IEEE 802.1Q trunk.
Step 3	**switchport trunk native vlan** *vlan-id*	Configure the VLAN that is sending and receiving untagged traffic on the trunk port. For *vlan-id*, the range is 1 to 4094.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **switchport**	Verify your entries in the *Trunking Native Mode VLAN* field.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default native VLAN, VLAN 1, use the **no switchport trunk native vlan** interface configuration command.

If a packet has a VLAN ID that is the same as the outgoing port native VLAN ID, the packet is sent untagged; otherwise, the switch sends the packet with a tag.

Load Sharing Using STP

Load sharing divides the bandwidth supplied by parallel trunks connecting switches. To avoid loops, STP normally blocks all but one parallel link between switches. Using load sharing, you divide the traffic between the links according to which VLAN the traffic belongs.

You configure load sharing on trunk ports by using STP port priorities or STP path costs. For load sharing using STP port priorities, both load-sharing links must be connected to the same switch. For load sharing using STP path costs, each load-sharing link can be connected to the same switch or to two different switches. For more information about STP, see Chapter 15, “Configuring STP.”

Load Sharing Using STP Port Priorities

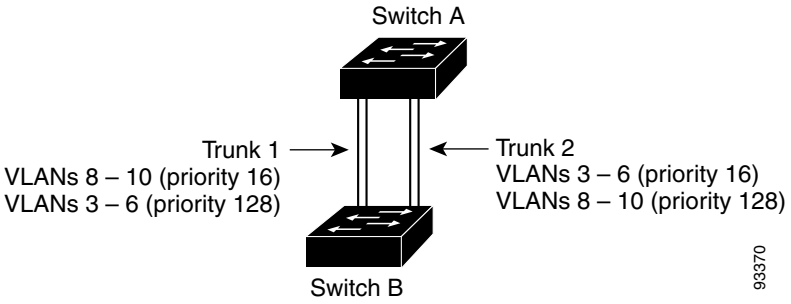
When two ports on the same switch form a loop, the STP port priority setting determines which port is enabled and which port is in a blocking state. You can set the priorities on a parallel trunk port so that the port carries all the traffic for a given VLAN. The trunk port with the higher priority (lower values) for a VLAN is forwarding traffic for that VLAN. The trunk port with the lower priority (higher values) for the same VLAN remains in a blocking state for that VLAN. One trunk port sends or receives all traffic for the VLAN.

Figure 11-3 shows two trunks connecting supported switches. In this example, the switches are configured as follows:

- VLANs 8 through 10 are assigned a port priority of 16 on Trunk 1.
- VLANs 3 through 6 retain the default port priority of 128 on Trunk 1.
- VLANs 3 through 6 are assigned a port priority of 16 on Trunk 2.
- VLANs 8 through 10 retain the default port priority of 128 on Trunk 2.

In this way, Trunk 1 carries traffic for VLANs 8 through 10, and Trunk 2 carries traffic for VLANs 3 through 6. If the active trunk fails, the trunk with the lower priority takes over and carries the traffic for all of the VLANs. No duplication of traffic occurs over any trunk port.

Figure 11-3 Load Sharing by Using STP Port Priorities



Beginning in privileged EXEC mode, follow these steps to configure the network shown in Figure 11-3.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode on Switch 1.
Step 2	**vtp domain** *domain-name*	Configure a VTP administrative domain. The domain name can be from 1 to 32 characters.
Step 3	**vtp mode server**	Configure Switch 1 as the VTP server.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show vtp status**	Verify the VTP configuration on both Switch A and Switch B. In the display, check the *VTP Operating Mode* and the *VTP Domain Name* fields.
Step 6	**show vlan**	Verify that the VLANs exist in the database on Switch A.
Step 7	**configure terminal**	Enter global configuration mode.
Step 8	**interface fastethernet 0/1**	Enter interface configuration mode, and define Fast Ethernet port 0/1 as the interface to be configured as a trunk.
Step 9	**switchport trunk encapsulation** {**isl** | **dot1q** | **negotiate**}	Configure the port to support ISL or 802.1Q encapsulation or to negotiate with the neighboring interface. You must configure each end of the link with the same encapsulation type.
Step 10	**switchport mode trunk**	Configure the port as a trunk port.
Step 11	**end**	Return to privilege EXEC mode.
Step 12	**show interfaces fastethernet 0/1switchport**	Verify the VLAN configuration.
Step 13		Repeat Steps 7 through 11 on Switch A for Fast Ethernet port 0/2.
Step 14		Repeat Steps 7 through 11 on Switch B to configure the trunk ports on Fast Ethernet ports 0/1 and 0/2.

	Command	Purpose
Step 15	**show vlan**	When the trunk links come up, VTP passes the VTP and VLAN information to Switch B. Verify that Switch B has learned the VLAN configuration.
Step 16	**configure terminal**	Enter global configuration mode on Switch A.
Step 17	**interface fastethernet 0/1**	Enter interface configuration mode, and define the interface to set the STP port priority.
Step 18	**spanning-tree vlan 8-10 port-priority 16**	Assign the port priority of 16 for VLANs 8 through 10.
Step 19	**spanning-tree vlan 10 port-priority 16**	Assign the port priority of 16 for VLAN 10.
Step 20	**exit**	Return to global configuration mode.
Step 21	**interface fastethernet 0/2**	Enter interface configuration mode, and define the interface to set the STP port priority.
Step 22	**spanning-tree vlan 3-6 port-priority 16**	Assign the port priority of 16 for VLANs 3 through 6.
Step 23	**end**	Return to privileged EXEC mode.
Step 24	**show running-config**	Verify your entries.
Step 25	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

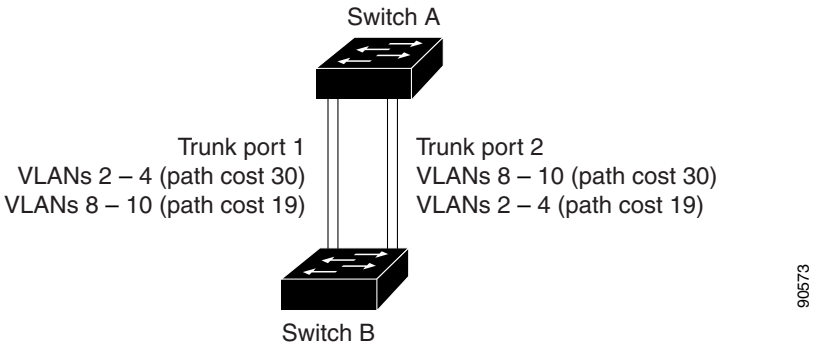
Load Sharing Using STP Path Cost

You can configure parallel trunks to share VLAN traffic by setting different path costs on a trunk and associating the path costs with different sets of VLANs. The VLANs keep the traffic separate. Because no loops exist, STP does not disable the ports, and redundancy is maintained in the event of a lost link.

In Figure 11-4, Trunk ports 1 and 2 are 100BASE-T ports. The path costs for the VLANs are assigned as follows:

- VLANs 2 through 4 are assigned a path cost of 30 on Trunk port 1.
- VLANs 8 through 10 retain the default 100BASE-T path cost on Trunk port 1 of 19.
- VLANs 8 through 10 are assigned a path cost of 30 on Trunk port 2.
- VLANs 2 through 4 retain the default 100BASE-T path cost on Trunk port 2 of 19.

Figure 11-4 Load-Sharing Trunks with Traffic Distributed by Path Cost



Beginning in privileged EXEC mode, follow these steps to configure the network shown in Figure 11-4:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode on Switch A.
Step 2	**interface fastethernet 0/1**	Enter interface configuration mode, and define Fast Ethernet port 0/1as the interface to be configured as a trunk.
Step 3	**switchport trunk encapsulation {isl | dot1q | negotiate}**	Configure the port to support ISL or 802.1Q encapsulation. You must configure each end of the link with the same encapsulation type.
Step 4	**switchport mode trunk**	Configure the port as a trunk port. The trunk defaults to ISL trunking.
Step 5	**exit**	Return to global configuration mode.
Step 6		Repeat Steps 2 through 4 on Switch A interface Fast Ethernet 0/2.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show running-config**	Verify your entries. In the display, make sure that interfaces Fast Ethernet 0/1 and Fast Ethernet 0/2 are configured as trunk ports.
Step 9	**show vlan**	When the trunk links come up, Switch A receives the VTP information from the other switches. Verify that Switch A has learned the VLAN configuration.
Step 10	**configure terminal**	Enter global configuration mode.
Step 11	**interface fastethernet 0/1**	Enter interface configuration mode, and define Fast Ethernet port 0/1 as the interface to set the STP cost.
Step 12	**spanning-tree vlan 2-4 cost 30**	Set the spanning-tree path cost to 30 for VLANs 2 through 4.
Step 13	**end**	Return to global configuration mode.
Step 14		Repeat Steps 9 through 11 on Switch A interface Fast Ethernet 0/2, and set the spanning-tree path cost to 30 for VLANs 8, 9, and 10.
Step 15	**exit**	Return to privileged EXEC mode.
Step 16	**show running-config**	Verify your entries. In the display, verify that the path costs are set correctly for interfaces Fast Ethernet 0/1 and 0/2.
Step 17	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring VMPS

The switch cannot be a VMPS server but can act as a client to the VMPS and communicate with it through the VLAN Query Protocol (VQP). VMPS dynamically assigns dynamic access port VLAN membership.

This section includes this information about configuring VMPS:

- "Understanding VMPS" section on page 11-27
- "Default VMPS Client Configuration" section on page 11-28

- "VMPS Configuration Guidelines" section on page 11-28
- "Configuring the VMPS Client" section on page 11-29
- "Monitoring the VMPS" section on page 11-32
- "Troubleshooting Dynamic Port VLAN Membership" section on page 11-32
- "VMPS Configuration Example" section on page 11-33

Understanding VMPS

When the VMPS receives a VQP request from a client switch, it searches its database for a MAC-address-to-VLAN mapping. The server response is based on this mapping and whether or not the server is in secure mode. Secure mode determines whether the server shuts down the port when a VLAN is not allowed on it or just denies the port access to the VLAN.

In response to a request, the VMPS takes one of these actions:

- If the assigned VLAN is restricted to a group of ports, the VMPS verifies the requesting port against this group and responds as follows:
  - If the VLAN is allowed on the port, the VMPS sends the VLAN name to the client in response.
  - If the VLAN is not allowed on the port and the VMPS is not in secure mode, the VMPS sends an *access-denied* response.
  - If the VLAN is not allowed on the port and the VMPS is in secure mode, the VMPS sends a *port-shutdown* response.
- If the VLAN in the database does not match the current VLAN on the port and active hosts exist on the port, the VMPS sends an *access-denied* or a *port-shutdown* response, depending on the secure mode of the VMPS.

If the switch receives an *access-denied* response from the VMPS, it continues to block traffic from the MAC address to or from the port. The switch continues to monitor the packets directed to the port and sends a query to the VMPS when it identifies a new address. If the switch receives a *port-shutdown* response from the VMPS, it disables the port. The port must be manually re-enabled by using the device manager, CLI, Network Assistant, or SNMP.

You can also use an explicit entry in the configuration table to deny access to specific MAC addresses for security reasons. If you enter the **none** keyword for the VLAN name, the VMPS sends an *access-denied* or *port-shutdown* response, depending on the VMPS secure mode setting.

Dynamic Port VLAN Membership

A dynamic (nontrunking) port on the switch can belong to only one VLAN, with a VLAN ID from 1 to 1005. When the link comes up, the switch does not forward traffic to or from this port until the VMPS provides the VLAN assignment. The VMPS receives the source MAC address from the first packet of a new host connected to the dynamic port and attempts to match the MAC address to a VLAN in the VMPS database.

If there is a match, the VMPS sends the VLAN number for that port. If the client switch was not previously configured, it uses the domain name from the first VTP packet it receives on its trunk port from the VMPS. If the client switch was previously configured, it includes its domain name in the query packet to the VMPS to obtain its VLAN number. The VMPS verifies that the domain name in the packet matches its own domain name before accepting the request and responds to the client with the assigned VLAN number for the client. If there is no match, the VMPS either denies the request or shuts down the port (depending on the VMPS secure mode setting).

Multiple hosts (MAC addresses) can be active on a dynamic port if they are all in the same VLAN; however, the VMPS shuts down a dynamic port if more than 20 hosts are active on the port.

If the link goes down on a dynamic port, the port returns to an isolated state and does not belong to a VLAN. Any hosts that come online through the port are checked again through the VQP with the VMPS before the port is assigned to a VLAN.

VMPS Database Configuration File

The VMPS contains a database configuration file that you create. This ASCII text file is stored on a switch-accessible TFTP server that functions as a server for VMPS. The file contains VMPS information, such as the domain name, the fallback VLAN name, and the MAC-address-to-VLAN mapping. The switch cannot act as the VMPS, but you can use a Catalyst 5000 or Catalyst 6000 series switch as the VMPS.

You can configure a fallback VLAN name. If you connect a device with a MAC address that is not in the database, the VMPS sends the fallback VLAN name to the client. If you do not configure a fallback VLAN and the MAC address does not exist in the database, the VMPS sends an *access-denied* response. If the VMPS is in secure mode, it sends a *port-shutdown* response.

Whenever port names are used in the VMPS database configuration file, the server must use the switch convention for naming ports. For example, Fa0/4Gi0/17 is fixed Fast Ethernet port number 4Gigabit Ethernet port number 17. If the switch is a cluster member, the command switch adds the name of the switch before the type. For example, *es3%Fa0/4es3%Gi0/17* refers to fixed Fast Ethernet port number 4 Gigabit Ethernet port number 17 on member switch 3. When port names are required, these naming conventions must be followed in the VMPS database configuration file when it is configured to support a cluster.

Default VMPS Client Configuration

Table 11-7 shows the default VMPS and dynamic port configuration on client switches.

Table 11-7 Default VMPS Client and Dynamic Port Configuration

Feature	Default Setting
VMPS domain server	None
VMPS reconfirm interval	60 minutes
VMPS server retry count	3
Dynamic ports	None configured

VMPS Configuration Guidelines

These guidelines and restrictions apply to dynamic access port VLAN membership:

- You should configure the VMPS before you configure ports as dynamic.
- The communication between a cluster of switches and VMPS is managed by the command switch and includes port-naming conventions that are different from standard port names. For the cluster-based port-naming conventions, see the "VMPS Database Configuration File" section on page 11-28.

- When you configure a port as a dynamic access port, the spanning-tree Port Fast feature is automatically enabled for that port. The Port Fast mode accelerates the process of bringing the port into the forwarding state.
- IEEE 802.1X ports cannot be configured as dynamic access ports. If you try to enable IEEE 802.1X on a dynamic-access (VQP) port, an error message appears, and IEEE 802.1X is not enabled. If you try to change an IEEE 802.1X-enabled port to dynamic VLAN assignment, an error message appears, and the VLAN configuration is not changed.
- Trunk ports cannot be dynamic access ports, but you can enter the **switchport access vlan dynamic** interface configuration command for a trunk port. In this case, the switch retains the setting and applies it if the port is later configured as an access port.

  You must turn off trunking on the port before the dynamic access setting takes effect.
- Dynamic access ports cannot be monitor ports.
- Secure ports cannot be dynamic access ports. You must disable port security on a port before it becomes dynamic.
- Dynamic access ports cannot be members of an EtherChannel group.
- Port channels cannot be configured as dynamic access ports.
- A dynamic access port can participate in fallback bridging.
- The VTP management domain of the VMPS client and the VMPS server must be the same.
- VQP does not support extended-range VLANs (VLAN IDs higher than 1006). Extended-range VLANs cannot be configured by VMPS.
- The VLAN configured on the VMPS server should not be a voice VLAN.

Configuring the VMPS Client

You configure dynamic VLANs by using the VMPS (server). The switch can be a VMPS client; it cannot be a VMPS server.

Entering the IP Address of the VMPS

You must first enter the IP address of the server to configure the switch as a client.

Note If the VMPS is being defined for a cluster of switches, enter the address on the command switch.

Beginning in privileged EXEC mode, follow these steps to enter the IP address of the VMPS:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vmps server** *ipaddress* **primary**	Enter the IP address of the switch acting as the primary VMPS server.
Step 3	**vmps server** *ipaddress*	Enter the IP address of the switch acting as a secondary VMPS server. You can enter up to three secondary server addresses.
Step 4	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 5	**show vmps**	Verify your entries in the *VMPS Domain Server* field of the display.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note The switch port that is connected to the VMPS server cannot be a dynamic access port. It can be either a static access port or a trunk port. See the "Configuring an Ethernet Interface as a Trunk Port" section on page 11-18.

Configuring Dynamic Access Ports on VMPS Clients

If you are configuring a port on a cluster member switch as a dynamic port, first use the **rcommand** privileged EXEC command to log into the member switch.

Caution Dynamic port VLAN membership is for end stations or hubs connected to end stations. Connecting dynamic access ports to other switches can cause a loss of connectivity.

Beginning in privileged EXEC mode, follow these steps to configure a dynamic access port on a VMPS client switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode and the switch port that is connected to the end station.
Step 3	**switchport mode access**	Set the port to access mode.
Step 4	**switchport access vlan dynamic**	Configure the port as eligible for dynamic VLAN membership. The dynamic access port must be connected to an end station.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport**	Verify your entries in the *Operational Mode* field of the display.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return an interface to its default configuration, use the **default interface** *interface-id* interface configuration command. To return an interface to its default switchport mode (dynamic desirable), use the **no switchport mode** interface configuration command. To reset the access mode to the default VLAN for the switch, use the **no switchport access** interface configuration command.

Note When you configure a dynamic access port by using the **switchport access vlan dynamic** interface configuration command, the port might allow unauthorized users to access network resources if the interface changes from access mode to trunk mode through the DTP negotiation. The workaround is to configure the port as a static access port.

Reconfirming VLAN Memberships

Beginning in privileged EXEC mode, follow these steps to confirm the dynamic port VLAN membership assignments that the switch has received from the VMPS:

	Command	Purpose
Step 1	**vmps reconfirm**	Reconfirm dynamic port VLAN membership.
Step 2	**show vmps**	Verify the dynamic VLAN reconfirmation status.

Changing the Reconfirmation Interval

VMPS clients periodically reconfirm the VLAN membership information received from the VMPS. You can set the number of minutes after which reconfirmation occurs.

If you are configuring a member switch in a cluster, this parameter must be equal to or greater than the reconfirmation setting on the command switch. You must also first use the **rcommand** privileged EXEC command to log into the member switch.

Beginning in privileged EXEC mode, follow these steps to change the reconfirmation interval:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vmps reconfirm** *minutes*	Enter the number of minutes between reconfirmations of the dynamic VLAN membership. Enter a number from 1 to 120. The default is 60 minutes.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show vmps**	Verify the dynamic VLAN reconfirmation status in the *Reconfirm Interval* field of the display.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no vmps reconfirm** global configuration command.

Changing the Retry Count

Beginning in privileged EXEC mode, follow these steps to change the number of times that the switch attempts to contact the VMPS before querying the next server:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vmps retry** *count*	Change the retry count. The retry range is from 1 to 10; the default is 3.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show vmps**	Verify your entry in the *Server Retry Count* field of the display.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no vmps retry** global configuration command.

Monitoring the VMPS

You can display information about the VMPS by using the **show vmps** privileged EXEC command. The switch displays this information about the VMPS:

VMPS VQP Version	The version of VQP used to communicate with the VMPS. The switch queries the VMPS that is using VQP version 1.
Reconfirm Interval	The number of minutes the switch waits before reconfirming the VLAN-to-MAC-address assignments.
Server Retry Count	The number of times VQP resends a query to the VMPS. If no response is received after this many tries, the switch starts to query the secondary VMPS.
VMPS domain server	The IP address of the configured VLAN membership policy servers. The switch sends queries to the one marked *current*. The one marked *primary* is the primary server.
VMPS Action	The result of the most recent reconfirmation attempt. A reconfirmation attempt can occur automatically when the reconfirmation interval expired, or you can force it by entering the **vmps reconfirm** privileged EXEC command or its Network Assistant or SNMP equivalent.

This is an example of output for the **show vmps** privileged EXEC command:

```
Switch# show vmps

VQP Client Status:
--------------------
VMPS VQP Version:   1
Reconfirm Interval: 60 min
Server Retry Count: 3
VMPS domain server: 172.20.128.86 (primary, current)
                    172.20.128.87

Reconfirmation status
---------------------
VMPS Action:         No Dynamic Port
```

Troubleshooting Dynamic Port VLAN Membership

The VMPS shuts down a dynamic port under these conditions:

- The VMPS is in secure mode, and it does not allow the host to connect to the port. The VMPS shuts down the port to prevent the host from connecting to the network.
- More than 20 active hosts reside on a dynamic port.

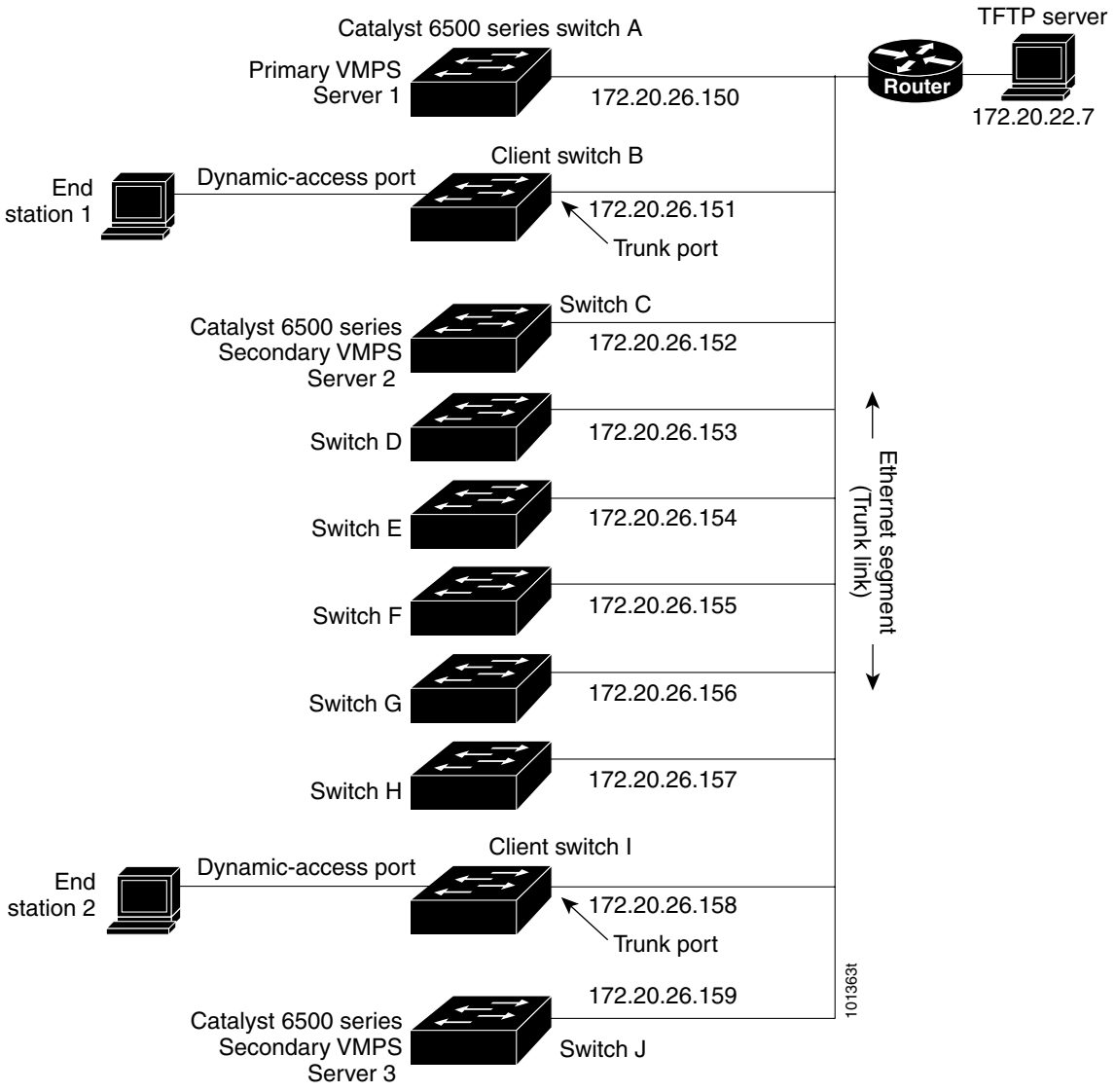
To re-enable a disabled dynamic port, enter the **no shutdown** interface configuration command.

VMPS Configuration Example

Figure 11-5 shows a network with a VMPS server switch and VMPS client switches with dynamic ports. In this example, these assumptions apply:

- The VMPS server and the VMPS client are separate switches.
- The Catalyst 6500 series Switch A is the primary VMPS server.
- The Catalyst 5000 series Switch C and Switch J are secondary VMPS servers.
- End stations are connected to the clients, Switch B and Switch I.
- The database configuration file is stored on the TFTP server with the IP address 172.20.22.7.

Figure 11-5 Dynamic Port VLAN Membership Configuration



CHAPTER 12

Configuring VTP

This chapter describes how to use the VLAN Trunking Protocol (VTP) and the VLAN database for managing VLANs on your Catalyst 3550switch.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

The chapter includes these sections:

- Understanding VTP, page 12-1
- Configuring VTP, page 12-6
- Monitoring VTP, page 12-15

Understanding VTP

VTP is a Layer 2 messaging protocol that maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs on a network-wide basis. VTP minimizes misconfigurations and configuration inconsistencies that can cause several problems, such as duplicate VLAN names, incorrect VLAN-type specifications, and security violations.

Before you create VLANs, you must decide whether to use VTP in your network. Using VTP, you can make configuration changes centrally on one or more switches and have those changes automatically communicated to all the other switches in the network. Without VTP, you cannot send information about VLANs to other switches. VTP configuration information is saved in the VTP VLAN database.

VTP only learns about normal-range VLANs (VLAN IDs 1 to 1005). Extended-range VLANs (VLAN IDs greater than 1005) are not supported by VTP or stored in the VTP VLAN database.

This section contains information about these VTP parameters:

- The VTP Domain, page 12-2
- VTP Modes, page 12-2
- VTP Advertisements, page 12-3
- VTP Version 2, page 12-4
- VTP Pruning, page 12-4

The VTP Domain

A VTP domain (also called a VLAN management domain) consists of one switch or several interconnected switches under the same administrative responsibility sharing the same VTP domain name. A switch can be in only one VTP domain.You can make global VLAN configuration changes for the domain.

By default, the switch is in VTP no-management-domain state until it receives an advertisement for a domain over a trunk link (a link that carries the traffic of multiple VLANs) or until you configure a domain name. Until the management domain name is specified or learned, you cannot create or modify VLANs on a VTP server, and VLAN information is not propagated over the network.

If the switch receives a VTP advertisement over a trunk link, it inherits the management domain name and the VTP configuration revision number. The switch then ignores advertisements with a different domain name or an earlier configuration revision number.

Caution Before adding a VTP client switch to a VTP domain, always verify that its VTP configuration revision number is ***lower*** than the configuration revision number of the other switches in the VTP domain. Switches in a VTP domain always use the VLAN configuration of the switch with the highest VTP configuration revision number. If you add a switch that has a revision number higher than the revision number in the VTP domain, it can erase all VLAN information from the VTP server and VTP domain. See the "Adding a VTP Client Switch to a VTP Domain" section on page 12-14 for the procedure for verifying and resetting the VTP configuration revision number.

When you make a change to the VLAN configuration on a VTP server, the change is propagated to all switches in the VTP domain. VTP advertisements are sent over all trunk connections, including Inter-Switch Link (ISL) and IEEE 802.1Q. VTP maps VLANs dynamically across multiple LAN types with unique names and internal index associates. Mapping eliminates excessive device administration required from network administrators.

If you configure a switch for VTP transparent mode, you can create and modify VLANs, but the changes are not sent to other switches in the domain, and they affect only the individual switch. However, configuration changes made when the switch is in this mode are saved in the switch running configuration and can be saved to the switch startup configuration file.

For domain name and password configuration guidelines, see the "VTP Configuration Guidelines" section on page 12-8.

VTP Modes

You can configure a supported switch to be in one of the VTP modes listed in Table 12-1.

Table 12-1 VTP Modes

VTP Mode	Description
VTP server	In VTP server mode, you can create, modify, and delete VLANs and specify other configuration parameters (such as the VTP version) for the entire VTP domain. VTP servers advertise their VLAN configurations to other switches in the same VTP domain and synchronize their VLAN configurations with other switches based on advertisements received over trunk links. In VTP server mode, VLAN configurations are saved in NVRAM. VTP server is the default mode.
VTP client	A VTP client behaves like a VTP server, but you cannot create, change, or delete VLANs on a VTP client. In VTP client mode, VLAN configurations are not saved in NVRAM.
VTP transparent	VTP transparent switches do not participate in VTP. A VTP transparent switch does not advertise its VLAN configuration and does not synchronize its VLAN configuration based on received advertisements. However, in VTP version 2, transparent switches do forward VTP advertisements that they receive from other switches from their trunk interfaces. You can create, modify, and delete VLANs on a switch in VTP transparent mode. The switch must be in VTP transparent mode when you create extended-range VLANs. See the "Configuring Extended-Range VLANs" section on page 11-11. When the switch is in VTP transparent mode, the VTP and VLAN configurations are saved in NVRAM, but they are not advertised to other switches. In this mode, VTP mode and domain name are saved in the switch running configuration and you can save this information in the switch startup configuration file by entering the **copy running-config startup-config** privileged EXEC command.

VTP Advertisements

Each switch in the VTP domain sends periodic global configuration advertisements from each trunk port to a reserved multicast address. Neighboring switches receive these advertisements and update their VTP and VLAN configurations as necessary.

Note Because trunk ports send and receive VTP advertisements, you must ensure that at least one trunk port is configured on the switch and that this trunk port is connected to the trunk port of a second switch. Otherwise, the switch cannot receive any VTP advertisements. For more information on trunk ports, see the "Configuring VLAN Trunks" section on page 11-15.

VTP advertisements distribute this global domain information:

- VTP domain name
- VTP configuration revision number
- Update identity and update timestamp
- MD5 digest VLAN configuration, including maximum transmission unit (MTU) size for each VLAN.
- Frame format

VTP advertisements distribute this VLAN information for each configured VLAN:

- VLAN IDs (ISL and 802.1Q)
- VLAN name
- VLAN type

- VLAN state
- Additional VLAN configuration information specific to the VLAN type

VTP Version 2

If you use VTP in your network, you must decide whether to use version 1 or version 2. By default, VTP operates in version 1.

VTP version 2 supports these features not supported in version 1:

- Token Ring support—VTP version 2 supports Token Ring Bridge Relay Function (TrBRF) and Token Ring Concentrator Relay Function (TrCRF) VLANs. For more information about Token Ring VLANs, see the "Configuring Normal-Range VLANs" section on page 11-4.
- Unrecognized Type-Length-Value (TLV) support—A VTP server or client propagates configuration changes to its other trunks, even for TLVs it is not able to parse. The unrecognized TLV is saved in NVRAM when the switch is operating in VTP server mode.
- Version-Dependent Transparent Mode—In VTP version 1, a VTP transparent switch inspects VTP messages for the domain name and version and forwards a message only if the version and domain name match. Because VTP version 2 supports only one domain, it forwards VTP messages in transparent mode without inspecting the version and domain name.
- Consistency Checks—In VTP version 2, VLAN consistency checks (such as VLAN names and values) are performed only when you enter new information through the CLI or SNMP. Consistency checks are not performed when new information is obtained from a VTP message or when information is read from NVRAM. If the MD5 digest on a received VTP message is correct, its information is accepted.

VTP Pruning

VTP pruning increases network available bandwidth by restricting flooded traffic to those trunk links that the traffic must use to reach the destination devices. Without VTP pruning, a switch floods broadcast, multicast, and unknown unicast traffic across all trunk links within a VTP domain even though receiving switches might discard them. VTP pruning is disabled by default.

VTP pruning blocks unneeded flooded traffic to VLANs on trunk ports that are included in the pruning-eligible list. Only VLANs included in the pruning-eligible list can be pruned. By default, VLANs 2 through 1001 are pruning eligible switch trunk ports. If the VLANs are configured as pruning-ineligible, the flooding continues. VTP pruning is supported with VTP version 1 and version 2.

Figure 12-1 shows a switched network without VTP pruning enabled. Port 1 on Switch A and Port 2 on Switch D are assigned to the Red VLAN. If a broadcast is sent from the host connected to Switch A, Switch A floods the broadcast and every switch in the network receives it, even though Switches C, E, and F have no ports in the Red VLAN.

Figure 12-1 Flooding Traffic without VTP Pruning

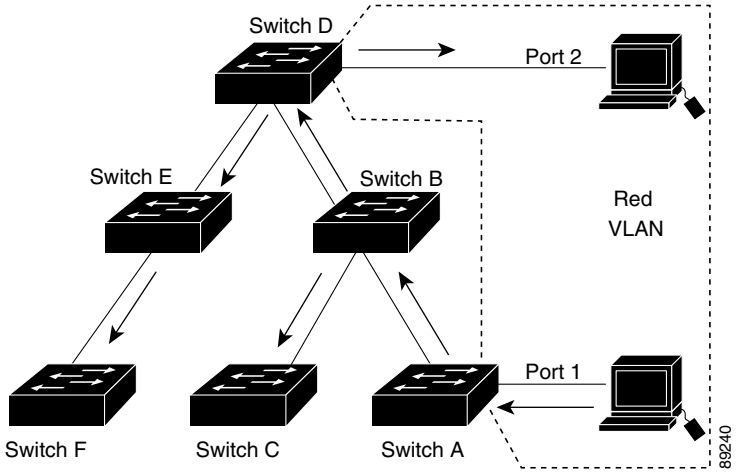
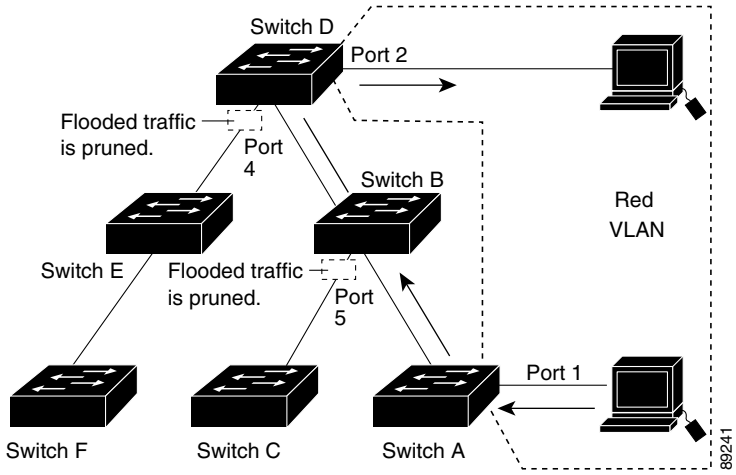


Figure 12-2 shows a switched network with VTP pruning enabled. The broadcast traffic from Switch A is not forwarded to Switches C, E, and F because traffic for the Red VLAN has been pruned on the links shown (Port 5 on Switch B and Port 4 on Switch D).

Figure 12-2 Optimized Flooded Traffic with VTP Pruning



Enabling VTP pruning on a VTP server enables pruning for the entire management domain. Making VLANs pruning-eligible or pruning-ineligible affects pruning eligibility for those VLANs on that device only (not on all switches in the VTP domain). See the "Enabling VTP Pruning" section on page 12-13. VTP pruning takes effect several seconds after you enable it. VTP pruning does not prune traffic from VLANs that are pruning-ineligible. VLAN 1 and VLANs 1002 to 1005 are always pruning-ineligible; traffic from these VLANs cannot be pruned. Extended-range VLANs (VLAN IDs higher than 1005) are also pruning-ineligible.

VTP pruning is not designed to function in VTP transparent mode. If one or more switches in the network are in VTP transparent mode, you should do one of these:

- Turn off VTP pruning in the entire network.
- Turn off VTP pruning by making all VLANs on the trunk of the switch upstream to the VTP transparent switch pruning ineligible.

To configure VTP pruning on an interface, use the **switchport trunk pruning vlan** interface configuration command (see the "Changing the Pruning-Eligible List" section on page 11-22). VTP pruning operates when an interface is trunking. You can set VLAN pruning-eligibility, whether or not VTP pruning is enabled for the VTP domain, whether or not any given VLAN exists, and whether or not the interface is currently trunking.

Configuring VTP

This section includes guidelines and procedures for configuring VTP. These sections are included:

- Default VTP Configuration, page 12-6
- VTP Configuration Options, page 12-7
- VTP Configuration Guidelines, page 12-8
- Configuring a VTP Server, page 12-9
- Configuring a VTP Client, page 12-10
- Disabling VTP (VTP Transparent Mode), page 12-12
- Enabling VTP Version 2, page 12-12
- Enabling VTP Pruning, page 12-13
- Adding a VTP Client Switch to a VTP Domain, page 12-14

Default VTP Configuration

Table 12-2 shows the default VTP configuration.

Table 12-2 Default VTP Configuration

Feature	Default Setting
VTP domain name	Null.
VTP mode	Server.
VTP version 2 enable state	Version 2 is disabled.
VTP password	None.
VTP pruning	Disabled.

VTP Configuration Options

You can configure VTP by using these configuration modes.

- VTP Configuration in Global Configuration Mode, page 12-7
- VTP Configuration in VLAN Configuration Mode, page 12-7

You access VLAN configuration mode by entering the **vlan database** privileged EXEC command.

For detailed information about **vtp** commands, see the command reference for this release.

VTP Configuration in Global Configuration Mode

You can use the **vtp** global configuration command to set the VTP password, the version, the VTP file name, the interface providing updated VTP information, the domain name, and the mode, and to disable or enable pruning. For more information about available keywords, see the command descriptions in the command reference for this release. The VTP information is saved in the VTP VLAN database. When VTP mode is transparent, the VTP domain name and mode are also saved in the switch running configuration file, and you can save it in the switch startup configuration file by entering the **copy running-config startup-config** privileged EXEC command. You must use this command if you want to save VTP mode as transparent, even if the switch resets.

When you save VTP information in the switch startup configuration file and reboot the switch, the switch configuration is determined as follows:

- If the VTP mode is transparent in the startup configuration and the VLAN database and the VTP domain name from the VLAN database matches that in the startup configuration file, the VLAN database is ignored (cleared), and the VTP and VLAN configurations in the startup configuration file are used. The VLAN database revision number remains unchanged in the VLAN database.
- If the VTP mode or domain name in the startup configuration do not match the VLAN database, the domain name and VTP mode and configuration for the first 1005 VLAN IDs use the VLAN database information.
- If the switch is running IOS Release 12.1(9)EA1 or later and you use an older configuration file to boot up the switch, the configuration file does not contain VTP or VLAN information, and the switch uses the VLAN database configurations.
- If the switch is running an IOS release earlier than 12.1(9)EA1 on the switch and you use a configuration file from IOS Release 12.1(9)EA1 or later to boot up the switch, the image on the switch does not recognize VLAN and VTP configurations in the configuration file, so the switch uses the VLAN database configuration.

VTP Configuration in VLAN Configuration Mode

You can configure all VTP parameters in VLAN configuration mode, which you access by entering the **vlan database** privileged EXEC command. For more information about available keywords, see the **vtp** VLAN configuration command description in the command reference for this release. When you enter the **exit** command in VLAN configuration mode, it applies all the commands that you entered and updates the VLAN database. VTP messages are sent to other switches in the VTP domain, and the privileged EXEC mode prompt appears.

If VTP mode is transparent, the domain name and the mode (transparent) are saved in the switch running configuration, and you can save this information in the switch startup configuration file by entering the **copy running-config startup-config** privileged EXEC command.

VTP Configuration Guidelines

These sections describe guidelines you should follow when implementing VTP in your network.

Domain Names

When configuring VTP for the first time, you must always assign a domain name. You must configure all switches in the VTP domain with the same domain name. Switches in VTP transparent mode do not exchange VTP messages with other switches, and you do not need to configure a VTP domain name for them.

Note If NVRAM and DRAM storage is sufficient, all switches in a VTP domain should be in VTP server mode.

Caution Do not configure a VTP domain if all switches are operating in VTP client mode. If you configure the domain, it is impossible to make changes to the VLAN configuration of that domain. Make sure that you configure at least one switch in the VTP domain for VTP server mode.

Passwords

You can configure a password for the VTP domain, but it is not required. If you do configure a domain password, all domain switches must share the same password and you must configure the password on each switch in the management domain. Switches without a password or with the wrong password reject VTP advertisements.

If you configure a VTP password for a domain, a switch that is booted without a VTP configuration does not accept VTP advertisements until you configure it with the correct password. After the configuration, the switch accepts the next VTP advertisement that uses the same password and domain name in the advertisement.

If you are adding a new switch to an existing network with VTP capability, the new switch learns the domain name only after the applicable password has been configured on it.

Caution When you configure a VTP domain password, the management domain does not function properly if you do not assign a management domain password to each switch in the domain.

VTP Version

Follow these guidelines when deciding which VTP version to implement:

- All switches in a VTP domain must run the same VTP version.
- A VTP version 2-capable switch can operate in the same VTP domain as a switch running VTP version 1 if version 2 is disabled on the version 2-capable switch (version 2 is disabled by default).
- Do not enable VTP version 2 on a switch unless all of the switches in the same VTP domain are version-2-capable. When you enable version 2 on a switch, all of the version-2-capable switches in the domain enable version 2. If there is a version 1-only switch, it does not exchange VTP information with switches with version 2 enabled.

- If there are TrBRF and TrCRF Token Ring networks in your environment, you must enable VTP version 2 for Token Ring VLAN switching to function properly. To run Token Ring and Token Ring-Net, disable VTP version 2.

Configuration Requirements

When you configure VTP, you must configure a trunk port so that the switch can send and receive VTP advertisements. For more information, see the "Configuring VLAN Trunks" section on page 11-15.

If you are configuring VTP on a cluster member switch to a VLAN, use the **rcommand** privileged EXEC command to log into the member switch. For more information about the command, see the command reference for this release.

If you are configuring extended-range VLANs on the switch, the switch must be in VTP transparent mode.

Configuring a VTP Server

When a switch is in VTP server mode, you can change the VLAN configuration and have it propagated throughout the network.

Note If extended-range VLANs are configured on the switch, you cannot change VTP mode to server. You receive an error message, and the configuration is not allowed.

Beginning in privileged EXEC mode, follow these steps to configure the switch as a VTP server:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vtp mode server**	Configure the switch for VTP server mode (the default).
Step 3	**vtp domain** *domain-name*	Configure the VTP administrative-domain name. The name can be from 1 to 32 characters. All switches operating in VTP server or client mode under the same administrative responsibility must be configured with the same domain name.
Step 4	**vtp password** *password*	(Optional) Set the password for the VTP domain. The password can be from 8 to 64 characters. If you configure a VTP password, the VTP domain does not function properly if you do not assign the same password to each switch in the domain.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show vtp status**	Verify your entries in the *VTP Operating Mode* and the *VTP Domain Name* fields of the display.

When you configure a domain name, it cannot be removed; you can only reassign a switch to a different domain.

To return the switch to a no-password state, use the **no vtp password** global configuration command.

This example shows how to use global configuration mode to configure the switch as a VTP server with the domain name *eng_group* and the password *mypassword*:

```
Switch# config terminal
Switch(config)# vtp mode server
Switch(config)# vtp domain eng_group
Switch(config)# vtp password mypassword
Switch(config)# end
```

You can also use VLAN configuration mode to configure VTP parameters. Beginning in privileged EXEC mode, follow these steps to use VLAN configuration mode to configure the switch as a VTP server:

	Command	Purpose
Step 1	**vlan database**	Enter VLAN configuration mode.
Step 2	**vtp server**	Configure the switch for VTP server mode (the default).
Step 3	**vtp domain** *domain-name*	Configure a VTP administrative-domain name. The name can be from 1 to 32 characters. All switches operating in VTP server or client mode under the same administrative responsibility must be configured with the same domain name.
Step 4	**vtp password** *password*	(Optional) Set a password for the VTP domain. The password can be from 8 to 64 characters. If you configure a VTP password, the VTP domain does not function properly if you do not assign the same password to each switch in the domain.
Step 5	**exit**	Update the VLAN database, propagate it throughout the administrative domain, and return to privileged EXEC mode.
Step 6	**show vtp status**	Verify your entries in the *VTP Operating Mode* and the *VTP Domain Name* fields of the display.

When you configure a domain name, it cannot be removed; you can only reassign a switch to a different domain.

To return the switch to a no-password state, use the **no vtp password** VLAN configuration command.

This example shows how to use VLAN configuration mode to configure the switch as a VTP server with the domain name *eng_group* and the password *mypassword*:

```
Switch# vlan database
Switch(vlan)# vtp server
Switch(vlan)# vtp domain eng_group
Switch(vlan)# vtp password mypassword
Switch(vlan)# exit
APPLY completed.
Exiting....
```

Configuring a VTP Client

When a switch is in VTP client mode, you cannot change its VLAN configuration. The client switch receives VTP updates from a VTP server in the VTP domain and then modifies its configuration accordingly.

Follow these guidelines:

- If extended-range VLANs are configured on the switch, you cannot change VTP mode to client. You receive an error message, and the configuration is not allowed.
- If you configure the switch for VTP client mode, the switch does not create the VLAN database file (vlan.dat). If the switch is then powered off, it resets the VTP configuration to the default. To keep the VTP configuration with VTP client mode after the switch restarts, you must first configure the VTP domain name before the VTP mode.

Caution If all switches are operating in VTP client mode, do not configure a VTP domain name. If you do, it is impossible to make changes to the VLAN configuration of that domain. Therefore, make sure you configure at least one switch as a VTP server.

Beginning in privileged EXEC mode, follow these steps to configure the switch as a VTP client:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vtp mode client**	Configure the switch for VTP client mode. The default setting is VTP server.
Step 3	**vtp domain** *domain-name*	(Optional) Enter the VTP administrative-domain name. The name can be from 1 to 32 characters. This should be the same domain name as the VTP server. All switches operating in VTP server or client mode under the same administrative responsibility must be configured with the same domain name.
Step 4	**vtp password** *password*	(Optional) Enter the password for the VTP domain.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show vtp status**	Verify your entries in the *VTP Operating Mode* and the *VTP Domain Name* fields of the display.

Use the **no vtp mode** global configuration command to return the switch to VTP server mode. To return the switch to a no-password state, use the **no vtp password** global configuration command. When you configure a domain name, it cannot be removed; you can only reassign a switch to a different domain.

Note You can also configure a VTP client by using the **vlan database** privileged EXEC command to enter VLAN configuration mode and entering the **vtp client** command, similar to the second procedure under "Configuring a VTP Server" section on page 12-9. Use the **no vtp client** VLAN configuration command to return the switch to VTP server mode or the **no vtp password** VLAN configuration command to return the switch to a no-password state. When you configure a domain name, it cannot be removed; you can only reassign a switch to a different domain.

Disabling VTP (VTP Transparent Mode)

When you configure the switch for VTP transparent mode, VTP is disabled on the switch. The switch does not send VTP updates and does not act on VTP updates received from other switches. However, a VTP transparent switch running VTP version 2 does forward received VTP advertisements on its trunk links.

Note Before you create extended-range VLANs (VLAN IDs 1006 to 4094), you must set VTP mode to transparent by using the **vtp mode transparent** global configuration command. Save this configuration to the startup configuration so that the switch boots up in VTP transparent mode. Otherwise, you lose the extended-range VLAN configuration if the switch resets and boots up in VTP server mode (the default).

Beginning in privileged EXEC mode, follow these steps to configure VTP transparent mode and save the VTP configuration in the switch startup configuration file:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vtp mode transparent**	Configure the switch for VTP transparent mode (disable VTP).
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show vtp status**	Verify your entries in the *VTP Operating Mode* and the *VTP Domain Name* fields of the display.
Step 5	**copy running-config startup-config**	(Optional) Save the configuration in the startup configuration file. **Note** Only VTP mode and domain name are saved in the switch running configuration and can be copied to the startup configuration file.

To return the switch to VTP server mode, use the **no vtp mode** global configuration command.

Note If extended-range VLANs are configured on the switch, you cannot change VTP mode to server. You receive an error message, and the configuration is not allowed.

Note You can also configure VTP transparent mode by using the **vlan database** privileged EXEC command to enter VLAN configuration mode and by entering the **vtp transparent** command, similar to the second procedure under the "Configuring a VTP Server" section on page 12-9. Use the **no vtp transparent** VLAN configuration command to return the switch to VTP server mode. If extended-range VLANs are configured on the switch, you cannot change VTP mode to server. You receive an error message, and the configuration is not allowed.

Enabling VTP Version 2

VTP version 2 is disabled by default on VTP version 2-capable switches. When you enable VTP version 2 on a switch, every VTP version 2-capable switch in the VTP domain enables version 2. You can only configure the version on switches in VTP server or transparent mode.

Caution VTP version 1 and VTP version 2 are not interoperable on switches in the same VTP domain. Every switch in the VTP domain must use the same VTP version. Do not enable VTP version 2 unless every switch in the VTP domain supports version 2.

Note In TrCRF and TrBRF Token ring environments, you must enable VTP version 2 for Token Ring VLAN switching to function properly. For Token Ring and Token Ring-Net media, VTP version 2 must be disabled.

For more information on VTP version configuration guidelines, see the "VTP Version" section on page 12-8.

Beginning in privileged EXEC mode, follow these steps to enable VTP version 2:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vtp version 2**	Enable VTP version 2 on the switch. VTP version 2 is disabled by default on VTP version 2-capable switches.
Step 3	end	Return to privileged EXEC mode.
Step 4	**show vtp status**	Verify that VTP version 2 is enabled in the *VTP V2 Mode* field of the display.

To disable VTP version 2, use the **no vtp version** global configuration command**.**

Note You can also enable VTP version 2 by using the **vlan database** privileged EXEC command to enter VLAN configuration mode and entering the **vtp v2-mode** VLAN configuration command. To disable VTP version 2, use the **no vtp v2-mode** VLAN configuration command.

Enabling VTP Pruning

Pruning increases available bandwidth by restricting flooded traffic to those trunk links that the traffic must use to access the destination devices. You can only enable VTP pruning on a switch in VTP server mode.

Beginning in privileged EXEC mode, follow these steps to enable VTP pruning in the VTP domain:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vtp pruning**	Enable pruning in the VTP administrative domain. By default, pruning is disabled. You need to enable pruning on only one switch in VTP server mode.
Step 3	end	Return to privileged EXEC mode.
Step 4	**show vtp status**	Verify your entries in the *VTP Pruning Mode* field of the display.

To disable VTP pruning, use the **no vtp pruning** global configuration command.

Note You can also enable VTP pruning by using the **vlan database** privileged EXEC command to enter VLAN configuration mode and entering the **vtp pruning** VLAN configuration command. To disable VTP pruning, use the **no vtp pruning** VLAN configuration command.

Pruning is supported with VTP version 1 and version 2. If you enable pruning on the VTP server, it is enabled for the entire VTP domain.

Only VLANs included in the pruning-eligible list can be pruned. By default, VLANs 2 through 1001 are pruning eligible on trunk ports. Reserved VLANs and extended-range VLANs cannot be pruned. To change the pruning-eligible VLANs, see the "Changing the Pruning-Eligible List" section on page 11-22.

Adding a VTP Client Switch to a VTP Domain

Before adding a VTP client to a VTP domain, always verify that its VTP configuration revision number is ***lower*** than the configuration revision number of the other switches in the VTP domain. Switches in a VTP domain always use the VLAN configuration of the switch with the highest VTP configuration revision number. If you add a switch that has a revision number higher than the revision number in the VTP domain, it can erase all VLAN information from the VTP server and VTP domain.

Beginning in privileged EXEC mode, follow these steps to verify and reset the VTP configuration revision number on a switch *before* adding it to a VTP domain:

	Command	Purpose
Step 1	**show vtp status**	Check the VTP configuration revision number. If the number is 0, add the switch to the VTP domain. If the number is greater than 0, follow these steps: **a.** Write down the domain name. **b.** Write down the configuration revision number. **c.** Continue with the next steps to reset the configuration revision number on the switch.
Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**vtp domain** *domain-name*	Change the domain name from the original one displayed in Step 1 to a new name.
Step 4	**end**	The VLAN information on the switch is updated and the configuration revision number is reset to 0. You return to privileged EXEC mode.
Step 5	**show vtp status**	Verify that the configuration revision number has been reset to 0.
Step 6	**configure terminal**	Enter global configuration mode.
Step 7	**vtp domain** *domain-name*	Enter the original domain name on the switch.
Step 8	**end**	The VLAN information on the switch is updated, and you return to privileged EXEC mode.
Step 9	**show vtp status**	(Optional) Verify that the domain name is the same as in Step 1 and that the configuration revision number is 0.

You can also change the VTP domain name by entering the **vlan database** privileged EXEC command to enter VLAN configuration mode and by entering the **vtp domain** *domain-name* command. In this mode, you must enter the **exit** command to update VLAN information and return to privileged EXEC mode.

After resetting the configuration revision number, add the switch to the VTP domain.

Note You can use the **vtp mode transparent** global configuration command or the **vtp transparent** VLAN configuration command to disable VTP on the switch, and then change its VLAN information without affecting the other switches in the VTP domain.

Monitoring VTP

You monitor VTP by displaying VTP configuration information: the domain name, the current VTP revision, and the number of VLANs. You can also display statistics about the advertisements sent and received by the switch.

Table 12-3 shows the privileged EXEC commands for monitoring VTP activity.

Table 12-3 VTP Monitoring Commands

Command	Purpose
show vtp status	Display the VTP switch configuration information.
show vtp counters	Display counters about VTP messages that have been sent and received.

CHAPTER 13

Configuring Voice VLAN

This chapter describes how to configure the voice VLAN feature on your Catalyst 3550 switch. Voice VLAN is sometimes referred to as an *auxiliary VLAN* in the Catalyst 6000 family switch documentation.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding Voice VLAN, page 13-1
- Configuring Voice VLAN, page 13-2
- Displaying Voice VLAN, page 13-6

Understanding Voice VLAN

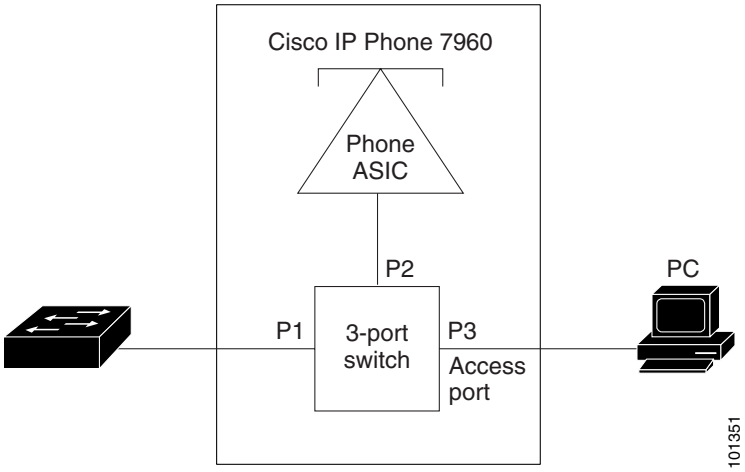
The voice VLAN feature enables access ports to carry IP voice traffic from an IP phone. The switch can connect to a Cisco 7960 IP Phone and carry IP voice traffic. Because the sound quality of an IP phone call can deteriorate if the data is unevenly sent, the switch supports quality of service (QoS) based on IEEE 802.1p class of service (CoS). QoS uses classification and scheduling to send network traffic from the switch in a predictable manner. For more information on QoS, see Chapter 28, "Configuring QoS." The Cisco 7960 IP Phone is a configurable device, and you can configure it to forward traffic with an IEEE 802.1p priority. You can configure the switch to trust or override the traffic priority assigned by an IP Phone.

The Cisco 7960 IP Phone contains an integrated three-port 10/100 switch as shown in Figure 13-1. The ports provide dedicated connections to these devices:

- Port 1 connects to the switch or other voice-over-IP (VoIP) device.
- Port 2 is an internal 10/100 interface that carries the IP phone traffic.
- Port 3 (access port) connects to a PC or other device.

Figure 13-1 shows one way to connect a Cisco 7960 IP Phone.

Figure 13-1 Cisco 7960 IP Phone Connected to a Switch



When the IP Phone connects to the switch, the access port (PC-to-telephone jack) of the IP phone can connect to a PC. Packets to and from the PC and to or from the IP phone share the same physical link to the switch and the same switch port. For deployment examples that use voice VLANs, see the"Network Configuration Examples" section on page 1-10.

Configuring Voice VLAN

This section describes how to configure voice VLAN on access ports. It contains this configuration information:

- Default Voice VLAN Configuration, page 13-2
- Voice VLAN Configuration Guidelines, page 13-3
- Configuring a Port to Connect to a Cisco 7960 IP Phone, page 13-3

Default Voice VLAN Configuration

The voice VLAN feature is disabled by default.

When the voice VLAN feature is enabled, all untagged traffic is sent according to the default CoS priority of the port.

The default CoS value is 0 for incoming traffic.

The CoS value is not trusted for IEEE 802.1p or IEEE 802.1Q tagged traffic.

The IP Phone overrides the priority of all incoming traffic (tagged and untagged) and sets the CoS value to 0.

Voice VLAN Configuration Guidelines

These are the voice VLAN configuration guidelines:

- You should configure voice VLAN on switch access ports.
- Before you enable voice VLAN, we recommend that you enable QoS on the switch by entering the **mls qos** global configuration command and configure the port trust state to trust by entering the **mls qos trust cos** interface configuration command.
- The voice VLAN should be present and active on the switch for the IP phone to correctly communicate on the voice VLAN. Use the **show vlan** privileged EXEC command to see if the VLAN is present (listed in the display). If the VLAN is not listed, see Chapter 11, "Configuring VLANs" for information on how to create the VLAN.
- The Port Fast feature is automatically enabled when voice VLAN is configured. When you disable voice VLAN, the Port Fast feature is not automatically disabled.
- When you enable port security on an interface that is also configured with a voice VLAN, you must set the maximum allowed secure addresses on the port to at least two plus the maximum number of secure addresses allowed on the access VLAN. When the port is connected to a Cisco IP phone, the IP phone requires up to two MAC addresses. The address of the IP phone is learned on the voice VLAN, and it might or might not be learned on the access VLAN. Connecting a PC to the IP phone requires additional MAC addresses.
- If any type of port security is enabled on the access VLAN, dynamic port security is automatically enabled on the voice VLAN.
- You cannot configure static secure or sticky secure MAC addresses on a voice VLAN.
- Voice VLAN ports can also be these port types:
  - Dynamic access port. See the "Configuring Dynamic Access Ports on VMPS Clients" section on page 11-30 for more information.
  - Secure port. See the "Configuring Port Security" section on page 21-7 for more information.
  - IEEE 802.1x authenticated port. See the "Using IEEE 802.1x Authentication with Voice VLAN Ports" section on page 8-15 for more information.

    Note If you enable IEEE 802.1x on an access port on which a voice VLAN is configured and to which a Cisco IP Phone is connected, the Cisco IP Phone loses connectivity to the switch for up to 30 seconds.

  - Protected port. See the "Configuring Protected Ports" section on page 21-5 for more information.

Configuring a Port to Connect to a Cisco 7960 IP Phone

Because a Cisco 7960 IP Phone also supports a connection to a PC or other device, a port connecting the switch to a Cisco 7960 IP Phone can carry mixed traffic.

You can configure the port to carry voice traffic in one of these ways:

- Configuring Ports to Carry Voice Traffic in IEEE 802.1Q Frames, page 13-4
- Configuring Ports to Carry Voice Traffic in IEEE 802.1p Priority-Tagged Frames, page 13-4

You can configure the IP phone to carry data traffic in one of these ways:

- Overriding the CoS Priority of Incoming Data Frames, page 13-5
- Configuring the IP Phone to Trust the CoS Priority of Incoming Data Frames, page 13-5

Configuring Ports to Carry Voice Traffic in IEEE 802.1Q Frames

Beginning in privileged EXEC mode, follow these steps to configure a port to carry voice traffic in IEEE 802.1Q frames for a specific VLAN:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS for the entire switch.
Step 3	**interface** *interface-id*	Specify the interface connected to the IP phone, and enter interface configuration mode.
Step 4	**mls qos trust cos**	Classify ingress traffic packets with packet CoS values. For untagged packets, use the port default CoS value.
Step 5	**switchport voice vlan** *vlan-id*	Instruct the Cisco IP Phone to forward all voice traffic through the specified VLAN. By default, the Cisco IP Phone forwards the voice traffic with an IEEE 802.1Q priority of 5. Valid VLAN IDs are from 1 to 4094.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces** *interface-id* **switchport** or **show running-config interface** *interface-id*	Verify your voice VLAN entries. Verify your QoS and voice VLAN entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove voice VLAN, use the **no switchport voice vlan** interface configuration command or the **switchport voice vlan none** interface configuration command.

Configuring Ports to Carry Voice Traffic in IEEE 802.1p Priority-Tagged Frames

Beginning in privileged EXEC mode, follow these steps to configure a port to instruct the IP phone to give voice traffic a higher priority and to forward all traffic through the native VLAN.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS for the entire switch.
Step 3	**interface** *interface-id*	Specify the interface connected to the IP phone, and enter interface configuration mode.
Step 4	**mls qos trust cos**	Classify ingress traffic packets with packet CoS values. For untagged packets, use the port default CoS value.

	Command	Purpose
Step 5	**switchport voice vlan dot1p**	Instruct the switch port to use IEEE 802.1p priority tagging for voice traffic and to use the default native VLAN (VLAN 0) to carry all traffic. By default, the Cisco IP phone forwards the voice traffic with an IEEE 802.1p priority of 5.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces** *interface-id* **switchport** or **show running-config interface** *interface-id*	Verify your voice VLAN entries. Verify your QoS and voice VLAN entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the port to its default setting, use the **no switchport voice vlan** interface configuration command.

Overriding the CoS Priority of Incoming Data Frames

You can connect a PC or other data device to a Cisco 7960 IP Phone port. The PC can generate packets with an assigned CoS value. You can configure the switch to override the priority of frames arriving on the IP phone port from connected devices.

Beginning in privileged EXEC mode, follow these steps to override the CoS priority received from the nonvoice port on the Cisco 7960 IP Phone:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the interface connected to the IP phone.
Step 3	**switchport priority extend cos** *value*	Set the IP phone access port to override the priority received from the PC or the attached device. The CoS value is a number from 0 to 7. Seven is the highest priority. The default is 0.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **switchport**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no switchport priority extend** interface configuration command or the **switchport priority extend cos 0** interface configuration command to return the port to its default setting.

Configuring the IP Phone to Trust the CoS Priority of Incoming Data Frames

You can connect a PC or other data device to a Cisco 7960 IP Phone port. The PC can generate packets with an assigned CoS value. You can configure the switch to trust the priority of frames arriving on the IP phone port from connected devices.

Beginning in privileged EXEC mode, follow these steps to trust the CoS priority received from the nonvoice port on the Cisco 7960 IP Phone:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the interface connected to the IP phone.
Step 3	**switchport priority extend trust**	Set the IP phone access port to trust the priority received from the PC or the attached device.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **switchport**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no switchport priority extend** interface configuration command or the **switchport priority extend cos 0** interface configuration command to return the port to its default setting.

Displaying Voice VLAN

To display voice VLAN for an interface, use the **show interfaces** *interface-id* **switchport** privileged EXEC command.

For detailed information about the fields in the display, see the command reference for this release.

CHAPTER 14

Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling

Virtual private networks (VPNs) provide enterprise-scale connectivity on a shared infrastructure, often Ethernet-based, with the same security, prioritization, reliability, and manageability requirements of private networks. Tunneling is a feature designed for service providers who carry traffic of multiple customers across their networks and are required to maintain the VLAN and Layer 2 protocol configurations of each customer without impacting the traffic of other customers. The Catalyst 3550 switch supports IEEE 802.1Q tunneling and Layer 2 protocol tunneling.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter contains these sections:

- Understanding IEEE 802.1Q Tunneling, page 14-1
- Configuring IEEE 802.1Q Tunneling, page 14-4
- Understanding Layer 2 Protocol Tunneling, page 14-7
- Configuring Layer 2 Protocol Tunneling, page 14-9
- Monitoring and Maintaining Tunneling Status, page 14-17

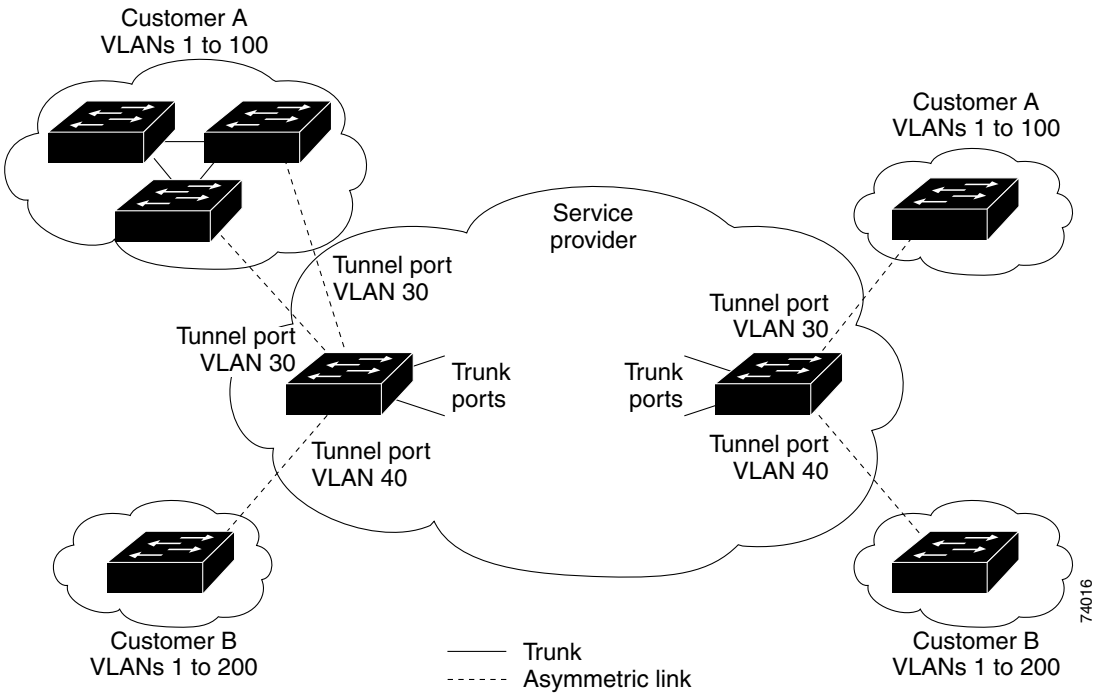
Understanding IEEE 802.1Q Tunneling

Business customers of service providers often have specific requirements for VLAN IDs and the number of VLANs to be supported. The VLAN ranges required by different customers in the same service-provider network might overlap, and traffic of customers through the infrastructure might be mixed. Assigning a unique range of VLAN IDs to each customer would restrict customer configurations and could easily exceed the VLAN limit (4096) of the IEEE 802.1Q specification.

Using the IEEE 802.1Q tunneling feature, service providers can use a single VLAN to support customers who have multiple VLANs. Customer VLAN IDs are preserved, and traffic from different customers is segregated within the service-provider network, even when they appear to be in the same VLAN. Using IEEE 802.1Q tunneling expands VLAN space by using a VLAN-in-VLAN hierarchy by again tagging the tagged packets. A port configured to support IEEE 802.1Q tunneling is called a tunnel port. When you configure tunneling, you assign a tunnel port to a VLAN ID that is dedicated to tunneling. Each customer requires a separate service-provider VLAN ID, but that VLAN ID supports all of the customer's VLANs.

Customer traffic tagged in the normal way with appropriate VLAN IDs comes from an IEEE 802.1Q trunk port on the customer device and into a tunnel port on the service-provider edge switch. The link between the customer device and the edge switch is asymmetric because one end is configured as an IEEE 802.1Q trunk port, and the other end is configured as a tunnel port. You assign the tunnel port interface to an access VLAN ID that is unique to each customer. See Figure 14-1.

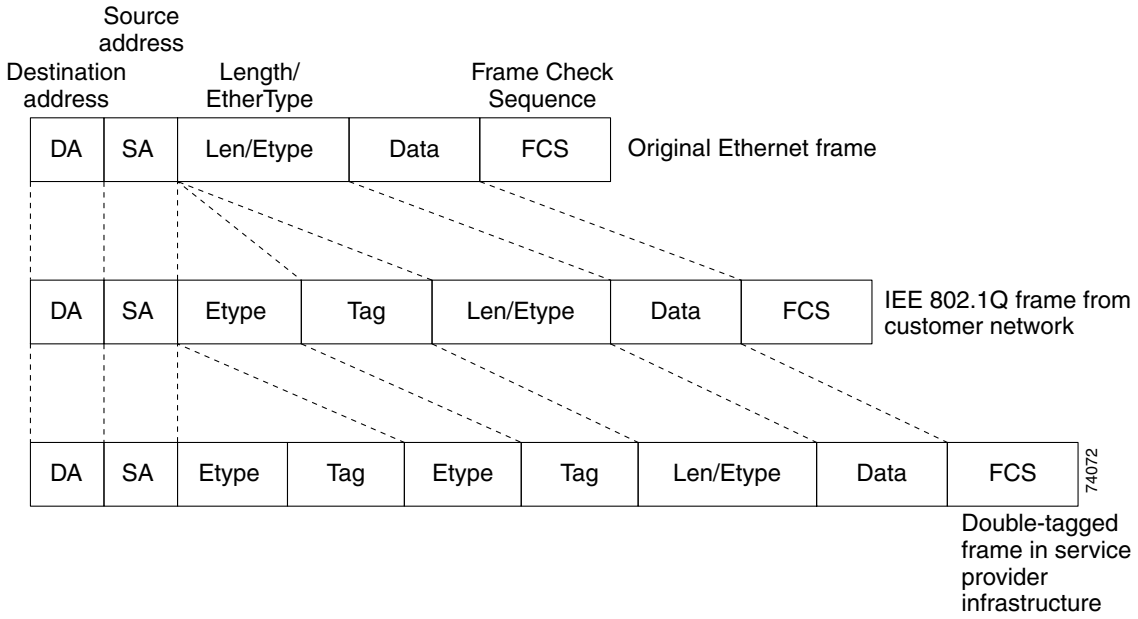
Figure 14-1 IEEE 802.1Q Tunnel Ports in a Service-Provider Network



Packets coming from the customer trunk port into the tunnel port on the service-provider edge switch are normally IEEE 802.1Q-tagged with the appropriate VLAN ID. When the tagged packets exit the trunk port into the service-provider network, they are encapsulated with another layer of an IEEE 802.1Q tag (called the *metro tag*) that contains the VLAN ID that is unique to the customer. The original customer IEEE 802.1Q tag is preserved in the encapsulated packet. Therefore, packets entering the service-provider network are double-tagged, with the outer (metro) tag containing the customer's access VLAN ID, and the inner VLAN ID being that of the incoming traffic.

When the double-tagged packet enters another trunk port in a service-provider core switch, the metro tag is stripped as the switch processes the packet. When the packet exits another trunk port on the same core switch, the same metro tag is again added to the packet. Figure 14-2 shows the tag structures of the Ethernet packets starting with the original, or normal, frame.

Figure 14-2 Original (Normal), IEEE 802.1Q, and Double-Tagged Ethernet Packet Formats



When the packet enters the trunk port of the service-provider egress switch, the metro tag is again stripped as the switch processes the packet. However, the metro tag is not added when the packet is sent out the tunnel port on the edge switch into the customer network. The packet is sent as a normal IEEE 802.1Q-tagged frame to preserve the original VLAN numbers in the customer network.

In Figure 14-1, Customer A was assigned VLAN 30, and Customer B was assigned VLAN 40. Packets entering the edge switch tunnel ports with IEEE 802.1Q tags are double-tagged when they enter the service-provider network, with the outer tag containing VLAN ID 30 or 40, appropriately, and the inner tag containing the original VLAN number, for example, VLAN 100. Even if both Customers A and B have VLAN 100 in their networks, the traffic remains segregated within the service-provider network because the metro tag is different. Each customer controls its own VLAN numbering space, which is independent of the VLAN numbering space used by other customers and the VLAN numbering space used by the service-provider network.

At the outbound tunnel port, the original VLAN numbers on the customer's network are recovered. It is possible to have multiple levels of tunneling and tagging, but the switch supports only one level in this release.

All packets entering the service-provider network through a tunnel port on an edge switch are treated as untagged packets, whether they are untagged or already tagged with IEEE 802.1Q headers. The packets are encapsulated with the metro tag VLAN ID (set to the access VLAN of the tunnel port) when they are sent through the service-provider network on an IEEE 802.1Q trunk port. The priority field on the metro tag is set to the interface class of service (CoS) priority configured on the tunnel port. (The default is zero if none is configured.)

Configuring IEEE 802.1Q Tunneling

This section includes this information about configuring IEEE 802.1Q tunneling:

- Default IEEE 802.1Q Tunneling Configuration, page 14-4
- IEEE 802.1Q Tunneling Configuration Guidelines, page 14-4
- IEEE 802.1Q Tunneling and Other Features, page 14-5
- Configuring an IEEE 802.1Q Tunneling Port, page 14-6

Default IEEE 802.1Q Tunneling Configuration

By default, IEEE 802.1Q tunneling is disabled because the default switchport mode is dynamic desirable. Tagging of IEEE 802.1Q native VLAN packets on all IEEE 802.1Q trunk ports is also disabled.

IEEE 802.1Q Tunneling Configuration Guidelines

When you configure IEEE 802.1Q tunneling, you should always use asymmetrical links for traffic going through a tunnel and should dedicate one VLAN for each tunnel. You should also be aware of configuration requirements for native VLANs and maximum transmission units (MTUs). For more information about MTUs, see the "System MTU" section on page 14-5.

Native VLANs

When configuring IEEE 802.1Q tunneling on an edge switch, you must use IEEE 802.1Q trunk ports for sending packets into the service-provider network. However, packets going through the core of the service-provider network can be carried through IEEE 802.1Q trunks, ISL trunks, or nontrunking links. When IEEE 802.1Q trunks are used in these core switches, the native VLANs of the IEEE 802.1Q trunks must not match any native VLAN of the nontrunking (tunneling) port on the same switch because traffic on the native VLAN would not be tagged on the IEEE 802.1Q sending trunk port.

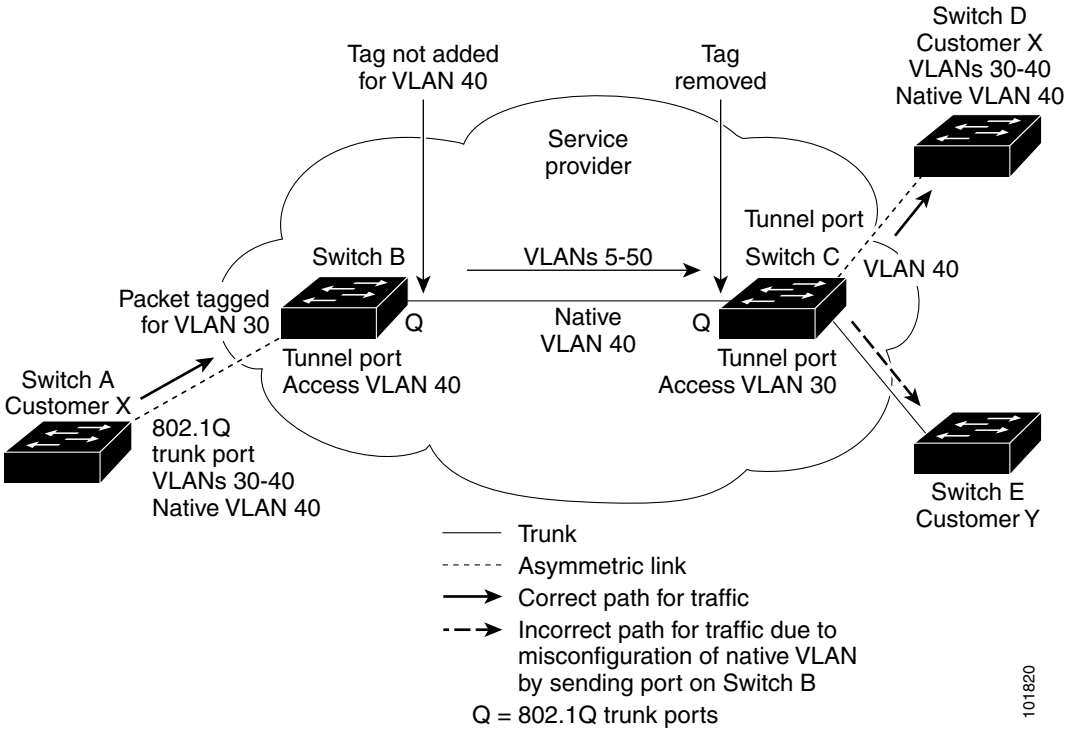
See Figure 14-3. VLAN 40 is configured as the native VLAN for the IEEE 802.1Q trunk port from Customer X at the ingress edge switch in the service-provider network (Switch B). Switch A of Customer X sends a tagged packet on VLAN 30 to the ingress tunnel port of Switch B in the service-provider network, which belongs to access VLAN 40. Because the access VLAN of the tunnel port (VLAN 40) is the same as the native VLAN of the edge-switch trunk port (VLAN 40), the metro tag is not added to tagged packets received from the tunnel port. The packet carries only the VLAN 30 tag through the service-provider network to the trunk port of the egress-edge switch (Switch C) and is misdirected through the egress switch tunnel port to Customer Y.

These are some ways to solve this problem:

- Use ISL trunks between core switches in the service-provider network. Although customer interfaces connected to edge switches must be IEEE 802.1Q trunks, we recommend using ISL trunks for connecting switches in the core layer.
- Use the **vlan dot1q tag native** global configuration command to configure the edge switch so that all packets going out an IEEE 802.1Q trunk, including the native VLAN, are tagged. If the switch is configured to tag native VLAN packets on all IEEE 802.1Q trunks, the switch accepts untagged packets, but sends only tagged packets.

- Ensure that the native VLAN ID on the edge-switch trunk port is not within the customer VLAN range. For example, if the trunk port carries traffic of VLANs 100 to 200, assign the native VLAN a number outside that range.

Figure 14-3 Potential Problem with IEEE 802.1Q Tunneling and Native VLANs



System MTU

The default system MTU for traffic on the switch is 1500 bytes. You can configure the switch to support larger frames by using the **system mtu** global configuration command. Because the IEEE 802.1Q tunneling feature increases the frame size by 4 bytes when the metro tag is added, you must configure all switches in the service-provider network to be able to process larger frames by increasing the switch system MTU size to at least 1504 bytes. The maximum allowable system MTU for Catalyst 3550 Gigabit Ethernet switches is 2000 bytes; the maximum system MTU for Fast Ethernet switches is 1546 bytes.

IEEE 802.1Q Tunneling and Other Features

Although IEEE 802.1Q tunneling works well for Layer 2 packet switching, there are incompatibilities between some Layer 2 features and Layer 3 switching.

- A tunnel port cannot be a routed port.
- IP routing is not supported on a VLAN that includes IEEE 802.1Q ports. Packets received from a tunnel port are forwarded based only on Layer 2 information. If routing is enabled on a switch virtual interface (SVI) that includes tunnel ports, untagged IP packets received from the tunnel port are recognized and routed by the switch. Customer can access the internet through its native VLAN. If this access is not needed, you should not configure SVIs on VLANs that include tunnel ports.

- Fallback bridging is not supported on tunnel ports. Because all IEEE 802.1Q-tagged packets received from a tunnel port are treated as non-IP packets, if fallback bridging is enabled on VLANs that have tunnel ports configured, IP packets would be improperly bridged across VLANs. Therefore, you must *not* enable fallback bridging on VLANs with tunnel ports.
- Tunnel ports do not support IP access control lists (ACLs).
- Layer 3 quality of service (QoS) ACLs and other QoS features related to Layer 3 information are not supported on tunnel ports. MAC-based QoS is supported on tunnel ports.
- EtherChannel port groups are compatible with tunnel ports as long as the IEEE 802.1Q configuration is consistent within an EtherChannel port group.
- Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), and UniDirectional Link Detection (UDLD) are supported on IEEE 802.1Q tunnel ports.
- Dynamic Trunking Protocol (DTP) is not compatible with IEEE 802.1Q tunneling because you must manually configure asymmetric links with tunnel ports and trunk ports.
- Loopback detection is supported on IEEE 802.1Q tunnel ports.
- When a port is configured as an IEEE 802.1Q tunnel port, spanning-tree bridge protocol data unit (BPDU) filtering is automatically enabled on the interface. Cisco Discovery Protocol (CDP) is automatically disabled on the interface.

Configuring an IEEE 802.1Q Tunneling Port

Beginning in privileged EXEC mode, follow these steps to configure a port as an IEEE 802.1Q tunnel port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode and the interface to be configured as a tunnel port. This should be the edge port in the service-provider network that connects to the customer switch. Valid interfaces include physical interfaces and port-channel logical interfaces (port channels 1 to 64).
Step 3	**switchport access vlan** *vlan-id*	Specify the default VLAN, which is used if the interface stops trunking. This VLAN ID is specific to the particular customer.
Step 4	**switchport mode dot1q-tunnel**	Set the interface as an IEEE 802.1Q tunnel port.
Step 5	**exit**	Return to global configuration mode.
Step 6	**vlan dot1q tag native**	(Optional) Set the switch to enable tagging of native VLAN packets on all IEEE 802.1Q trunk ports. When not set, and a customer VLAN ID is the same as the native VLAN, the trunk port does not apply a metro tag, and packets could be sent to the wrong destination.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show dot1q-tunnel**	Display the tunnel ports on the switch.
Step 9	**show vlan dot1q tag native**	Display IEEE 802.1Q native VLAN tagging status.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no switchport mode dot1q-tunnel** interface configuration command to return the port to the default state of dynamic desirable. Use the **no vlan dot1q tag native** global configuration command to disable tagging of native VLAN packets.

This example shows how to configure an interface as a tunnel port, enable tagging of native VLAN packets, and verify the configuration. In this configuration, the VLAN ID for the customer connected to Gigabit Ethernet interface 7 is VLAN 22.

```
Switch(config)# interface gigabitethernet0/7
Switch(config-if)# switchport access vlan 22
% Access VLAN does not exist. Creating vlan 22
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# exit
Switch(config)# vlan dot1q tag native
Switch(config)# end
Switch# show dot1q-tunnel interface gigabitethernet0/7
Port
-----
Gi0/1Port
-----
Switch# show vlan dot1q tag native
dot1q native vlan tagging is enabled
```

Understanding Layer 2 Protocol Tunneling

Customers at different sites connected across a service-provider network need to use various Layer 2 protocols to scale their topologies to include all remote sites, as well as the local sites. STP must run properly, and every VLAN should build a proper spanning tree that includes the local site and all remote sites across the service-provider network. Cisco Discovery Protocol (CDP) must discover neighboring Cisco devices from local and remote sites. VLAN Trunking Protocol (VTP) must provide consistent VLAN configuration throughout all sites in the customer network.

When protocol tunneling is enabled, edge switches on the inbound side of the service-provider network encapsulate Layer 2 protocol packets with a special MAC address and send them across the service-provider network. Core switches in the network do not process these packets but forward them as normal packets. Layer 2 protocol data units (PDUs) for CDP, STP, or VTP cross the service-provider network and are delivered to customer switches on the outbound side of the service-provider network. Identical packets are received by all customer ports on the same VLANs with these results:

- Users on each of a customer's sites can properly run STP, and every VLAN can build a correct spanning tree based on parameters from all sites and not just from the local site.
- CDP discovers and shows information about the other Cisco devices connected through the service-provider network.
- VTP provides consistent VLAN configuration throughout the customer network, propagating to all switches through the service provider.

Note To provide interoperability with third-party vendors, you can use the Layer 2 protocol-tunnel bypass feature. Bypass mode transparently forwards control PDUs to vendor switches that have different ways of controlling protocol tunneling. You implement bypass mode by enabling Layer 2 protocol tunneling on the egress trunk port. When Layer 2 protocol tunneling is enabled on the trunk port, the encapsulated tunnel MAC address is removed and the protocol packets have their normal MAC address.

Layer 2 protocol tunneling can be used independently or can enhance IEEE 802.1Q tunneling. If protocol tunneling is not enabled on IEEE 802.1Q tunneling ports, remote switches at the receiving end of the service-provider network do not receive the PDUs and cannot properly run STP, CDP, and VTP. When protocol tunneling *is* enabled, Layer 2 protocols within each customer's network are totally separate from those running within the service-provider network. Customer switches on different sites that send traffic through the service-provider network with IEEE 802.1Q tunneling achieve complete knowledge of the customer's VLAN. If IEEE 802.1Q tunneling is not used, you can still enable Layer 2 protocol tunneling by connecting to the customer switch through access ports and by enabling tunneling on the service-provider access port.

As an example, in Figure 14-4, Customer X has four switches in the same VLAN that are connected through the service-provider network. If the network does not tunnel PDUs, switches on the far ends of the network cannot properly run STP, CDP, and VTP. For example, STP for a VLAN on a switch in Customer X, Site 1 will build a spanning tree on the switches at that site without considering convergence parameters based on Customer X's switch in Site 2. This could result in the topology shown in Figure 14-5.

Figure 14-4 Layer 2 Protocol Tunneling

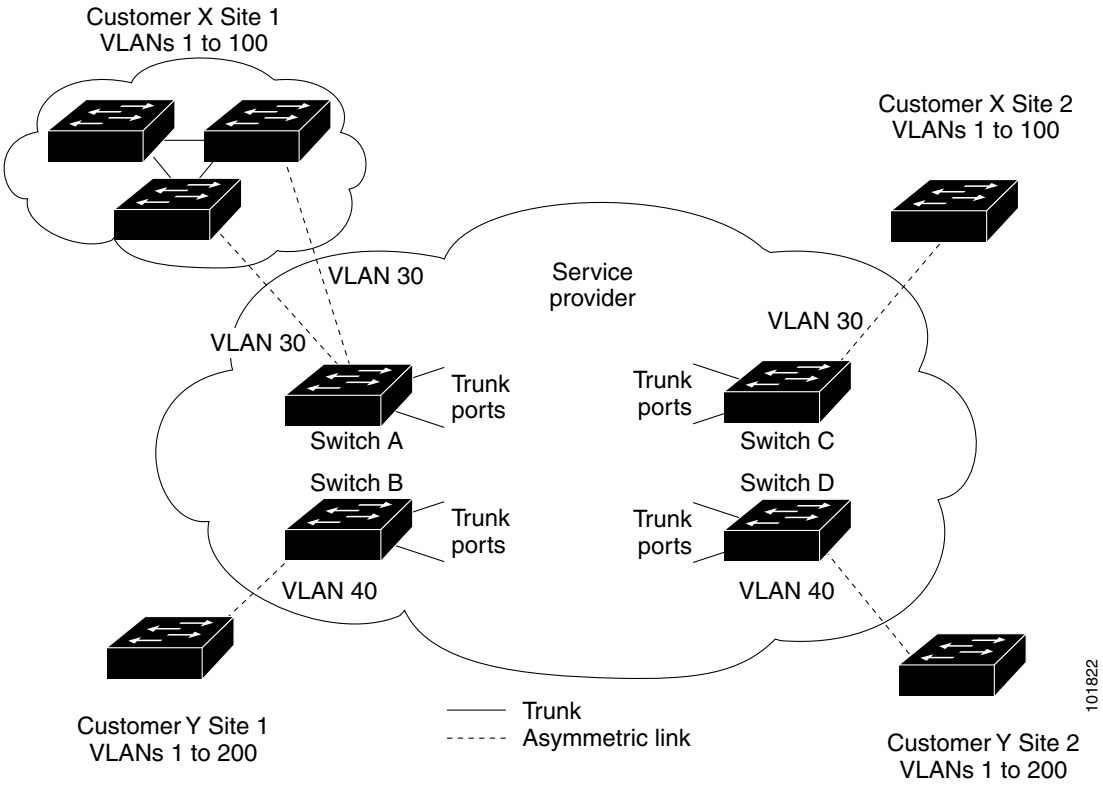
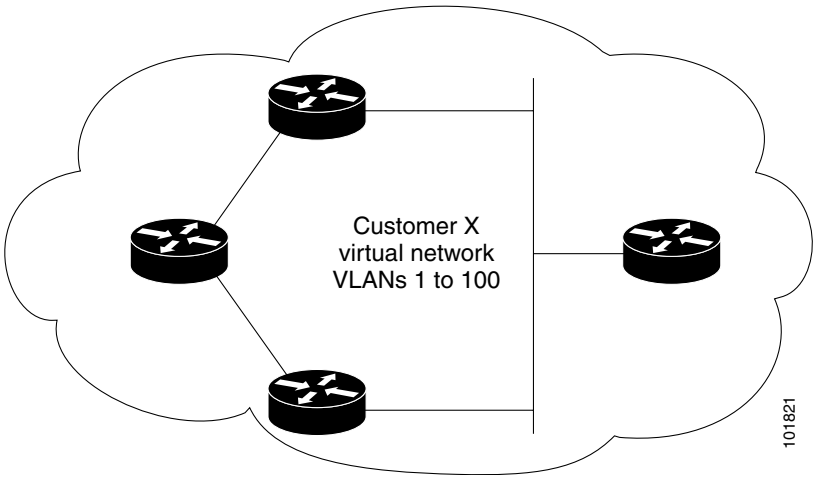


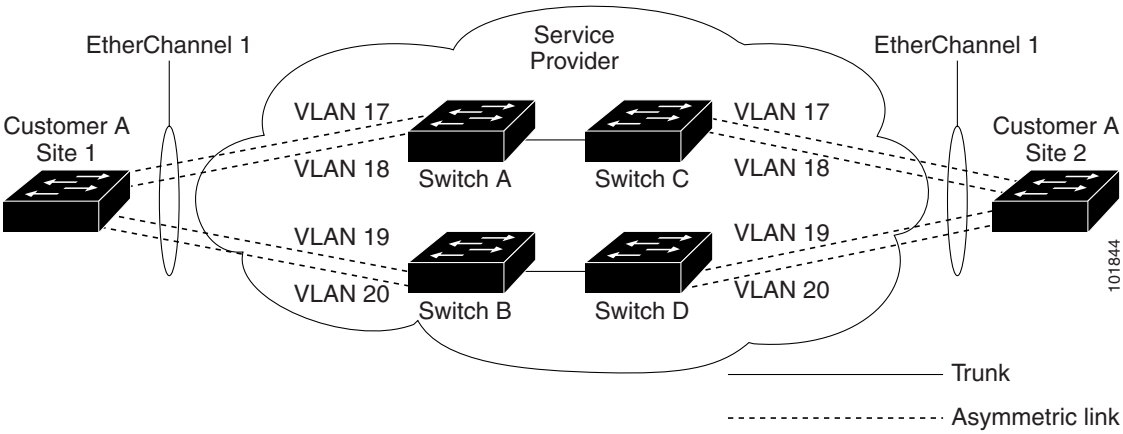
Figure 14-5 Layer 2 Network Topology without Proper Convergence



In a SP network, you can use Layer 2 protocol tunneling to enhance the creation of EtherChannels by emulating a point-to-point network topology. When you enable protocol tunneling (PAgP or LACP) on the SP switch, remote customer switches receive the PDUs and can negotiate the automatic creation of EtherChannels.

For example, in Figure 14-6, Customer A has two switches in the same VLAN that are connected through the SP network. When the network tunnels PDUs, switches on the far ends of the network can negotiate the automatic creation of EtherChannels without needing dedicated lines. See the "Configuring Layer 2 Tunneling for EtherChannels" section on page 14-13 for instructions on configuring Layer 2 protocol tunneling for EtherChannels.

Figure 14-6 Layer 2 Protocol Tunneling for EtherChannels



Configuring Layer 2 Protocol Tunneling

You can enable Layer 2 protocol tunneling (by protocol) on the access ports or tunnel ports that are connected to the customer in the edge switches of the service-provider network. The service-provider edge switches connected to the customer switch perform the tunneling process. Edge-switch tunnel ports are connected to customer IEEE 802.1Q trunk ports. Edge-switch access ports are connected to customer access ports.

The switch supports Layer 2 protocol tunneling for CDP, STP, and VTP. For emulated point-to-point network topologies, it also supports PAgP, LACP, and UDLD protocols.

Caution PAgP, LACP, and UDLD protocol tunneling is only intended to emulate a point-to-point topology. An erroneous configuration that sends tunneled packets to many ports could lead to a network failure.

When the Layer 2 PDUs that entered the service-provider inbound edge switch through the tunnel port or the access port exit through its trunk port into the service-provider network, the switch overwrites the customer PDU-destination MAC address with a well-known Cisco proprietary multicast address (01-00-0c-cd-cd-d0). If IEEE 802.1Q tunneling is enabled, packets are also double-tagged; the outer tag is the customer metro tag, and the inner tag is the customer's VLAN tag. The core switches ignore the inner tags and forward the packet to all trunk ports in the same metro VLAN. The edge switches on the outbound side restore the proper Layer 2 protocol and MAC address information and forward the packets to all tunnel or access ports in the same metro VLAN. Therefore, the Layer 2 PDUs remain intact and are delivered across the service-provider network to the other side of the customer network.

See Figure 14-4, with Customer X and Customer Y in access VLANs 30 and 40, respectively. Asymmetric links connect the Customers in Site 1 to edge switches in the service-provider network. The Layer 2 PDUs (for example, BPDUs) coming into Switch 2 from Customer Y in Site 1 are forwarded to the infrastructure as double-tagged packets with the well-known MAC address as the destination MAC address. These double-tagged packets have the metro VLAN tag of 40, as well as an inner VLAN tag (for example, VLAN 100). When the double-tagged packets enter Switch D, the metro VLAN tag 40 is removed. The well-known MAC address is replaced with the respective Layer 2 protocol MAC address, and the packet is sent to Customer Y on Site 2 as a single-tagged frame in VLAN 100.

You can also enable Layer 2 protocol tunneling on access ports on the edge switch connected to access ports on the customer switch. In this case, the encapsulation and de-encapsulation process is the same as described in the previous paragraph, except that the packets are not double-tagged in the service-provider network. The single tag is the customer-specific access VLAN tag.

This section has this information about configuring Layer 2 protocol tunneling:

- Default Layer 2 Protocol Tunneling Configuration, page 14-10
- Layer 2 Protocol Tunneling Configuration Guidelines, page 14-11
- Configuring Layer 2 Tunneling, page 14-12
- Configuring Layer 2 Tunneling for EtherChannels, page 14-13

Default Layer 2 Protocol Tunneling Configuration

Table 14-1 shows the default Layer 2 protocol tunneling configuration.

Table 14-1 Default Layer 2 Ethernet Interface VLAN Configuration

Feature	Default Setting
Layer 2 protocol tunneling	Disabled.
Shutdown threshold	None set.
Drop threshold	None set.
CoS value	If a CoS value is configured on the interface for data packets, that value is the default used for Layer 2 PDUs. If none is configured, the default is 5.

Layer 2 Protocol Tunneling Configuration Guidelines

These are some configuration guidelines and operating characteristics of Layer 2 protocol tunneling:

- The switch supports tunneling of CDP, STP, including multiple STP (MSTP), and VTP. Protocol tunneling is disabled by default but can be enabled for the individual protocols on IEEE 802.1Q tunnel ports or on access ports.
- Tunneling is not supported on trunk ports. If you enter the **l2protocol-tunnel** interface configuration command on a trunk port, the command is accepted, but Layer 2 tunneling does not take affect unless you change the port to a tunnel port or an access port.
- The switch supports PAgP, LACP, and UDLD tunneling for emulated point-to-point network topologies. Protocol tunneling is disabled by default but can be enabled for the individual protocols on IEEE 802.1Q tunnel ports or on access ports.
- If you enable PAgP or LACP tunneling, we recommend that you also enable UDLD on the interface for faster link-failure detection.
- Loopback detection is not supported on Layer 2 protocol tunneling of PAgP, LACP, or UDLD packets.
- EtherChannel port groups are compatible with tunnel ports when the IEEE 802.1Q configuration is consistent within an EtherChannel port group.
- If an encapsulated PDU (with the proprietary destination MAC address) is received from a tunnel port or an access port with Layer 2 tunneling enabled, the tunnel port is shut down to prevent loops. The port also shuts down when a configured shutdown threshold for the protocol is reached. You can manually re-enable the port (by entering a **shutdown** and a **no shutdown** command sequence). If errdisable recovery is enabled, the operation is retried after a specified time interval.
- Only decapsulated PDUs are forwarded to the customer network. The spanning-tree instance running on the service-provider network does not forward BPDUs to tunnel ports. CDP packets are not forwarded from tunnel ports.
- When protocol tunneling is enabled on an interface, you can set a per-protocol, per-port, shutdown threshold for the PDUs generated by the customer network. If the limit is exceeded, the port shuts down. You can also limit BPDU rate by using QoS ACLs and policy maps on a tunnel port.
- When protocol tunneling is enabled on an interface, you can set a per-protocol, per-port, drop threshold for the PDUs generated by the customer network. If the limit is exceeded, the port drops PDUs until the rate at which it receives them is below the drop threshold.
- Because tunneled PDUs (especially STP BPDUs) must be delivered to all remote sites so that the customer virtual network operates properly, you can give PDUs higher priority within the service-provider network than data packets received from the same tunnel port. By default, the PDUs use the same CoS value as data packets.

Configuring Layer 2 Tunneling

Beginning in privileged EXEC mode, follow these steps to configure a port for Layer 2 protocol tunneling:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the interface to be configured as a tunnel port. This should be the edge port in the service-provider network that connects to the customer switch. Valid interfaces can be physical interfaces and port-channel logical interfaces (port channels 1 to 64).
Step 3	**switchport mode access** or **switchport mode dot1q-tunnel**	Configure the interface as an access port or as an IEEE 802.1Q tunnel port.
Step 4	**l2protocol-tunnel** [**cdp** | **stp** | **vtp**]	Enable protocol tunneling for the desired protocol. If no keyword is entered, tunneling is enabled for all three Layer 2 protocols.
Step 5	**l2protocol-tunnel shutdown-threshold** [**cdp** | **stp** | **vtp**] *value*	(Optional) Configure the threshold for packets-per-second accepted for encapsulation. The interface is disabled if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured. **Note** If you also set a drop threshold on this interface, the shutdown-threshold value must be greater than or equal to the drop-threshold value.
Step 6	**l2protocol-tunnel drop-threshold** [**cdp** | **stp** | **vtp**] *value*	(Optional) Configure the threshold for packets-per-second accepted for encapsulation. The interface drops packets if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured. If you also set a shutdown threshold on this interface, the drop-threshold value must be less than or equal to the shutdown-threshold value.
Step 7	**exit**	Return to global configuration mode.
Step 8	**errdisable recovery cause l2ptguard**	(Optional) Configure the recovery mechanism from a Layer 2 maximum-rate error so that the interface is re-enabled and can try again. Errdisable recovery is disabled by default; when enabled, the default time interval is 300 seconds.
Step 9	**l2protocol-tunnel cos** *value*	(Optional) Configure the CoS value for all tunneled Layer 2 PDUs. The range is 0 to 7; the default is the default CoS value for the interface. If none is configured, the default is 5.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**show l2protocol**	Display the Layer 2 tunnel ports on the switch, including the protocols configured, the thresholds, and the counters.
Step 12	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no l2protocol-tunnel** [**cdp** | **stp** | **vtp**] interface configuration command to disable protocol tunneling for one of the Layer 2 protocols or for all three. Use the **no l2protocol-tunnel shutdown-threshold** [**cdp** | **stp** | **vtp**] and the **no l2protocol-tunnel drop-threshold** [**cdp** | **stp** | **vtp**] commands to return the shutdown and drop thresholds to the default settings.

This example shows how to configure Layer 2 protocol tunneling for CDP, STP, and VTP and to verify the configuration.

```
Switch(config)# interface FastEthernet0/11
Switch(config-if)# l2protocol-tunnel cdp
Switch(config-if)# l2protocol-tunnel stp
Switch(config-if)# l2protocol-tunnel vtp
Switch(config-if)# l2protocol-tunnel shutdown-threshold 1500
Switch(config-if)# l2protocol-tunnel drop-threshold 1000
Switch(config-if)# exit
Switch(config)# l2protocol-tunnel cos 7
Switch(config)# end
Switch# show l2protocol
COS for Encapsulated Packets: 7
Port    Protocol Shutdown  Drop      Encapsulation Decapsulation Drop
                 Threshold Threshold Counter       Counter       Counter
------- -------- --------- --------- ------------- ------------- -------------
Fa0/11  cdp           1500      1000 2288          2282          0
        stp           1500      1000 116           13            0
        vtp           1500      1000 3             67            0
        pagp          ----      ---- 0             0             0
        lacp          ----      ---- 0             0             0
        udld          ----      ---- 0             0             0
```

Configuring Layer 2 Tunneling for EtherChannels

To configure Layer 2 point-to-point tunneling to facilitate the automatic creation of EtherChannels, you need to configure both the SP edge switch and the customer switch.

Configuring the SP Edge Switch

Beginning in privileged EXEC mode, follow these steps to configure a SP edge switch for Layer 2 protocol tunneling for EtherChannels:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the interface to be configured as a tunnel port. This should be the edge port in the SP network that connects to the customer switch. Valid interfaces are physical interfaces.
Step 3	**switchport mode dot1q-tunnel**	Configure the interface as an IEEE 802.1Q tunnel port.

	Command	Purpose
Step 4	**l2protocol-tunnel point-to-point** [**pagp** | **lacp** | **udld**]	(Optional) Enable point-to-point protocol tunneling for the desired protocol. If no keyword is entered, tunneling is enabled for all three protocols. **Caution** To avoid a network failure, make sure that the network is a point-to-point topology before you enable tunneling for PAgP, LACP, or UDLD packets.
Step 5	**l2protocol-tunnel shutdown-threshold** [**point-to-point** [**pagp** | **lacp** | **udld**]] *value*	(Optional) Configure the threshold for packets-per-second accepted for encapsulation. The interface is disabled if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured. **Note** If you also set a drop threshold on this interface, the shutdown-threshold value must be greater than or equal to the drop-threshold value.
Step 6	**l2protocol-tunnel drop-threshold** [**point-to-point** [**pagp** | **lacp** | **udld**]] *value*	(Optional) Configure the threshold for packets-per-second accepted for encapsulation. The interface drops packets if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured. **Note** If you also set a shutdown threshold on this interface, the drop-threshold value must be less than or equal to the shutdown-threshold value.
Step 7	no cdp enable	Disable CDP on the interface.
Step 8	spanning-tree bpdufilter enable	Enable BPDU filtering on the interface.
Step 9	**exit**	Return to global configuration mode.
Step 10	**errdisable recovery cause l2ptguard**	(Optional) Configure the recovery mechanism from a Layer 2 maximum-rate error so that the interface is re-enabled and can try again. Errdisable recovery is disabled by default; when enabled, the default time interval is 300 seconds.
Step 11	**l2protocol-tunnel cos** *value*	(Optional) Configure the CoS value for all tunneled Layer 2 PDUs. The range is 0 to 7; the default is the default CoS value for the interface. If none is configured, the default is 5.
Step 12	**end**	Return to privileged EXEC mode.
Step 13	**show l2protocol**	Display the Layer 2 tunnel ports on the switch, including the protocols configured, the thresholds, and the counters.
Step 14	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no l2protocol-tunnel** [**point-to-point** [**pagp** | **lacp** | **udld**]] interface configuration command to disable point-to-point protocol tunneling for one of the Layer 2 protocols or for all three. Use the **no l2protocol-tunnel shutdown-threshold** [**point-to-point** [**pagp** | **lacp** | **udld**]] and the **no l2protocol-tunnel drop-threshold** [[**point-to-point** [**pagp** | **lacp** | **udld**]] commands to return the shutdown and drop thresholds to the default settings.

Configuring the Customer Switch

After configuring the SP edge switch, begin in privileged EXEC mode and follow these steps to configure a customer switch for Layer 2 protocol tunneling for EtherChannels:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter the interface configuration mode. This should be the customer switch port.
Step 3	**switchport trunk encapsulation dot1q**	Set the trunking encapsulation format to IEEE 802.1Q.
Step 4	switchport mode trunk	Enable trunking on the interface.
Step 5	udld enable	Enable UDLD in **normal** mode on the interface.
Step 6	**channel-group** *channel-group-number* **mode desirable**	Assign the interface to a channel group, and specify **desirable** for the PAgP mode. For more information about configuring EtherChannels, see Chapter 30, "Configuring EtherChannels."
Step 7	exit	Return to global configuration mode.
Step 8	interface port-channel *port-channel number*	Enter port-channel interface mode.
Step 9	**shutdown**	Shut down the interface.
Step 10	**no shutdown**	Enable the interface.
Step 11	**end**	Return to privileged EXEC mode.
Step 12	**show l2protocol**	Display the Layer 2 tunnel ports on the switch, including the protocols configured, the thresholds, and the counters.
Step 13	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no switchport mode trunk**, the **no udld enable**, and the **no channel group** *channel-group-number* **mode desirable** interface configuration commands to return the interface to the default settings.

For EtherChannels, you need to configure both the SP edge switches and the customer switches. Enable Layer 2 protocol tunneling on the SP edge switches as described in the following example. (See Figure 14-6 on page 14-9.)

This example shows how to configure the SP edge switches A and B. VLANs 17, 18, 19, and 20 are the SP access VLANs, Fast Ethernet interfaces 0/1 and 0/2 are point-to-point tunnel ports with PAgP and UDLD tunneling enabled, the drop threshold is 1000 pps, and Fast Ethernet interface 0/3 is a trunk port toward the other SP switch. Use the same configuration for SP edge switches C and D.

SP edge switch A configuration:

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# switchport access vlan 17
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface fastethernet0/2
Switch(config-if)# switchport access vlan 18
Switch(config-if)# switchport mode dot1q-tunnel
```

```
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface fastethernet0/3
Switch(config-if)# switchport trunk encapsulation isl
Switch(config-if)# switchport mode trunk
```

SP edge switch B configuration:

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# switchport access vlan 19
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface fastethernet0/2
Switch(config-if)# switchport access vlan 20
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface fastethernet0/3
Switch(config-if)# switchport trunk encapsulation isl
Switch(config-if)# switchport mode trunk
```

This example shows how to configure the customer switch at Site 1. Fast Ethernet interfaces 0/1, 0/2, 0/3, and 0/4 are set for IEEE 802.1Q trunking, UDLD is enabled, EtherChannel group 1 is enabled, and the port channel is shut down and then enabled to activate the EtherChannel configuration. Use the same configuration for the customer switch at Site 2.

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface fastethernet0/2
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface fastethernet0/3
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface fastethernet0/4
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface port-channel 1
Switch(config-if)# shutdown
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

Monitoring and Maintaining Tunneling Status

Table 14-2 shows the privileged EXEC commands for monitoring and maintaining IEEE 802.1Q and Layer 2 protocol tunneling.

Table 14-2 Commands for Monitoring and Maintaining Tunneling

Command	Purpose
clear l2protocol-tunnel counters	Clear the protocol counters on Layer 2 protocol tunneling ports.
show dot1q-tunnel	Display IEEE 802.1Q tunnel ports on the switch.
show dot1q-tunnel interface *interface-id*	Verify if a specific interface is a tunnel port.
show l2protocol-tunnel	Display information about Layer 2 protocol tunneling ports.
show errdisable recovery	Verify if the recovery timer from a Layer 2 protocol-tunnel error disable state is enabled.
show l2protocol-tunnel interface *interface-id*	Display information about a specific Layer 2 protocol tunneling port.
show l2protocol-tunnel summary	Display only Layer 2 protocol summary information.
show vlan dot1q native	Display the status of native VLAN tagging on the switch.

For detailed information about these displays, see the command reference for this release.

CHAPTER 15

Configuring STP

This chapter describes how to configure the Spanning Tree Protocol (STP) on port-based VLANs on your Catalyst 3550 switch. The switch can use either the per-VLAN spanning-tree plus (PVST+) protocol based on the IEEE 802.1D standard and Cisco proprietary extensions, or the rapid per-VLAN spanning-tree plus (rapid-PVST+) protocol based on the IEEE 802.1w standard.

For information about the Multiple Spanning Tree Protocol (MSTP) and how to map multiple VLANs to the same spanning-tree instance, see Chapter 16, "Configuring MSTP."

For information about other spanning-tree features such as Port Fast, UplinkFast, root guard, and so forth, see Chapter 17, "Configuring Optional Spanning-Tree Features."

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding Spanning-Tree Features, page 15-1
- Configuring Spanning-Tree Features, page 15-11
- Displaying the Spanning-Tree Status, page 15-24

Understanding Spanning-Tree Features

These sections describe how basic spanning-tree features work:

- STP Overview, page 15-2
- Spanning-Tree Topology and BPDUs, page 15-3
- Bridge ID, Switch Priority, and Extended System ID, page 15-4
- Spanning-Tree Interface States, page 15-4
- How a Switch or Port Becomes the Root Switch or Root Port, page 15-7
- Spanning Tree and Redundant Connectivity, page 15-8
- Spanning-Tree Address Management, page 15-8
- Accelerated Aging to Retain Connectivity, page 15-8
- Spanning-Tree Modes and Protocols, page 15-9
- Supported Spanning-Tree Instances, page 15-9

- Spanning-Tree Interoperability and Backward Compatibility, page 15-10
- STP and IEEE 802.1Q Trunks, page 15-10
- VLAN-Bridge Spanning Tree, page 15-10

For configuration information, see the "Configuring Spanning-Tree Features" section on page 15-11.

For information about optional spanning-tree features, see Chapter 17, "Configuring Optional Spanning-Tree Features."

STP Overview

STP is a Layer 2 link management protocol that provides path redundancy while preventing loops in the network. For a Layer 2 Ethernet network to function properly, only one active path can exist between any two stations. Multiple active paths among end stations cause loops in the network. If a loop exists in the network, end stations might receive duplicate messages. Switches might also learn end-station MAC addresses on multiple Layer 2 interfaces. These conditions result in an unstable network. Spanning-tree operation is transparent to end stations, which cannot detect whether they are connected to a single LAN segment or a switched LAN of multiple segments.

The STP uses a spanning-tree algorithm to select one switch of a redundantly connected network as the root of the spanning tree. The algorithm calculates the best loop-free path through a switched Layer 2 network by assigning a role to each port based on the role of the port in the active topology:

- Root—A forwarding port elected for the spanning-tree topology
- Designated—A forwarding port elected for every switched LAN segment
- Alternate—A blocked port providing an alternate path to the root bridge in the spanning tree
- Backup—A blocked port in a loopback configuration

The switch that has *all* of its ports as the designated role or as the backup role is the root switch. The switch that has at least *one* of its ports in the designated role is called the designated switch.

Spanning tree forces redundant data paths into a standby (blocked) state. If a network segment in the spanning tree fails and a redundant path exists, the spanning-tree algorithm recalculates the spanning-tree topology and activates the standby path. Switches send and receive spanning-tree frames, called bridge protocol data units (BPDUs), at regular intervals. The switches do not forward these frames but use them to construct a loop-free path. BPDUs contain information about the sending switch and its ports, including switch and MAC addresses, switch priority, port priority, and path cost. Spanning tree uses this information to elect the root switch and root port for the switched network and the root port and designated port for each switched segment.

When two interfaces on a switch are part of a loop, the spanning-tree port priority and path cost settings determine which interface is put in the forwarding state and which is put in the blocking state. The spanning-tree port priority value represents the location of an interface in the network topology and how well it is located to pass traffic. The path cost value represents the media speed.

Spanning-Tree Topology and BPDUs

The stable, active spanning-tree topology of a switched network is determined by these elements:

- The unique bridge ID (switch priority and MAC address) associated with each VLAN on each switch
- The spanning-tree path cost to the root switch
- The port identifier (port priority and MAC address) associated with each Layer 2 interface

When the switches in a network are powered up, each functions as the root switch. Each switch sends a configuration BPDU through all of its ports. The BPDUs communicate and compute the spanning-tree topology. Each configuration BPDU contains this information:

- The unique bridge ID of the switch that the sending switch identifies as the root switch
- The spanning-tree path cost to the root
- The bridge ID of the sending switch
- Message age
- The identifier of the sending interface
- Values for the hello, forward-delay, and max-age protocol timers

When a switch receives a configuration BPDU that contains *superior* information (lower bridge ID, lower path cost, and so forth), it stores the information for that port. If this BPDU is received on the root port of the switch, the switch also forwards it with an updated message to all attached LANs for which it is the designated switch.

If a switch receives a configuration BPDU that contains *inferior* information to that currently stored for that port, it discards the BPDU. If the switch is a designated switch for the LAN from which the inferior BPDU was received, it sends that LAN a BPDU containing the up-to-date information stored for that port. In this way, inferior information is discarded, and superior information is propagated on the network.

A BPDU exchange results in these actions:

- One switch in the network is elected as the root switch (the logical center of the spanning-tree topology in a switched network).

  For each VLAN, the switch with the highest switch priority (the lowest numerical priority value) is elected as the root switch. If all switches are configured with the default priority (32768), the switch with the lowest MAC address in the VLAN becomes the root switch. The switch priority value occupies the most significant bits of the bridge ID, as shown in Table 15-1 on page 15-4.
- A root port is selected for each switch (except the root switch). This port provides the best path (lowest cost) when the switch forwards packets to the root switch.
- The shortest distance to the root switch is calculated for each switch based on the path cost.
- A designated switch for each LAN segment is selected. The designated switch incurs the lowest path cost when forwarding packets from that LAN to the root switch. The port through which the designated switch is attached to the LAN is called the designated port.
- Interfaces included in the spanning-tree instance are selected. Root ports and designated ports are put in the forwarding state.
- All paths that are not needed to reach the root switch from anywhere in the switched network are placed in the spanning-tree blocking mode.

Bridge ID, Switch Priority, and Extended System ID

The IEEE 802.1D standard requires that each switch has an unique bridge identifier (bridge ID), which determines the selection of the root switch. Because each VLAN is considered as a different *logical bridge* with PVST+ and rapid PVST+, the same switch must have as many different bridge IDs as VLANs configured on it. Each VLAN on the switch has a unique 8-byte bridge ID; the two most-significant bytes are used for the switch priority, and the remaining six bytes are derived from the switch MAC address.

In Cisco IOS Release 12.1(8)EA1 and later, the switch supports the 802.1t spanning-tree extensions. Some of the bits previously used for the switch priority are now used as the VLAN identifier. The result is that fewer MAC addresses are reserved for the switch, and a larger range of VLAN IDs can be supported, all while maintaining the uniqueness of the bridge ID. As shown in Table 15-1, the two bytes previously used for the switch priority are reallocated into a 4-bit priority value and a 12-bit extended system ID value equal to the VLAN ID. In earlier releases, the switch priority is a 16-bit value.

Table 15-1 Switch Priority Value and Extended System ID

Switch Priority Value				Extended System ID (Set Equal to the VLAN ID)											
Bit 16	**Bit 15**	**Bit 14**	**Bit 13**	**Bit 12**	**Bit 11**	**Bit 10**	**Bit 9**	**Bit 8**	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**
32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

Spanning tree uses the extended system ID, the switch priority, and the allocated spanning-tree MAC address to make the bridge ID unique for each VLAN. With earlier releases, spanning tree used one MAC address per VLAN to make the bridge ID unique for each VLAN.

Support for the extended system ID affects how you manually configure the root switch, the secondary root switch, and the switch priority of a VLAN. For example, when you change the switch priority value, you change the probability that the switch will be elected as the root switch. Configuring a higher value decreases the probability; a lower value increases the probability. For more information, see the "Configuring the Root Switch" section on page 15-15, the "Configuring a Secondary Root Switch" section on page 15-16, and the "Configuring the Switch Priority of a VLAN" section on page 15-19.

Spanning-Tree Interface States

Propagation delays can occur when protocol information passes through a switched LAN. As a result, topology changes can take place at different times and at different places in a switched network. When an interface transitions directly from nonparticipation in the spanning-tree topology to the forwarding state, it can create temporary data loops. Interfaces must wait for new topology information to propagate through the switched LAN before starting to forward frames. They must allow the frame lifetime to expire for forwarded frames that have used the old topology.

Each Layer 2 interface on a switch using spanning tree exists in one of these states:

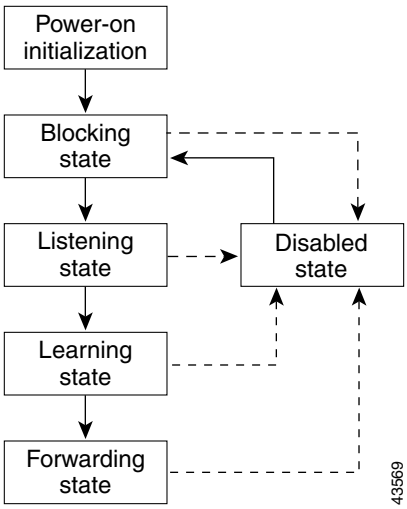
- Blocking—The interface does not participate in frame forwarding.
- Listening—The first transitional state after the blocking state when the spanning tree determines that the interface should participate in frame forwarding.
- Learning—The interface prepares to participate in frame forwarding.
- Forwarding—The interface forwards frames.
- Disabled—The interface is not participating in spanning tree because of a shutdown port, no link on the port, or no spanning-tree instance running on the port.

An interface moves through these states:

- From initialization to blocking
- From blocking to listening or to disabled
- From listening to learning or to disabled
- From learning to forwarding or to disabled
- From forwarding to disabled

Figure 15-1 illustrates how an interface moves through the states.

Figure 15-1 Spanning-Tree Interface States



When you power up the switch, spanning tree is enabled by default, and every interface in the switch, VLAN, or network goes through the blocking state and the transitory states of listening and learning. Spanning tree stabilizes each interface at the forwarding or blocking state.

When the spanning-tree algorithm places a Layer 2 interface in the forwarding state, this process occurs:

1. The interface is in the listening state while spanning tree waits for protocol information to transition the interface to the blocking state.
2. While the spanning tree waits for the forward-delay timer to expire, it moves the interface to the learning state and resets the forward-delay timer.
3. In the learning state, the interface continues to block frame forwarding as the switch learns end-station location information for the forwarding database.
4. When the forward-delay timer expires, spanning tree moves the interface to the forwarding state, where both learning and frame forwarding are enabled.

Blocking State

A Layer 2 interface in the blocking state does not participate in frame forwarding. After initialization, a BPDU is sent to each interface in the switch. A switch initially functions as the root until it exchanges BPDUs with other switches. This exchange establishes which switch in the network is the root or root

switch. If there is only one switch in the network, no exchange occurs, the forward-delay timer expires, and the interfaces move to the listening state. An interface always enters the blocking state after switch initialization.

An interface in the blocking state performs as follows:

- Discards frames received on the port
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Receives BPDUs

Listening State

The listening state is the first state a Layer 2 interface enters after the blocking state. The interface enters this state when the spanning tree determines that the interface should participate in frame forwarding.

An interface in the listening state performs as follows:

- Discards frames received on the port
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Receives BPDUs

Learning State

A Layer 2 interface in the learning state prepares to participate in frame forwarding. The interface enters the learning state from the listening state.

An interface in the learning state performs as follows:

- Discards frames received on the port
- Discards frames switched from another interface for forwarding
- Learns addresses
- Receives BPDUs

Forwarding State

A Layer 2 interface in the forwarding state forwards frames. The interface enters the forwarding state from the learning state.

An interface in the forwarding state performs as follows:

- Receives and forwards frames received on the port
- Forwards frames switched from another port
- Learns addresses
- Receives BPDUs

Disabled State

A Layer 2 interface in the disabled state does not participate in frame forwarding or in the spanning tree. An interface in the disabled state is nonoperational.

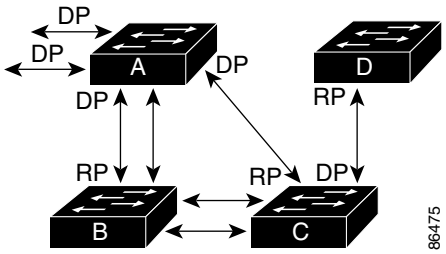
A disabled interface performs as follows:

- Discards frames received on the port
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Does not receive BPDUs

How a Switch or Port Becomes the Root Switch or Root Port

If all switches in a network are enabled with default spanning-tree settings, the switch with the lowest MAC address becomes the root switch. In Figure 15-2, Switch A is elected as the root switch because the switch priority of all the switches is set to the default (32768) and Switch A has the lowest MAC address. However, because of traffic patterns, number of forwarding interfaces, or link types, Switch A might not be the ideal root switch. By increasing the priority (lowering the numerical value) of the ideal switch so that it becomes the root switch, you force a spanning-tree recalculation to form a new topology with the ideal switch as the root.

Figure 15-2 Spanning-Tree Topology



RP = Root Port
DP = Designated Port

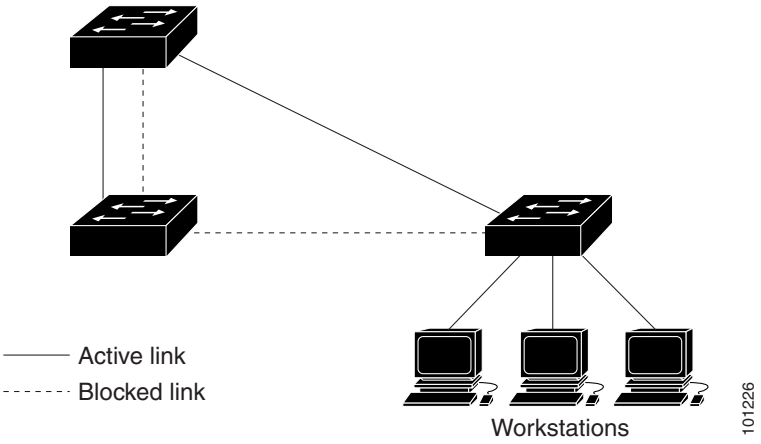
When the spanning-tree topology is calculated based on default parameters, the path between source and destination end stations in a switched network might not be ideal. For instance, connecting higher-speed links to an interface that has a higher number than the root port can cause a root-port change. The goal is to make the fastest link the root port.

For example, assume that one port on Switch B is a Gigabit Ethernet link and that another port on Switch B (a 10/100 Mbps link) is the root port. Network traffic might be more efficient over the Gigabit Ethernet link. By changing the spanning-tree port priority on the Gigabit Ethernet interface to a higher priority (lower numerical value) than the root port, the Gigabit Ethernet interface becomes the new root port.

Spanning Tree and Redundant Connectivity

You can create a redundant backbone with spanning tree by connecting two switch interfaces to another device or to two different devices. Spanning tree automatically disables one interface but enables it if the other one fails, as shown in Figure 15-3. If one link is high-speed and the other is low-speed, the low-speed link is always disabled. If the speeds are the same, the port priority and port ID are added together, and spanning tree disables the link with the lowest value.

Figure 15-3 Spanning Tree and Redundant Connectivity



You can also create redundant links between switches by using EtherChannel groups. For more information, see Chapter 30, "Configuring EtherChannels."

Spanning-Tree Address Management

IEEE 802.1D specifies 17 multicast addresses, ranging from 0x00180C2000000 to 0x0180C2000010, to be used by different bridge protocols. These addresses are static addresses that cannot be removed.

Regardless of the spanning-tree state, the switch receives but does not forward packets destined for addresses between 0x0180C2000000 and 0x0180C200000F.

If spanning tree is enabled, the switch CPU receives packets destined for 0x0180C2000000 and 0x0180C2000010. If spanning-tree is disabled, the switch forwards those packets as unknown multicast addresses.

Accelerated Aging to Retain Connectivity

The default for aging dynamic addresses is 5 minutes, the default setting of the **mac-address-table aging-time** global configuration command. However, a spanning-tree reconfiguration can cause many station locations to change. Because these stations could be unreachable for 5 minutes or more during a reconfiguration, the address-aging time is accelerated so that station addresses can be dropped from the address table and then relearned. The accelerated aging is the same as the forward-delay parameter value (**spanning-tree vlan** *vlan-id* **forward-time** *seconds* global configuration command) when the spanning tree reconfigures.

Because each VLAN is a separate spanning-tree instance, the switch accelerates aging on a per-VLAN basis. A spanning-tree reconfiguration on one VLAN can cause the dynamic addresses learned on that VLAN to be subject to accelerated aging. Dynamic addresses on other VLANs can be unaffected and remain subject to the aging interval entered for the switch.

Spanning-Tree Modes and Protocols

The switch supports these spanning-tree modes and protocols:

PVST+—This spanning-tree mode is based on the IEEE 802.1D standard and Cisco proprietary extensions. It is the default spanning-tree mode used on all Ethernet, Fast Ethernet, and Gigabit Ethernet port-based VLANs. The PVST+ runs on each VLAN on the switch up to the maximum supported, ensuring that each has a loop-free path through the network.

The PVST+ provides Layer 2 load balancing for the VLAN on which it runs. You can create different logical topologies by using the VLANs on your network to ensure that all of your links are used but that no one link is oversubscribed. Each instance of PVST+ on a VLAN has a single root switch. This root switch propagates the spanning-tree information associated with that VLAN to all other switches in the network. Because each switch has the same information about the network, this process ensures that the network topology is maintained.

- Rapid PVST+—This spanning-tree mode is the same as PVST+ except that is uses a rapid convergence based on the IEEE 802.1w standard. To provide rapid convergence, the rapid PVST+ immediately deletes dynamically learned MAC address entries on a per-port basis upon receiving a topology change. By contrast, PVST+ uses a short aging time for dynamically learned MAC address entries.

  The rapid PVST+ uses the same configuration as PVST+ (except where noted), and the switch needs only minimal extra configuration. The benefit of rapid PVST+ is that you can migrate a large PVST+ install base to rapid PVST+ without having to learn the complexities of the MSTP configuration and without having to reprovision your network. In rapid-PVST+ mode, each VLAN runs its own spanning-tree instance up to the maximum supported.

- MSTP—This spanning-tree mode is based on the IEEE 802.1s standard. You can map multiple VLANs to the same spanning-tree instance, which reduces the number of spanning-tree instances required to support a large number of VLANs. The MSTP runs on top of the RSTP (based on IEEE 802.1w, which provides for rapid convergence of the spanning tree by eliminating the forward delay and by quickly transitioning root ports and designated ports to the forwarding state. You cannot run MSTP without RSTP.

  The most common initial deployment of MSTP is in the backbone and distribution layers of a Layer 2 switched network. For more information, see Chapter 16, “Configuring MSTP.” For information about the number of supported spanning-tree instances, see the next section.

Supported Spanning-Tree Instances

In PVST+ or rapid-PVST+ mode, the switch supports up to 128 spanning-tree instances.

In MSTP mode, the switch supports up to 65 MST instances. The number of VLANs that can be mapped to a particular MST instance is unlimited.

For information about how spanning tree interoperates with the VLAN Trunking Protocol (VTP), see the “Spanning-Tree Configuration Guidelines” section on page 15-12.

Spanning-Tree Interoperability and Backward Compatibility

Table 15-2 lists the interoperability and compatibility among the supported spanning-tree modes in a network.

Table 15-2 PVST+, MSTP, and Rapid-PVST+ Interoperability

	PVST+	MSTP	Rapid PVST+
PVST+	Yes	Yes (with restrictions)	Yes (reverts to PVST+)
MSTP	Yes (with restrictions)	Yes	Yes (reverts to PVST+)
Rapid PVST+	Yes (reverts to PVST+)	Yes (reverts to PVST+)	Yes

In a mixed MSTP and PVST+ network, the common spanning-tree (CST) root must be inside the MST backbone, and a PVST+ switch cannot connect to multiple MST regions.

When a network contains switches running rapid PVST+ and switches running PVST+, we recommend that the rapid-PVST+ switches and PVST+ switches be configured for different spanning-tree instances. In the rapid-PVST+ spanning-tree instances, the root switch must be a rapid-PVST+ switch. In the PVST+ instances, the root switch must be a PVST+ switch. The PVST+ switches should be at the edge of the network.

STP and IEEE 802.1Q Trunks

The IEEE 802.1Q standard for VLAN trunks imposes some limitations on the spanning-tree strategy for a network. The standard requires only one spanning-tree instance for *all* VLANs allowed on the trunks. However, in a network of Cisco switches connected through IEEE 802.1Q trunks, the switches maintain one spanning-tree instance for *each* VLAN allowed on the trunks.

When you connect a Cisco switch to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco switch uses PVST+ to provide spanning-tree interoperability. If rapid PVST+ is enabled, the switch uses it instead of PVST+. The switch combines the spanning-tree instance of the IEEE 802.1Q VLAN of the trunk with the spanning-tree instance of the non-Cisco IEEE 802.1Q switch.

However, all PVST+ or rapid-PVST+ information is maintained by Cisco switches separated by a cloud of non-Cisco IEEE 802.1Q switches. The non-Cisco IEEE 802.1Q cloud separating the Cisco switches is treated as a single trunk link between the switches.

PVST+ is automatically enabled on 802.1Q trunks, and no user configuration is required. The external spanning-tree behavior on access ports and Inter-Switch Link (ISL) trunk ports is not affected by PVST+.

For more information on IEEE 802.1Q trunks, see Chapter 11, “Configuring VLANs.”

VLAN-Bridge Spanning Tree

Cisco VLAN-bridge spanning tree is used with the fallback bridging feature (bridge groups), which forwards non-IP protocols such as DECnet between two or more VLAN bridge domains or routed ports. The VLAN-bridge spanning tree allows the bridge groups to form a spanning tree on top of the individual VLAN spanning trees to prevent loops from forming if there are multiple connections among VLANs. It also prevents the individual spanning trees from the VLANs being bridged from collapsing into a single spanning tree.

To support VLAN-bridge spanning tree, some of the spanning-tree timers are increased. To use the fallback bridging feature, you must have the enhanced multilayer software image installed on your switch. For more information, see Chapter 5, "Configuring Fallback Bridging."

Configuring Spanning-Tree Features

These sections describe how to configure spanning-tree features:

- Default Spanning-Tree Configuration, page 15-11
- Spanning-Tree Configuration Guidelines, page 15-12
- Changing the Spanning-Tree Mode, page 15-13 (required)
- Disabling Spanning Tree, page 15-14 (optional)
- Configuring the Root Switch, page 15-15 (optional)
- Configuring a Secondary Root Switch, page 15-16 (optional)
- Configuring the Port Priority, page 15-17 (optional)
- Configuring the Path Cost, page 15-18 (optional)
- Configuring the Switch Priority of a VLAN, page 15-19 (optional)
- Configuring Spanning-Tree Timers, page 15-20 (optional)

Default Spanning-Tree Configuration

Table 15-3 shows the default spanning-tree configuration.

Table 15-3 Default Spanning-Tree Configuration

Feature	Default Setting
Enable state	Enabled on VLAN 1. For more information, see the "Supported Spanning-Tree Instances" section on page 15-9.
Spanning-tree mode	PVST+. (Rapid PVST+ and MSTP are disabled.)
Switch priority	32768.
Spanning-tree port priority (configurable on a per-interface basis)	128.
Spanning-tree port cost (configurable on a per-interface basis)	1000 Mbps: 4. 100 Mbps: 19. 10 Mbps: 100.
Spanning-tree VLAN port priority (configurable on a per-VLAN basis)	128.

Table 15-3 Default Spanning-Tree Configuration (continued)

Feature	Default Setting
Spanning-tree VLAN port cost (configurable on a per-VLAN basis)	1000 Mbps: 4. 100 Mbps: 19. 10 Mbps: 100.
Spanning-tree timers	Hello time: 2 seconds. Forward-delay time: 15 seconds. Maximum-aging time: 20 seconds. Transmit hold count: 6 BPDUs

Spanning-Tree Configuration Guidelines

If more VLANs are defined in the VTP than there are spanning-tree instances, you can enable PVST+ or rapid PVST+ on only 128 VLANs. The remaining VLANs operate with spanning tree disabled. If the number of VLANs exceeds 128, we recommend that you enable the MSTP to map multiple VLANs to a single spanning-tree instance. For more information, see Chapter 16, "Configuring MSTP."

For information on the recommended trunk port configuration, see the"Interaction with Other Features" section on page 11-19.

If 128 instances of spanning tree are already in use, you can disable spanning tree on one of the VLANs and then enable it on the VLAN where you want it to run. Use the **no spanning-tree vlan** *vlan-id* global configuration command to disable spanning tree on a specific VLAN, and use the **spanning-tree vlan** *vlan-id* global configuration command to enable spanning tree on the desired VLAN.

Caution Switches that are not running spanning tree still forward received BPDUs so that the other switches on the VLAN with a running spanning-tree instance can break loops. Therefore, spanning tree must be running on enough switches to break all the loops in the network. For example, at least one switch on each loop in the VLAN must be running spanning tree. It is not absolutely necessary to run spanning tree on all switches in the VLAN; however, if you are running spanning tree only on a minimal set of switches, an incautious change to the network that introduces another loop into the VLAN can result in a broadcast storm.

Note If you have already used all available spanning-tree instances on your switch, adding another VLAN anywhere in the VTP domain creates a VLAN that is not running spanning tree on that switch. If you have the default allowed list on the trunk ports of that switch, the new VLAN is carried on all trunk ports. Depending on the topology of the network, this could create a loop in the new VLAN that will not be broken, particularly if there are several adjacent switches that have all run out of spanning-tree instances. You can prevent this possibility by setting up allowed lists on the trunk ports of switches that have used up their allocation of spanning-tree instances. Setting up allowed lists is not necessary in many cases and can make it more labor-intensive to add another VLAN to the network.

Spanning-tree commands determine the configuration of VLAN spanning-tree instances. You create a spanning-tree instance when you assign an interface to a VLAN. The spanning-tree instance is removed when the last interface is moved to another VLAN. You can configure switch and port parameters before a spanning-tree instance is created; these parameters are applied when the spanning-tree instance is created.

The switch supports PVST+, rapid PVST+, and MSTP, but only one version can be active at any time. (For example, all VLANs run PVST+, all VLANs run rapid PVST+, or all VLANs run MSTP.) For information about the different spanning-tree modes and how they interoperate, see the "Spanning-Tree Interoperability and Backward Compatibility" section on page 15-10.

For configuration guidelines about UplinkFast, BackboneFast, and cross-stack UplinkFast, see the "Optional Spanning-Tree Configuration Guidelines" section on page 17-14.

Caution Loop guard works only on point-to-point links. We recommend that each end of the link has a directly connected device that is running STP.

Changing the Spanning-Tree Mode

The switch supports three spanning-tree modes: PVST+, rapid PVST+, or MSTP. By default, the switch runs the PVST+ protocol.

Beginning in privileged EXEC mode, follow these steps to change the spanning-tree mode. If you want to enable a mode that is different from the default mode, this procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mode** {**pvst** | **mst** | **rapid-pvst**}	Configure a spanning-tree mode. • Select **pvst** to enable PVST+ (the default setting). • Select **mst** to enable MSTP (and RSTP). For more configuration steps, see Chapter 16, "Configuring MSTP." • Select **rapid-pvst** to enable rapid PVST+.
Step 3	**interface** *interface-id*	(Recommended for rapid-PVST+ mode only) Specify an interface to configure, and enter interface configuration mode. Valid interfaces include physical ports, VLANs, and port channels. Valid VLAN IDs are 1 to 4094. The port-channel range is 1 to 64.
Step 4	**spanning-tree link-type point-to-point**	(Recommended for rapid-PVST+ mode only) Specify that the link type for this port is point-to-point. If you connect this port (local port) to a remote port through a point-to-point link and the local port becomes a designated port, the switch negotiates with the remote port and rapidly transitions the local port to the forwarding state.
Step 5	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 6	**clear spanning-tree detected-protocols**	(Recommended for rapid-PVST+ mode only) If any port on the switch is connected to a port on a legacy IEEE 802.1D switch, restart the protocol migration process on the entire switch. This step is optional if the designated switch determines that this switch is running rapid PVST+.
Step 7	**show spanning-tree summary** and **show spanning-tree interface** *interface-id*	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no spanning-tree mode** global configuration command. To return the port to its default setting, use the **no spanning-tree link-type** interface configuration command.

Disabling Spanning Tree

Spanning tree is enabled by default on VLAN 1 and on all newly created VLANs up to the spanning-tree limit specified in the "Supported Spanning-Tree Instances" section on page 15-9. Disable spanning tree only if you are sure there are no loops in the network topology.

Caution When spanning tree is disabled and loops are present in the topology, excessive traffic and indefinite packet duplication can drastically reduce network performance.

Beginning in privileged EXEC mode, follow these steps to disable spanning tree on a per-VLAN basis. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no spanning-tree vlan** *vlan-id*	Disable spanning tree on a per-VLAN basis. For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To re-enable spanning tree, use the **spanning-tree vlan** *vlan-id* global configuration command.

Configuring the Root Switch

The switch maintains a separate spanning-tree instance for each active VLAN configured on it. A bridge ID, consisting of the switch priority and the switch MAC address, is associated with each instance. For each VLAN, the switch with the lowest bridge ID becomes the root switch for that VLAN.

To configure a switch to become the root for the specified VLAN, use the **spanning-tree vlan** *vlan-id* **root** global configuration command to modify the switch priority from the default value (32768) to a significantly lower value. When you enter this command, the switch checks the switch priority of the root switches for each VLAN. Because of the extended system ID support, the switch sets its own priority for the specified VLAN to 24576 if this value will cause this switch to become the root for the specified VLAN.

If any root switch for the specified VLAN has a switch priority lower than 24576, the switch sets its own priority for the specified VLAN to 4096 less than the lowest switch priority. (4096 is the value of the least-significant bit of a 4-bit switch priority value as shown in Table 15-1 on page 15-4.)

Note The **spanning-tree vlan** *vlan-id* **root** global configuration command fails if the value necessary to be the root switch is less than 1.

Before Cisco IOS Release 12.1(8)EA1, entering the **spanning-tree vlan** *vlan-id* **root** global configuration command on a Catalyst 3550 switch (no extended system ID) caused it to set its own switch priority for the specified VLAN to 8192 if this value caused this switch to become the root for the specified VLAN. If any root switch for the specified VLAN has a switch priority lower than 8192, the switch sets its own priority for the specified VLAN to 1 less than the lowest switch priority.

These examples show the effect of the **spanning-tree vlan** *vlan-id* **root** command with and without the extended system ID support:

- For Catalyst 3550 switches with the extended system ID (Cisco IOS Release 12.1(8)EA1 and later), if all network devices in VLAN 20 have the default priority of 32768, entering the **spanning-tree vlan 20 root primary** command on the switch sets the switch priority to 24576, which causes this switch to become the root switch for VLAN 20.
- For Catalyst 3550 switches without the extended system ID (software earlier than Cisco IOS Release 12.1(8)EA1), if all network devices in VLAN 100 have the default priority of 32768, entering the **spanning-tree vlan 100 root primary** command on the switch sets the switch priority for VLAN 100 to 8192, which causes this switch to become the root switch for VLAN 100.

Note If your network consists of switches that both do and do not support the extended system ID, it is unlikely that the switch with the extended system ID support will become the root switch. The extended system ID increases the switch priority value every time the VLAN number is greater than the priority of the connected switches running older software.

Note The root switch for each spanning-tree instance should be a backbone or distribution switch. Do not configure an access switch as the spanning-tree primary root.

Use the **diameter** keyword to specify the Layer 2 network diameter (that is, the maximum number of switch hops between any two end stations in the Layer 2 network). When you specify the network diameter, the switch automatically sets an optimal hello time, forward-delay time, and maximum-age time for a network of that diameter, which can significantly reduce the convergence time. You can use the **hello** keyword to override the automatically calculated hello time.

Note After configuring the switch as the root switch, we recommend that you avoid manually configuring the hello time, forward-delay time, and maximum-age time by using the **spanning-tree vlan** *vlan-id* **hello-time**, **spanning-tree vlan** *vlan-id* **forward-time**, and the **spanning-tree vlan** *vlan-id* **max-age** global configuration commands.

Beginning in privileged EXEC mode, follow these steps to configure a switch to become the root for the specified VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **root primary** [**diameter** *net-diameter* [**hello-time** *seconds*]]	Configure a switch to become the root for the specified VLAN. For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • (Optional) For **diameter** *net-diameter*, specify the maximum number of switches between any two end stations. The range is 2 to 7. • (Optional) For **hello-time** *seconds*, specify the interval in seconds between the generation of configuration messages by the root switch. The range is 1 to 10 seconds; the default is 2 seconds. **Note** When you enter this command without the optional keywords, the switch recalculates the forward-time, hello-time, max-age, and priority settings. If you had previously configured these parameters, the switch recalculates them.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree detail**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree vlan** *vlan-id* **root** global configuration command.

Configuring a Secondary Root Switch

When you configure a Catalyst 3550 switch that supports the extended system ID as the secondary root, the switch priority is modified from the default value (32768) to 28672. The switch is then likely to become the root switch for the specified VLAN if the primary root switch fails. This is assuming that the other network switches use the default switch priority of 32768 and therefore are unlikely to become the root switch. For Catalyst 3550 switches without the extended system ID support (software earlier than Cisco IOS Release 12.1(8)EA1), the switch priority is changed to 16384.

You can execute this command on more than one switch to configure multiple backup root switches. Use the same network diameter and hello-time values as you used when you configured the primary root switch with the **spanning-tree vlan** *vlan-id* **root primary** global configuration command.

Beginning in privileged EXEC mode, follow these steps to configure a switch to become the secondary root for the specified VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **root secondary** [**diameter** *net-diameter* [**hello-time** *seconds*]]	Configure a switch to become the secondary root for the specified VLAN. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • (Optional) For **diameter** *net-diameter*, specify the maximum number of switches between any two end stations. The range is 2 to 7. • (Optional) For **hello-time** *seconds*, specify the interval in seconds between the generation of configuration messages by the root switch. The range is 1 to 10 seconds; the default is 2 seconds. Use the same network diameter and hello-time values that you used when configuring the primary root switch. See the "Configuring the Root Switch" section on page 15-15.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree detail**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree vlan** *vlan-id* **root** global configuration command.

Configuring the Port Priority

If a loop occurs, spanning tree uses the port priority when selecting an interface to put into the forwarding state. You can assign higher priority values (lower numerical values) to interfaces that you want selected first and lower priority values (higher numerical values) that you want selected last. If all interfaces have the same priority value, spanning tree puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Beginning in privileged EXEC mode, follow these steps to configure the port priority of an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. Valid interfaces include physical interfaces and port-channel logical interfaces (**port-channel** *port-channel-number*).

	Command	Purpose
Step 3	**spanning-tree port-priority** *priority*	Configure the port priority for an interface. For *priority*, the range is 0 to 240 in increments of 16. The default is 128. The lower the number, the higher the priority. Valid priority values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected.
Step 4	**spanning-tree vlan** *vlan-id* **port-priority** *priority*	Configure the VLAN port priority for an interface. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. For *priority*, the range is 0 to 240 in increments of 16. The default is 128. The lower the number, the higher the priority. Valid priority values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show spanning-tree interface** *interface-id* or **show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note The **show spanning-tree interface** *interface-id* privileged EXEC command displays information only if the port is in a link-up operative state. Otherwise, you can use the **show running-config interface** privileged EXEC command to confirm the configuration.

To return the interface to its default setting, use the **no spanning-tree** [**vlan** *vlan-id*] **port-priority** interface configuration command. For information on how to configure load sharing on trunk ports by using spanning-tree port priorities, see the "Load Sharing Using STP" section on page 11-23.

Configuring the Path Cost

The spanning-tree path cost default value is derived from the media speed of an interface. If a loop occurs, spanning tree uses cost when selecting an interface to put in the forwarding state. You can assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last. If all interfaces have the same cost value, spanning tree puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Beginning in privileged EXEC mode, follow these steps to configure the cost of an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. Valid interfaces include physical interfaces and port-channel logical interfaces (**port-channel** *port-channel-number*).
Step 3	**spanning-tree cost** *cost*	Configure the cost for an interface. If a loop occurs, spanning tree uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission. For *cost*, the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 4	**spanning-tree vlan** *vlan-id* **cost** *cost*	Configure the cost for a VLAN. If a loop occurs, spanning tree uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *cost*, the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show spanning-tree interface** *interface-id* or **show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note The **show spanning-tree interface** *interface-id* privileged EXEC command displays information only for ports that are in a link-up operative state. Otherwise, you can use the **show running-config** privileged EXEC command to confirm the configuration.

To return the interface to its default setting, use the **no spanning-tree** [**vlan** *vlan-id*] **cost** interface configuration command. For information on how to configure load sharing on trunk ports by using spanning-tree path costs, see the "Load Sharing Using STP" section on page 11-23.

Configuring the Switch Priority of a VLAN

You can configure the switch priority and make it more likely that the switch will be chosen as the root switch.

Note Exercise care when using this command. For most situations, we recommend that you use the **spanning-tree vlan** *vlan-id* **root primary** and the **spanning-tree vlan** *vlan-id* **root secondary** global configuration commands to modify the switch priority.

Beginning in privileged EXEC mode, follow these steps to configure the switch priority of a VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **priority** *priority*	Configure the switch priority of a VLAN. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *priority*, the range is 0 to 61440 in increments of 4096; the default is 32768. The lower the number, the more likely the switch will be chosen as the root switch. Valid priority values are 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. All other values are rejected.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree vlan** *vlan-id* **priority** global configuration command.

Configuring Spanning-Tree Timers

Table 15-4 describes the timers that affect the entire spanning-tree performance.

Table 15-4 Spanning-Tree Timers

Variable	Description
Hello timer	Determines how often the switch broadcasts hello messages to other switches.
Forward-delay timer	Determines how long each of the listening and learning states last before the interface begins forwarding.
Maximum-age timer	Determines the amount of time the switch stores protocol information received on an interface.
Transmit hold count	Controls the number of BPDUs sent every second.

The sections that follow provide the configuration steps.

Configuring the Hello Time

You can configure the interval between the generation of configuration messages by the root switch by changing the hello time.

Note Exercise care when using this command. For most situations, we recommend that you use the **spanning-tree vlan** *vlan-id* **root primary** and the **spanning-tree vlan** *vlan-id* **root secondary** global configuration commands to modify the hello time.

Beginning in privileged EXEC mode, follow these steps to configure the hello time of a VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **hello-time** *seconds*	Configure the hello time of a VLAN. The hello time is the interval between the generation of configuration messages by the root switch. These messages mean that the switch is alive. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *seconds*, the range is 1 to 10; the default is 2.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree vlan** *vlan-id* **hello-time** global configuration command.

Configuring the Forwarding-Delay Time for a VLAN

Beginning in privileged EXEC mode, follow these steps to configure the forwarding-delay time for a VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **forward-time** *seconds*	Configure the forward time of a VLAN. The forward delay is the number of seconds a port waits before changing from its spanning-tree learning and listening states to the forwarding state. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *seconds*, the range is 4 to 30; the default is 15.
Step 3	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 4	**show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree vlan** *vlan-id* **forward-time** global configuration command.

Configuring the Maximum-Aging Time for a VLAN

Beginning in privileged EXEC mode, follow these steps to configure the maximum-aging time for a VLAN. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree vlan** *vlan-id* **max-age** *seconds*	Configure the maximum-aging time of a VLAN. The maximum-aging time is the number of seconds a switch waits without receiving spanning-tree configuration messages before attempting a reconfiguration. • For *vlan-id*, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For *seconds*, the range is 6 to 40; the default is 20.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree vlan** *vlan-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree vlan** *vlan-id* **max-age** global configuration command.

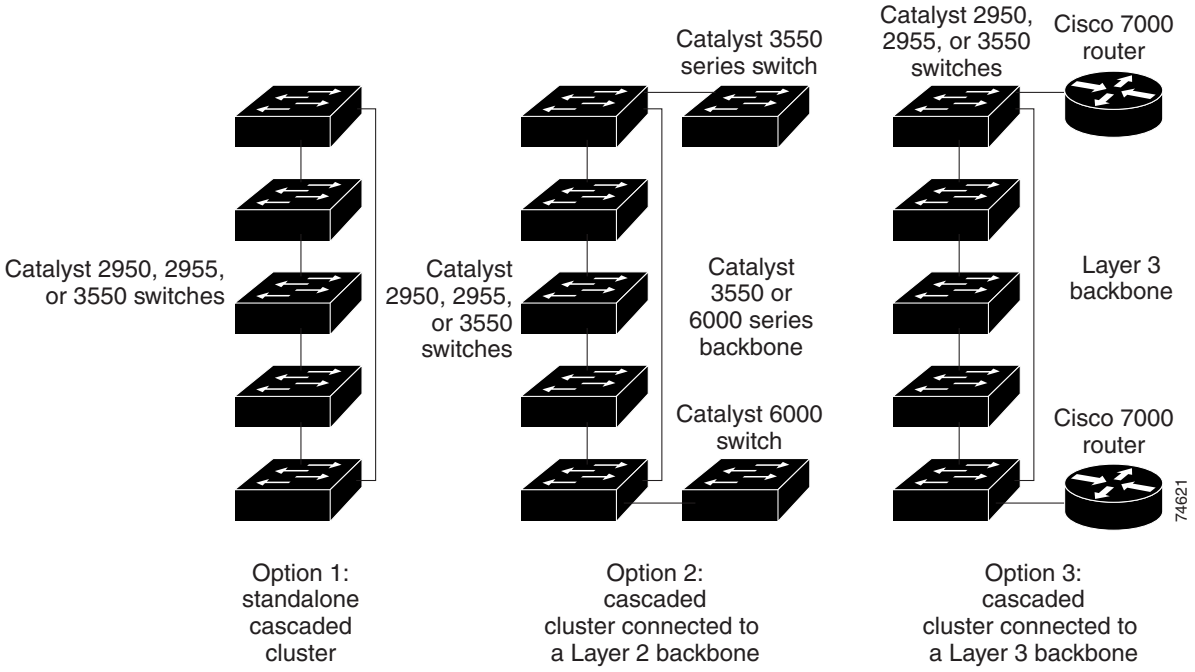
Configuring Spanning Tree for Use in a Cascaded Stack

Spanning tree uses default values that can be reduced when configuring your switch in cascaded configurations. If a root switch is part of a cluster that is one switch from a cascaded stack, you can customize spanning tree to reconverge more quickly after a switch failure. Figure 15-4 shows switches in three cascaded stacks that use the GigaStack GBIC. Table 15-5 shows the default spanning-tree settings and those that are acceptable for these configurations.

Table 15-5 Default and Acceptable Spanning-Tree Parameter Settings (in seconds)

STP Parameter	STP Default	Acceptable for Option 1	Acceptable for Option 2	Acceptable for Option 3
Hello Time	2	1	1	1
Max Age	20	6	10	6
Forwarding Delay	15	4	7	4

Figure 15-4 Gigabit Ethernet Stack



Configuring the Transmit Hold Count

You can configure the maximum number of BPDUs that can be sent in one second by changing the transmit hold-count value.

Note Increasing the transmit hold-count value can have a significant impact on CPU utilization in rapid-PVST+ mode. Decreasing this value might slow down convergence. We recommend that you maintain the default setting.

Beginning in privileged EXEC mode, follow these steps to configure the transmit hold count. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree transmit hold-count** *value*	Configure the number of BPDUs sent every second. For *value*, the range is 1 to 20; the default is 6.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree detail**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no spanning-tree transmit hold-count** *valu***e** global configuration command.

Displaying the Spanning-Tree Status

To display the spanning-tree status, use one or more of the privileged EXEC commands in Table 15-6:

Table 15-6 Commands for Displaying Spanning-Tree Status

Command	Purpose
show spanning-tree active	Displays spanning-tree information on active interfaces only.
show spanning-tree detail	Displays a detailed summary of interface information.
show spanning-tree interface *interface-id*	Displays spanning-tree information for the specified interface.
show spanning-tree summary [**totals**]	Displays a summary of port states or displays the total lines of the STP state section.

You can clear spanning-tree counters by using the **clear spanning-tree** [**interface** *interface-id*] privileged EXEC command.

For information about other keywords for the **show spanning-tree** privileged EXEC command, see the command reference for this release.

CHAPTER 16

Configuring MSTP

This chapter describes how to configure the Cisco implementation of the IEEE 802.1s Multiple STP (MSTP) on your Catalyst 3550 switch.

Note The multiple spanning-tree (MST) implementation in Cisco IOS Release 12.2(25)SEC is based on the IEEE 802.1s standard. The MST implementations in earlier Cisco IOS releases are prestandard.

The MSTP enables multiple VLANs to be mapped to the same spanning-tree instance, reducing the number of spanning-tree instances needed to support a large number of VLANs. The MSTP provides for multiple forwarding paths for data traffic and enables load balancing. It improves the fault tolerance of the network because a failure in one instance (forwarding path) does not affect other instances (forwarding paths). The most common initial deployment of MSTP is in the backbone and distribution layers of a Layer 2 switched network. This deployment provides the highly available network required in a service-provider environment.

When the switch is in the MST mode, the Rapid Spanning Tree Protocol (RSTP), which is based on IEEE 802.1w, is automatically enabled. The RSTP provides rapid convergence of the spanning tree through explicit handshaking that eliminates the IEEE 802.1D forwarding delay and quickly transitions root ports and designated ports to the forwarding state.

Both MSTP and RSTP improve the spanning-tree operation and maintain backward-compatibility with equipment that is based on the (original) IEEE 802.1D spanning tree, with existing Cisco-proprietary Multiple Instance STP (MISTP) and with existing Cisco per-VLAN spanning-tree plus (PVST+) and rapid per-VLAN spanning-tree plus (rapid PVST+). For information about PVST+ and rapid PVST+, see Chapter 15, "Configuring STP." For information about other spanning-tree features such as Port Fast, UplinkFast, root guard, and so forth, see Chapter 17, "Configuring Optional Spanning-Tree Features."

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding MSTP, page 16-2
- Understanding RSTP, page 16-8
- Configuring MSTP Features, page 16-14
- Displaying the MST Configuration and Status, page 16-26

Understanding MSTP

MSTP, which uses RSTP for rapid convergence, enables VLANs to be grouped into a spanning-tree instance, with each instance having a spanning-tree topology independent of other spanning-tree instances. This architecture provides multiple forwarding paths for data traffic, enables load balancing, and reduces the number of spanning-tree instances required to support a large number of VLANs.

These sections describe how the MSTP works:

- Multiple Spanning-Tree Regions, page 16-2
- IST, CIST, and CST, page 16-2
- Hop Count, page 16-5
- Boundary Ports, page 16-6
- IEEE 802.1s Implementation, page 16-6
- Interoperability with IEEE 802.1D STP, page 16-8

For configuration information, see the "Configuring MSTP Features" section on page 16-14.

Multiple Spanning-Tree Regions

For switches to participate in multiple spanning-tree (MST) instances, you must consistently configure the switches with the same MST configuration information. A collection of interconnected switches that have the same MST configuration comprises an MST region as shown in Figure 16-1 on page 16-4.

The MST configuration determines to which MST region each switch belongs. The configuration includes the name of the region, the revision number, and the MST VLAN-to-instance assignment map. You configure the switch for a region by using the **spanning-tree mst configuration** global configuration command, after which the switch enters the MST configuration mode. From this mode, you can map VLANs to an MST instance by using the **instance** MST configuration command, specify the region name by using the **name** MST configuration command, and set the revision number by using the **revision** MST configuration command.

A region can have one or multiple members with the same MST configuration; each member must be capable of processing RSTP bridge protocol data units (BPDUs). There is no limit to the number of MST regions in a network, but each region can only support up to 65 spanning-tree instances. Instances can be identified by any number in the range from 0 to 4094. You can assign a VLAN to only one spanning-tree instance at a time.

IST, CIST, and CST

Unlike PVST+ and rapid PVST+ in which all the spanning-tree instances are independent, the MSTP establishes and maintains two types of spanning trees:

- An internal spanning tree (IST), which is the spanning tree that runs in an MST region.

  Within each MST region, the MSTP maintains multiple spanning-tree instances. Instance 0 is a special instance for a region, known as the internal spanning tree (IST). All other MST instances are numbered from 1 to 4094.

The IST is the only spanning-tree instance that sends and receives BPDUs. All of the other spanning-tree instance information is contained in M-records, which are encapsulated within MSTP BPDUs. Because the MSTP BPDU carries information for all instances, the number of BPDUs that need to be processed to support multiple spanning-tree instances is significantly reduced.

All MST instances within the same region share the same protocol timers, but each MST instance has its own topology parameters, such as root switch ID, root path cost, and so forth. By default, all VLANs are assigned to the IST.

An MST instance is local to the region; for example, MST instance 1 in region A is independent of MST instance 1 in region B, even if regions A and B are interconnected.

- A common and internal spanning tree (CIST), which is a collection of the ISTs in each MST region, and the common spanning tree (CST) that interconnects the MST regions and single spanning trees.

  The spanning tree computed in a region appears as a subtree in the CST that encompasses the entire switched domain. The CIST is formed by the spanning-tree algorithm among switches that support the IEEE 802.1w, IEEE 802.1s, and IEEE 802.1D standards. The CIST in a MST region is the same as the CST outside a region.

For more information, see the "Operations Within an MST Region" section on page 16-3 and the "Operations Between MST Regions" section on page 16-3.

Note The implementation of the IEEE 802.1s standard changes some of the terminology associated with MST implementations. For a summary of these changes, see Table 16-1 on page 16-5

Operations Within an MST Region

The IST connects all the MSTP switches in a region. When the IST converges, the root of the IST becomes the CIST regional root (called the *IST master* before the implementation of the IEEE 802.1s standard as shown in Figure 16-1 on page 16-4. It is the switch within the region with the lowest switch ID and path cost to the CIST root. The CIST regional root is also the CIST root if there is only one region in the network. If the CIST root is outside the region, one of the MSTP switches at the boundary of the region is selected as the CIST regional root.

When an MSTP switch initializes, it sends BPDUs claiming itself as the root of the CIST and the CIST regional root, with both of the path costs to the CIST root and to the CIST regional root set to zero. The switch also initializes all of its MST instances and claims to be the root for all of them. If the switch receives superior MST root information (lower switch ID, lower path cost, and so forth) than stored for the switch, it relinquishes its claim as the CIST regional root.

During initialization, a region might have many subregions, each with its own CIST regional root. As switches receive superior IST information, they leave their old subregions and join the new subregion that contains the true CIST regional root. Thus all subregions shrink, except for the one that contains the true CIST regional root.

For correct operation, all switches in the MST region must agree on the same CIST regional root. Therefore, any two switches in the region synchronize their port roles for an MST instance only if they converge to a common CIST regional root.

Operations Between MST Regions

If there are multiple regions or legacy IEEE 802.1D switches within the network, MSTP establishes and maintains the CST, which includes all MST regions and all legacy STP switches in the network. The MST instances combine with the IST at the boundary of the region to become the CST.

The IST connects all the MSTP switches in the region and appears as a subtree in the CIST that encompasses the entire switched domain. The root of the subtree is the CIST regional root. The MST region appears as a virtual switch to adjacent STP switches and MST regions.

Figure 16-1 shows a network with three MST regions and a legacy IEEE 802.1D switch (D). The CIST regional root for region 1 (A) is also the CIST root. The CIST regional root for region 2 (B) and the CIST regional root for region 3 (C) are the roots for their respective subtrees within the CIST. The RSTP runs in all regions.

Figure 16-1 MST Regions, CIST Regional Roots, and the CST Root

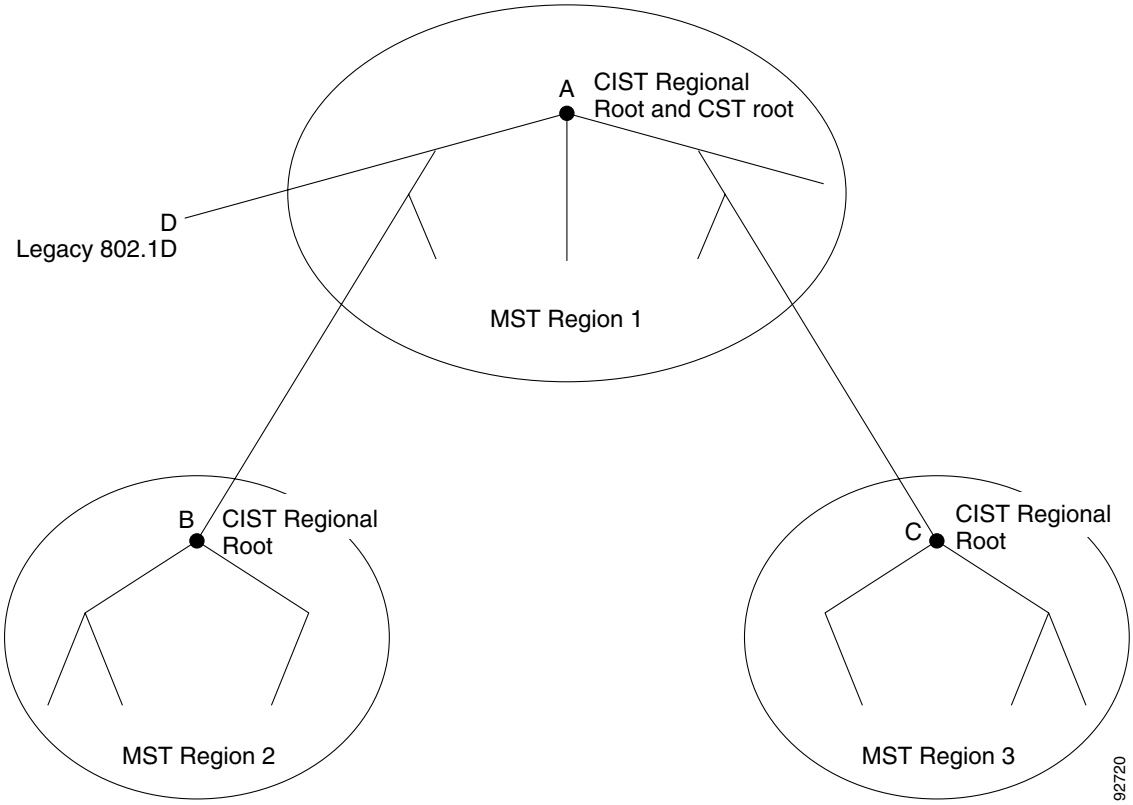


Figure 16-1 does not show additional MST instances for each region. Note that the topology of MST instances can be different from that of the IST for the same region.

Only the CST instance sends and receives BPDUs, and MST instances add their spanning-tree information into the BPDUs to interact with neighboring switches and compute the final spanning-tree topology. Because of this, the spanning-tree parameters related to BPDU transmission (for example, hello time, forward time, max-age, and max-hops) are configured only on the CST instance but affect all MST instances. Parameters related to the spanning-tree topology (for example, switch priority, port VLAN cost, and port VLAN priority) can be configured on both the CST instance and the MST instance.

MSTP switches use version 3 RSTP BPDUs or IEEE 802.1D STP BPDUs to communicate with legacy IEEE 802.1D switches. MSTP switches use MSTP BPDUs to communicate with MSTP switches.

IEEE 802.1s Terminology

Some MST naming conventions used in Cisco's prestandard implementation have been changed to identify some *internal* or *regional* parameters. These parameters are significant only within an MST region, as opposed to external parameters that are relevant to the whole network. Because the CIST is the only spanning-tree instance that spans the whole network, only the CIST parameters require the external rather than internal or regional qualifiers.

- The CIST root is the root switch for the unique instance that spans the whole network, the CIST.
- The CIST external root path cost is the cost to the CIST root. This cost is left unchanged within an MST region. Remember that an MST region looks like a single switch for the CIST. The CIST external root path cost is the root path cost calculated between these virtual switches and switches that do not belong to any region.
- The CIST regional root was called the IST master in the prestandard implementation. If the CIST root is in the region, the CIST regional root is the CIST root. Otherwise, the CIST regional root is the closest switch to the CIST root in the region. The CIST regional root acts as a root switch for the IST.
- The CIST internal root path cost is the cost to the CIST regional root in a region. This cost is only relevant to the IST, instance 0.

Table 16-1 on page 16-5 compares the IEEE standard and the Cisco prestandard terminology.

Table 16-1 Prestandard and Standard Terminology

IEEE Standard	Cisco Prestandard	Cisco Standard
CIST regional root	IST master	CIST regional root
CIST internal root path cost	IST master path cost	CIST internal path cost
CIST external root path cost	Root path cost	Root path cost
MSTI regional root	Instance root	Instance root
MSTI Internal root path cost	Root path cost	Root path cost

Hop Count

The IST and MST instances do not use the message-age and maximum-age information in the configuration BPDU to compute the spanning-tree topology. Instead, they use the path cost to the root and a hop-count mechanism similar to the IP time-to-live (TTL) mechanism.

By using the **spanning-tree mst max-hops** global configuration command, you can configure the maximum hops inside the region and apply it to the IST and all MST instances in that region. The hop count achieves the same result as the message-age information (determines when to trigger a reconfiguration). The root switch of the instance always sends a BPDU (or M-record) with a cost of 0 and the hop count set to the maximum value. When a switch receives this BPDU, it decrements the received remaining hop count by one and propagates this value as the remaining hop count in the BPDUs it generates. When the count reaches zero, the switch discards the BPDU and ages the information held for the port.

The message-age and maximum-age information in the RSTP portion of the BPDU remain the same throughout the region, and the same values are propagated by the region designated ports at the boundary.

Boundary Ports

In the Cisco prestandard implementation, a boundary port connects an MST region to a single spanning-tree region running RSTP, to a single spanning-tree region running PVST+ or rapid PVST+, or to another MST region with a different MST configuration. A boundary port also connects to a LAN, the designated switch of which is either a single spanning-tree switch or a switch with a different MST configuration.

There is no definition of a boundary port in the IEEE 802.1s standard. The IEEE 802.1Q-2002 standard identifies two kinds of messages that a port can receive: internal (coming from the same region) and external. When a message is external, it is received only by the CIST. If the CIST role is root or alternate, or if the external BPDU is a topology change, it could have an impact on the MST instances. When a message is internal, the CIST part is received by the CIST, and each MST instance receives its respective M-record. The Cisco prestandard implementation treats a port that receives an external message as a boundary port. This means a port cannot receive a mix of internal and external messages.

An MST region includes both switches and LANs. A segment belongs to the region of its designated port. Therefore, a port in a different region than the designated port for a segment is a boundary port. This definition allows two ports internal to a region to share a segment with a port belonging to a different region, creating the possibility of receiving both internal and external messages on a port.

The primary change from the Cisco prestandard implementation is that a designated port is not defined as boundary, unless it is running in an STP-compatible mode.

Note If there is a legacy STP switch on the segment, messages are always considered external.

The other change from the prestandard implementation is that the CIST regional root switch ID field is now inserted where an RSTP or legacy IEEE 802.1Q switch has the sender switch ID. The whole region performs like a single virtual switch by sending a consistent sender switch ID to neighboring switches. In this example, switch C would receive a BPDU with the same consistent sender switch ID of root, whether or not A or B is designated for the segment.

IEEE 802.1s Implementation

The Cisco implementation of the IEEE MST standard includes features required to meet the standard, as well as the desirable prestandard functionality that is not yet incorporated into the published standard.

Port Role Naming Change

The boundary role is no longer in the final MST standard, but the boundary concept is maintained in Cisco's implementation. However, an MST instance port at a region boundary might not follow the state of the corresponding CIST port. Two cases exist now:

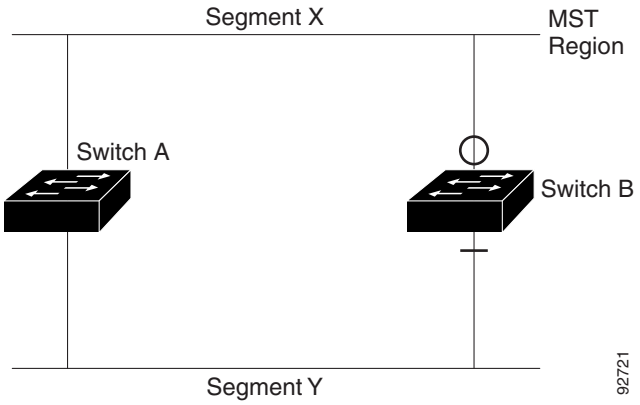
- The boundary port is the root port of the CIST regional root— When the CIST Instance port is proposed and is in sync, it can send back an agreement and move to the forwarding state only after all the corresponding MSTI ports are in sync (and thus forwarding). The MSTI ports now have a special *Master* role.
- The boundary port is not the root port of the CIST regional root—The MSTI ports follow the state and role of the CIST port. The standard provides less information, and it might be difficult to understand why an MSTI port can be alternately blocking when it receives no BPDUs (MRecords). In this case, although the boundary role no longer exists, the **show** commands identify a port as boundary in the *type* column of the output.

Interoperation Between Legacy and Standard Switches

Because automatic detection of prestandard switches can fail, you can use an interface configuration command to identify prestandard ports. A region cannot be formed between a standard and a prestandard switch, but they can interoperate by using the CIST. Only the capability of load balancing over different instances is lost in that particular case. The CLI displays different flags depending on the port configuration when a port receives prestandard BPDUs. A syslog message also appears the first time a switch receives a prestandard BPDU on a port that has not been configured for prestandard BPDU transmission.

Figure 16-2 on page 16-7 illustrates this scenario. Assume that A is a standard switch and B a prestandard switch, both configured to be in the same region. A is the root switch for the CIST, and thus B has a root port (BX) on segment X and an alternate port (BY) on segment Y. If segment Y flaps, and the port on BY becomes the alternate before sending out a single prestandard BPDU, AY cannot detect that a prestandard switch is connected to Y, and continues to send standard BPDUs. The port BY is thus fixed in a boundary, and no load balancing is possible between A and B. The same problem exists on segment X, but, B might transmit topology changes.

Figure 16-2 Standard and Prestandard Switch Interoperation



Note We recommend that you minimize the interaction between standard and prestandard MST implementations.

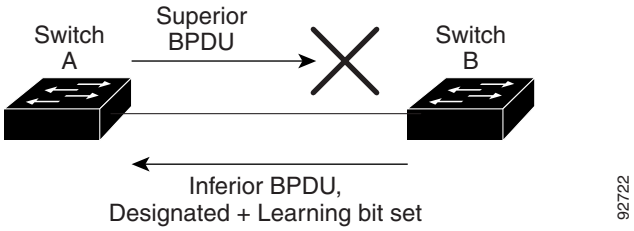
Detecting Unidirectional Link Failure

This feature is not yet present in the IEEE MST standard, but it is included in this Cisco IOS release. The software checks the consistency of the port role and state in the received BPDUs to detect unidirectional link failures that could cause bridging loops.

When a designated port detects a conflict, it keeps its role, but reverts to discarding state because disrupting connectivity in case of inconsistency is preferable to opening a bridging loop.

Figure 16-3 on page 16-8 illustrates a unidirectional link failure that typically creates a switching loop. Switch A is the root switch, and its BPDUs are lost on the link leading to switch B. RSTP and MST BPDUs include the role and state of the sending port. With this information, switch A can detect that switch B does not react to the superior BPDUs it sends and that switch B is the designated, not root switch. As a result, switch A blocks (or keeps blocking) its port, thus preventing the bridging loop.

Figure 16-3 Detecting Unidirectional Link Failure



Interoperability with IEEE 802.1D STP

A switch running MSTP supports a built-in protocol migration mechanism that enables it to interoperate with legacy IEEE 802.1D switches. If this switch receives a legacy IEEE 802.1D configuration BPDU (a BPDU with the protocol version set to 0), it sends only IEEE 802.1D BPDUs on that port. An MSTP switch can also detect that a port is at the boundary of a region when it receives a legacy BPDU, an MSTP BPDU (version 3) associated with a different region, or an RSTP BPDU (version 2).

However, the switch does not automatically revert to the MSTP mode if it no longer receives IEEE 802.1D BPDUs because it cannot determine whether the legacy switch has been removed from the link unless the legacy switch is the designated switch. A switch might also continue to assign a boundary role to a port when the switch to which this switch is connected has joined the region. To restart the protocol migration process (force the renegotiation with neighboring switches), use the **clear spanning-tree detected-protocols** privileged EXEC command.

If all the legacy switches on the link are RSTP switches, they can process MSTP BPDUs as if they are RSTP BPDUs. Therefore, MSTP switches send either a version 0 configuration and TCN BPDUs or version 3 MSTP BPDUs on a boundary port. A boundary port connects to a LAN, the designated switch of which is either a single spanning-tree switch or a switch with a different MST configuration.

Understanding RSTP

The RSTP takes advantage of point-to-point wiring and provides rapid convergence of the spanning tree. Reconfiguration of the spanning tree can occur in less than 1 second (in contrast to 50 seconds with the default settings in the IEEE 802.1D spanning tree).

These section describes how the RSTP works:

- Port Roles and the Active Topology, page 16-9
- Rapid Convergence, page 16-10
- Synchronization of Port Roles, page 16-11
- Bridge Protocol Data Unit Format and Processing, page 16-12

For configuration information, see the "Configuring MSTP Features" section on page 16-14.

Port Roles and the Active Topology

The RSTP provides rapid convergence of the spanning tree by assigning port roles and by determining the active topology. The RSTP builds upon the IEEE 802.1D STP to select the switch with the highest switch priority (lowest numerical priority value) as the root switch as described in the "Spanning-Tree Topology and BPDUs" section on page 15-3. Then the RSTP assigns one of these port roles to individual ports:

- Root port—Provides the best path (lowest cost) when the switch forwards packets to the root switch.
- Designated port—Connects to the designated switch, which incurs the lowest path cost when forwarding packets from that LAN to the root switch. The port through which the designated switch is attached to the LAN is called the designated port.
- Alternate port—Offers an alternate path toward the root switch to that provided by the current root port.
- Backup port—Acts as a backup for the path provided by a designated port toward the leaves of the spanning tree. A backup port can exist only when two ports are connected in a loopback by a point-to-point link or when a switch has two or more connections to a shared LAN segment.
- Disabled port—Has no role within the operation of the spanning tree.

A port with the root or a designated port role is included in the active topology. A port with the alternate or backup port role is excluded from the active topology.

In a stable topology with consistent port roles throughout the network, the RSTP ensures that every root port and designated port immediately transition to the forwarding state while all alternate and backup ports are always in the discarding state (equivalent to blocking in IEEE 802.1D). The port state controls the operation of the forwarding and learning processes. Table 16-2 provides a comparison of IEEE 802.1D and RSTP port states.

Table 16-2 Port State Comparison

Operational Status	STP Port State (IEEE 802.1D)	RSTP Port State	Is Port Included in the Active Topology?
Enabled	Blocking	Discarding	No
Enabled	Listening	Discarding	No
Enabled	Learning	Learning	Yes
Enabled	Forwarding	Forwarding	Yes
Disabled	Disabled	Discarding	No

To be consistent with Cisco STP implementations, this guide defines the port state as *blocking* instead of *discarding*. Designated ports start in the listening state.

Rapid Convergence

The RSTP provides for rapid recovery of connectivity following the failure of a switch, a switch port, or a LAN. It provides rapid convergence for edge ports, new root ports, and ports connected through point-to-point links as follows:

- Edge ports—If you configure a port as an edge port on an RSTP switch by using the **spanning-tree portfast** interface configuration command, the edge port immediately transitions to the forwarding state. An edge port is the same as a Port Fast-enabled port, and you should enable it only on ports that connect to a single end station.
- Root ports—If the RSTP selects a new root port, it blocks the old root port and immediately transitions the new root port to the forwarding state.
- Point-to-point links—If you connect a port to another port through a point-to-point link and the local port becomes a designated port, it negotiates a rapid transition with the other port by using the proposal-agreement handshake to ensure a loop-free topology.

  As shown in Figure 16-4, Switch A is connected to Switch B through a point-to-point link, and all of the ports are in the blocking state. Assume that the priority of Switch A is a smaller numerical value than the priority of Switch B. Switch A sends a proposal message (a configuration BPDU) with the proposal flag set) to Switch B, proposing itself as the designated switch.

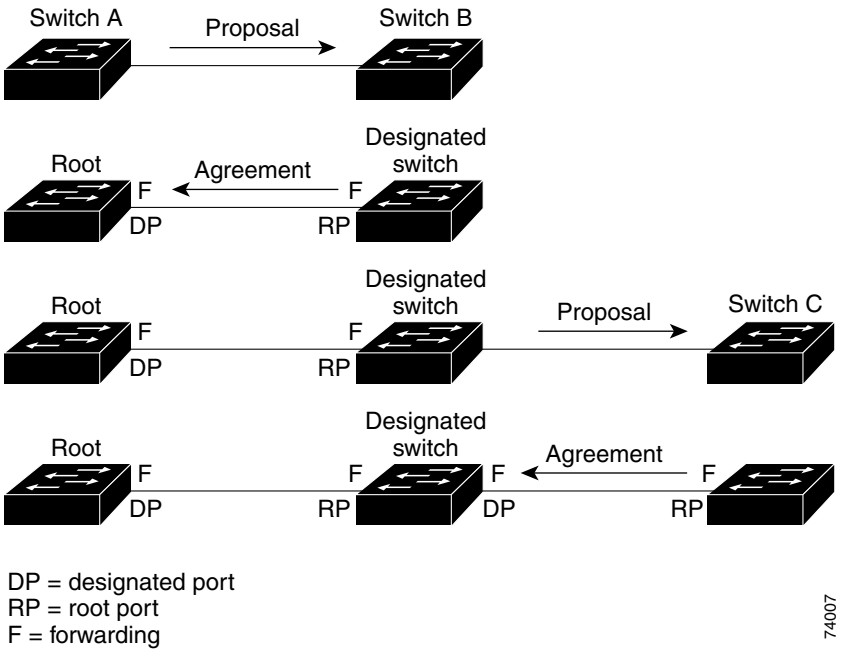
  After receiving the proposal message, Switch B selects as its new root port the port from which the proposal message was received, forces all nonedge ports to the blocking state, and sends an agreement message (a BPDU with the agreement flag set) through its new root port.

  After receiving Switch B's agreement message, Switch A also immediately transitions its designated port to the forwarding state. No loops in the network are formed because Switch B blocked all of its nonedge ports and because there is a point-to-point link between Switches A and B.

  When Switch C is connected to Switch B, a similar set of handshaking messages are exchanged. Switch C selects the port connected to Switch B as its root port, and both ends immediately transition to the forwarding state. With each iteration of this handshaking process, one more switch joins the active topology. As the network converges, this proposal-agreement handshaking progresses from the root toward the leaves of the spanning tree.

  The switch determines the link type from the port duplex mode: a full-duplex port is considered to have a point-to-point connection; a half-duplex port is considered to have a shared connection. You can override the default setting that is determined by the duplex setting by using the **spanning-tree link-type** interface configuration command.

Figure 16-4 Proposal and Agreement Handshaking for Rapid Convergence



Synchronization of Port Roles

When the switch receives a proposal message on one of its ports and that port is selected as the new root port, the RSTP forces all other ports to synchronize with the new root information.

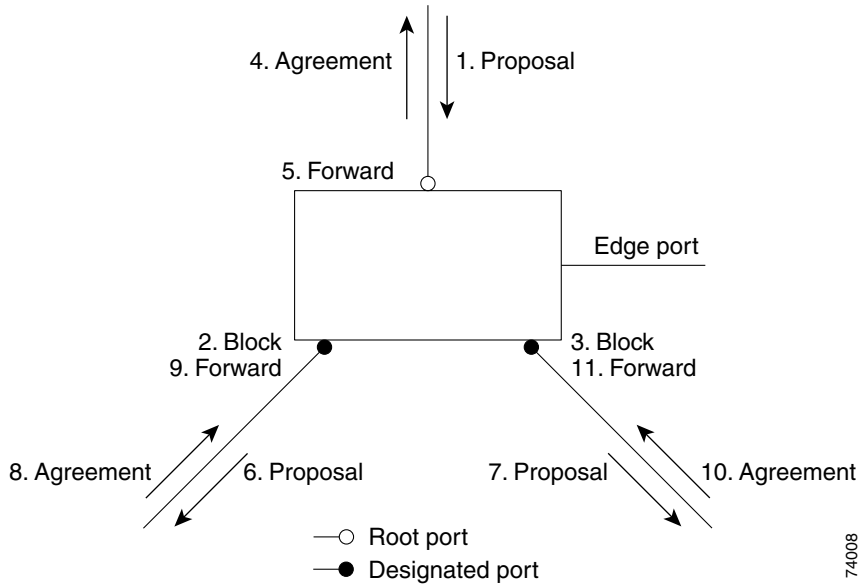
The switch is synchronized with superior root information received on the root port if all other ports are synchronized. An individual port on the switch is synchronized if

- That port is in the blocking state
- It is an edge port (a port configured to be at the edge of the network)

If a designated port is in the forwarding state and is not configured as an edge port, it transitions to the blocking state when the RSTP forces it to synchronize with new root information. In general, when the RSTP forces a port to synchronize with root information and the port does not satisfy any of the above conditions, its port state is set to blocking.

After ensuring that all of the ports are synchronized, the switch sends an agreement message to the designated switch corresponding to its root port. When the switches connected by a point-to-point link are in agreement about their port roles, the RSTP immediately transitions the port states to forwarding. The sequence of events is shown in Figure 16-5.

Figure 16-5 Sequence of Events During Rapid Convergence



Bridge Protocol Data Unit Format and Processing

The RSTP BPDU format is the same as the IEEE 802.1D BPDU format except that the protocol version is set to 2. A new 1-byte version 1 Length field is set to zero, which means that no version 1 protocol information is present. Table 16-3 shows the RSTP flag fields.

Table 16-3 RSTP BPDU Flags

Bit	Function
0	Topology change (TC)
1	Proposal
2–3:	Port role:
00	Unknown
01	Alternate port
10	Root port
11	Designated port
4	Learning
5	Forwarding
6	Agreement
7	Topology change acknowledgement (TCA)

The sending switch sets the proposal flag in the RSTP BPDU to propose itself as the designated switch on that LAN. The port role in the proposal message is always set to the designated port.

The sending switch sets the agreement flag in the RSTP BPDU to accept the previous proposal. The port role in the agreement message is always set to the root port.

The RSTP does not have a separate topology change notification (TCN) BPDU. It uses the topology change (TC) flag to show the topology changes. However, for interoperability with IEEE 802.1D switches, the RSTP switch processes and generates TCN BPDUs.

The learning and forwarding flags are set according to the state of the sending port.

Processing Superior BPDU Information

If a port receives superior root information (lower switch ID, lower path cost, and so forth) than currently stored for the port, the RSTP triggers a reconfiguration. If the port is proposed and is selected as the new root port, RSTP forces all the other ports to synchronize.

If the BPDU received is an RSTP BPDU with the proposal flag set, the switch sends an agreement message after all of the other ports are synchronized. If the BPDU is an IEEE 802.1D BPDU, the switch does not set the proposal flag and starts the forward-delay timer for the port. The new root port requires twice the forward-delay time to transition to the forwarding state.

If the superior information received on the port causes the port to become a backup or alternate port, RSTP sets the port to the blocking state but does not send the agreement message. The designated port continues sending BPDUs with the proposal flag set until the forward-delay timer expires, at which time the port transitions to the forwarding state.

Processing Inferior BPDU Information

If a designated port receives an inferior BPDU (higher switch ID, higher path cost, and so forth than currently stored for the port) with a designated port role, it immediately replies with its own information.

Topology Changes

This section describes the differences between the RSTP and the IEEE 802.1D in handling spanning-tree topology changes.

- Detection—Unlike IEEE 802.1D in which *any* transition between the blocking and the forwarding state causes a topology change, *only* transitions from the blocking to the forwarding state cause a topology change with RSTP (only an increase in connectivity is considered a topology change). State changes on an edge port do not cause a topology change. When an RSTP switch detects a topology change, it deletes the learned information on all of its nonedge ports except on those from which it received the TC notification.
- Notification—Unlike IEEE 802.1D, which uses TCN BPDUs, the RSTP does not use them. However, for IEEE 802.1D interoperability, an RSTP switch processes and generates TCN BPDUs.
- Acknowledgement—When an RSTP switch receives a TCN message on a designated port from an IEEE 802.1D switch, it replies with an IEEE 802.1D configuration BPDU with the TCA bit set. However, if the TC-while timer (the same as the topology-change timer in IEEE 802.1D) is active on a root port connected to an IEEE 802.1D switch and a configuration BPDU with the TCA bit set is received, the TC-while timer is reset.

  This behavior is only required to support IEEE 802.1D switches. The RSTP BPDUs never have the TCA bit set.
- Propagation—When an RSTP switch receives a TC message from another switch through a designated or root port, it propagates the change to all of its nonedge, designated ports and to the root port (excluding the port on which it is received). The switch starts the TC-while timer for all such ports and flushes the information learned on them.

- Protocol migration—For backward compatibility with IEEE 802.1D switches, RSTP selectively sends IEEE 802.1D configuration BPDUs and TCN BPDUs on a per-port basis.

  When a port is initialized, the migrate-delay timer is started (specifies the minimum time during which RSTP BPDUs are sent), and RSTP BPDUs are sent. While this timer is active, the switch processes all BPDUs received on that port and ignores the protocol type.

  If the switch receives an IEEE 802.1D BPDU after the port migration-delay timer has expired, it assumes that it is connected to an IEEE 802.1D switch and starts using only IEEE 802.1D BPDUs. However, if the RSTP switch is using IEEE 802.1D BPDUs on a port and receives an RSTP BPDU after the timer has expired, it restarts the timer and starts using RSTP BPDUs on that port.

Configuring MSTP Features

These sections describe how to configure basic MSTP features:

- Default MSTP Configuration, page 16-14
- MSTP Configuration Guidelines, page 16-15
- Specifying the MST Region Configuration and Enabling MSTP, page 16-16 (required)
- Configuring the Root Switch, page 16-17 (optional)
- Configuring a Secondary Root Switch, page 16-19 (optional)
- Configuring the Port Priority, page 16-20 (optional)
- Configuring the Path Cost, page 16-21 (optional)
- Configuring the Switch Priority, page 16-21 (optional)
- Configuring the Hello Time, page 16-22 (optional)
- Configuring the Forwarding-Delay Time, page 16-23 (optional)
- Configuring the Maximum-Aging Time, page 16-23 (optional)
- Configuring the Maximum-Hop Count, page 16-24 (optional)
- Specifying the Link Type to Ensure Rapid Transitions, page 16-24 (optional)
- Designating the Neighbor Type, page 16-25
- Restarting the Protocol Migration Process, page 16-26 (optional)

Default MSTP Configuration

Table 16-4 shows the default MSTP configuration.

Table 16-4 Default MSTP Configuration

Feature	Default Setting
Spanning-tree mode	PVST+ (Rapid PVST+ and MSTP are disabled).
Switch priority (configurable on a per-CIST interface basis)	32768.
Spanning-tree port priority (configurable on a per-CIST interface basis)	128.

Table 16-4 Default MSTP Configuration (continued)

Feature	Default Setting
Spanning-tree port cost (configurable on a per-CIST interface basis)	1000 Mbps: 4. 100 Mbps: 19 10 Mbps: 100.
Hello time	2 seconds.
Forward-delay time	15 seconds.
Maximum-aging time	20 seconds.
Maximum hop count	20 hops.

For information about the supported number of spanning-tree instances, see the "Supported Spanning-Tree Instances" section on page 15-9.

MSTP Configuration Guidelines

These are the configuration guidelines for MSTP:

- When you enable MST by using the **spanning-tree mode mst** global configuration command, RSTP is automatically enabled. Per-VLAN RSTP is not supported in software releases earlier than Cisco IOS Release 12.1(13)EA1.
- For two or more switches to be in the same MST region, they must have the same VLAN-to-instance map, the same configuration revision number, and the same name.
- The switch supports up to 65 MST instances. The number of VLANs that can be mapped to a particular MST instance is unlimited.
- PVST+, rapid PVST+, and MSTP are supported, but only one version can be active at any time. (For example, all VLANs run PVST+, all VLANs run rapid PVST+, or all VLANs run MSTP.) For more information, see the "Spanning-Tree Interoperability and Backward Compatibility" section on page 15-10. For information on the recommended trunk port configuration, see the"Interaction with Other Features" section on page 11-19.
- VTP propagation of the MST configuration is not supported. However, you can manually configure the MST configuration (region name, revision number, and VLAN-to-instance mapping) on each switch within the MST region by using the command-line interface (CLI) or through the SNMP support.
- For load balancing across redundant paths in the network to work, all VLAN-to-instance mapping assignments must match; otherwise, all traffic flows on a single link.
- All MST boundary ports must be forwarding for load balancing between a PVST+ and an MST cloud or between a rapid-PVST+ and an MST cloud. For this to occur, the IST master of the MST cloud should also be the root of the CST. If the MST cloud consists of multiple MST regions, one of the MST regions must contain the CST root, and all of the other MST regions must have a better path to the root contained within the MST cloud than a path through the PVST+ or rapid-PVST+ cloud. You might have to manually configure the switches in the clouds.

- Partitioning the network into a large number of regions is not recommended. However, if this situation is unavoidable, we recommend that you partition the switched LAN into smaller LANs interconnected by routers or non-Layer 2 devices.
- For configuration guidelines about UplinkFast, BackboneFast, and cross-stack UplinkFast, see the "Optional Spanning-Tree Configuration Guidelines" section on page 17-14.

Specifying the MST Region Configuration and Enabling MSTP

For two or more switches to be in the same MST region, they must have the same VLAN-to-instance mapping, the same configuration revision number, and the same name.

A region can have one member or multiple members with the same MST configuration; each member must be capable of processing RSTP BPDUs. There is no limit to the number of MST regions in a network, but each region can only support up to 65 spanning-tree instances. You can assign a VLAN to only one spanning-tree instance at a time.

Beginning in privileged EXEC mode, follow these steps to specify the MST region configuration and enable MSTP. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst configuration**	Enter MST configuration mode.
Step 3	**instance** *instance-id* **vlan** *vlan-range*	Map VLANs to an MST instance. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 1 to 4094. • For **vlan** *vlan-range*, the range is 1 to 4094. When you map VLANs to an MST instance, the mapping is incremental, and the VLANs specified in the command are added to or removed from the VLANs that were previously mapped. To specify a VLAN range, use a hyphen; for example, **instance 1 vlan 1-63** maps VLANs 1 through 63 to MST instance 1. To specify a VLAN series, use a comma; for example, **instance 1 vlan 10, 20, 30** maps VLANs 10, 20, and 30 to MST instance 1.
Step 4	**name** *name*	Specify the configuration name. The *name* string has a maximum length of 32 characters and is case sensitive.
Step 5	**revision** *version*	Specify the configuration revision number. The range is 0 to 65535.
Step 6	**show pending**	Verify your configuration by displaying the pending configuration.
Step 7	**exit**	Apply all changes, and return to global configuration mode.

	Command	Purpose
Step 8	**spanning-tree mode mst**	Enable MSTP. RSTP is also enabled. ⚠ **Caution** Changing spanning-tree modes can disrupt traffic because all spanning-tree instances are stopped for the previous mode and restarted in the new mode. You cannot run both MSTP and PVST+ or both MSTP and rapid PVST+ at the same time.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show running-config**	Verify your entries.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default MST region configuration, use the **no spanning-tree mst configuration** global configuration command. To return to the default VLAN-to-instance map, use the **no instance** *instance-id* [**vlan** *vlan-range*] MST configuration command. To return to the default name, use the **no name** MST configuration command. To return to the default revision number, use the **no revision** MST configuration command. To re-enable PVST+, use the **no spanning-tree mode** or the **spanning-tree mode pvst** global configuration command.

This example shows how to enter MST configuration mode, map VLANs 10 to 20 to MST instance 1, name the region *region1*, set the configuration revision to 1, display the pending configuration, apply the changes, and return to global configuration mode:

```
Switch(config)# spanning-tree mst configuration
Switch(config-mst)# instance 1 vlan 10-20
Switch(config-mst)# name region1
Switch(config-mst)# revision 1
Switch(config-mst)# show pending
Pending MST configuration
Name      [region1]
Revision  1
Instance  Vlans Mapped
--------  ---------------------
0         1-9,21-4094
1         10-20
-------------------------------

Switch(config-mst)# exit
Switch(config)#
```

Configuring the Root Switch

The switch maintains a spanning-tree instance for the group of VLANs mapped to it. A switch ID, consisting of the switch priority and the switch MAC address, is associated with each instance. The switch with the lowest switch ID becomes the root switch for the group of VLANs.

To configure a switch to become the root, use the **spanning-tree mst** *instance-id* **root** global configuration command to modify the switch priority from the default value (32768) to a significantly lower value so that the switch becomes the root switch for the specified spanning-tree instance. When you enter this command, the switch checks the switch priorities of the root switches. Because of the extended system ID support, the switch sets its own priority for the specified instance to 24576 if this value will cause this switch to become the root for the specified spanning-tree instance.

If any root switch for the specified instance has a switch priority lower than 24576, the switch sets its own priority to 4096 less than the lowest switch priority. (4096 is the value of the least-significant bit of a 4-bit switch priority value as shown in Table 15-1 on page 15-4.)

Note Catalyst 3550 switches running software earlier than Cisco IOS Release 12.1(8)EA1 do not support the extended system ID. Catalyst 3550 switches running software earlier than Cisco IOS Release 12.1(9)EA1 do not support the MSTP.

If your network consists of switches that both do and do not support the extended system ID, it is unlikely that the switch with the extended system ID support will become the root switch. The extended system ID increases the switch priority value every time the VLAN number is greater than the priority of the connected switches running older software.

The root switch for each spanning-tree instance should be a backbone or distribution switch. Do not configure an access switch as the spanning-tree primary root.

Use the **diameter** keyword, which is available only for MST instance 0, to specify the Layer 2 network diameter (that is, the maximum number of switch hops between any two end stations in the Layer 2 network). When you specify the network diameter, the switch automatically sets an optimal hello time, forward-delay time, and maximum-age time for a network of that diameter, which can significantly reduce the convergence time. You can use the **hello** keyword to override the automatically calculated hello time.

Note After configuring the switch as the root switch, we recommend that you avoid manually configuring the hello time, forward-delay time, and maximum-age time by using the **spanning-tree mst hello-time**, **spanning-tree mst forward-time**, and the **spanning-tree mst max-age** global configuration commands after configuring the switch as the root switch.

Beginning in privileged EXEC mode, follow these steps to configure a switch as the root switch. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst** *instance-id* **root primary** [**diameter** *net-diameter* [**hello-time** *seconds*]]	Configure a switch as the root switch. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094. • (Optional) For **diameter** *net-diameter*, specify the maximum number of switches between any two end stations. The range is 2 to 7. This keyword is available only for MST instance 0. • (Optional) For **hello-time** *seconds*, specify the interval in seconds between the generation of configuration messages by the root switch. The range is 1 to 10 seconds; the default is 2 seconds.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst** *instance-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree mst** *instance-id* **root** global configuration command.

Configuring a Secondary Root Switch

When you configure a Catalyst 3550 switch that supports the extended system ID as the secondary root, the spanning-tree switch priority is modified from the default value (32768) to 28672. The switch is then likely to become the root switch for the specified instance if the primary root switch fails. This is assuming that the other network switches use the default switch priority of 32768 and therefore are unlikely to become the root switch. For Catalyst 3550 switches without the extended system ID support (software earlier than Cisco IOS Release 12.1(8)EA1), the switch priority is changed to 16384.

You can execute this command on more than one switch to configure multiple backup root switches. Use the same network diameter and hello-time values that you used when you configured the primary root switch with the **spanning-tree mst** *instance-id* **root primary** global configuration command.

Beginning in privileged EXEC mode, follow these steps to configure a switch as the secondary root switch. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst** *instance-id* **root secondary** [**diameter** *net-diameter* [**hello-time** *seconds*]]	Configure a switch as the secondary root switch. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094. • (Optional) For **diameter** *net-diameter*, specify the maximum number of switches between any two end stations. The range is 2 to 7. This keyword is available only for MST instance 0. • (Optional) For **hello-time** *seconds*, specify the interval in seconds between the generation of configuration messages by the root switch. The range is 1 to 10 seconds; the default is 2 seconds. Use the same network diameter and hello-time values that you used when configuring the primary root switch. See the "Configuring the Root Switch" section on page 16-17.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst** *instance-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree mst** *instance-id* **root** global configuration command.

Configuring the Port Priority

If a loop occurs, the MSTP uses the port priority when selecting an interface to put into the forwarding state. You can assign higher priority values (lower numerical values) to interfaces that you want selected first and lower priority values (higher numerical values) that you want selected last. If all interfaces have the same priority value, the MSTP puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Beginning in privileged EXEC mode, follow these steps to configure the MSTP port priority of an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. Valid interfaces include physical ports and port channels. Valid port-channel numbers are 1 to 64.
Step 3	**spanning-tree mst** *instance-id* **port-priority** *priority*	Configure the port priority for an MST instance. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094. • For *priority*, the range is 0 to 240 in increments of 16. The default is 128. The lower the number, the higher the priority. Valid priority values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show spanning-tree mst interface** *interface-id* or **show spanning-tree mst** *instance-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note The **show spanning-tree mst interface** *interface-id* privileged EXEC command displays information only if the port is in a link-up operative state. Otherwise, you can use the **show running-config interface** privileged EXEC command to confirm the configuration.

To return the interface to its default setting, use the **no spanning-tree mst** *instance-id* **port-priority** interface configuration command.

Configuring the Path Cost

The MSTP path cost default value is derived from the media speed of an interface. If a loop occurs, the MSTP uses cost when selecting an interface to put in the forwarding state. You can assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last. If all interfaces have the same cost value, the MSTP puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Beginning in privileged EXEC mode, follow these steps to configure the MSTP cost of an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. Valid interfaces include physical ports and port channels. Valid port-channel numbers are 1 to 64.
Step 3	**spanning-tree mst** *instance-id* **cost** *cost*	Configure the cost for an MST instance. If a loop occurs, the MSTP uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094. • For *cost*, the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show spanning-tree mst interface** *interface-id* or **show spanning-tree mst** *instance-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note The **show spanning-tree mst interface** *interface-id* privileged EXEC command displays information only for ports that are in a link-up operative state. Otherwise, you can use the **show running-config** privileged EXEC command to confirm the configuration.

To return the interface to its default setting, use the **no spanning-tree mst** *instance-id* **cost** interface configuration command.

Configuring the Switch Priority

You can configure the switch priority and make it more likely that the switch will be chosen as the root switch.

Note Exercise care when using this command. For most situations, we recommend that you use the **spanning-tree mst** *instance-id* **root primary** and the **spanning-tree mst** *instance-id* **root secondary** global configuration commands to modify the switch priority.

Beginning in privileged EXEC mode, follow these steps to configure the switch priority. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst** *instance-id* **priority** *priority*	Configure the switch priority for an MST instance. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094. • For *priority*, the range is 0 to 61440 in increments of 4096; the default is 32768. The lower the number, the more likely the switch will be chosen as the root switch. Valid priority values are 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. All other values are rejected.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst** *instance-id*	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree mst** *instance-id* **priority** global configuration command.

Configuring the Hello Time

You can configure the interval between the generation of configuration messages by the root switch by changing the hello time.

Note Exercise care when using this command. For most situations, we recommend that you use the **spanning-tree mst** *instance-id* **root primary** and the **spanning-tree mst** *instance-id* **root secondary** global configuration commands to modify the hello time.

Beginning in privileged EXEC mode, follow these steps to configure the hello time for all MST instances. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst hello-time** *seconds*	Configure the hello time for all MST instances. The hello time is the interval between the generation of configuration messages by the root switch. These messages mean that the switch is alive. For *seconds*, the range is 1 to 10; the default is 2.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree mst hello-time** global configuration command.

Configuring the Forwarding-Delay Time

Beginning in privileged EXEC mode, follow these steps to configure the forwarding-delay time for all MST instances. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst forward-time** *seconds*	Configure the forward time for all MST instances. The forward delay is the number of seconds a port waits before changing from its spanning-tree learning and listening states to the forwarding state. For *seconds*, the range is 4 to 30; the default is 15.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree mst forward-time** global configuration command.

Configuring the Maximum-Aging Time

Beginning in privileged EXEC mode, follow these steps to configure the maximum-aging time for all MST instances. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst max-age** *seconds*	Configure the maximum-aging time for all MST instances. The maximum-aging time is the number of seconds a switch waits without receiving spanning-tree configuration messages before attempting a reconfiguration. For *seconds*, the range is 6 to 40; the default is 20.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree mst max-age** global configuration command.

Configuring the Maximum-Hop Count

Beginning in privileged EXEC mode, follow these steps to configure the maximum-hop count for all MST instances. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree mst max-hops** *hop-count*	Specify the number of hops in a region before the BPDU is discarded, and the information held for a port is aged. For *hop-count*, the range is 1 to 255; the default is 20.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree mst**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree mst max-hops** global configuration command.

Specifying the Link Type to Ensure Rapid Transitions

If you connect a port to another port through a point-to-point link and the local port becomes a designated port, the RSTP negotiates a rapid transition with the other port by using the proposal-agreement handshake to ensure a loop-free topology as described in the "Rapid Convergence" section on page 16-10.

By default, the link type is determined from the duplex mode of the interface: a full-duplex port is considered to have a point-to-point connection; a half-duplex port is considered to have a shared connection. If you have a half-duplex link physically connected point-to-point to a single port on a remote switch running MSTP, you can override the default setting of the link type and enable rapid transitions to the forwarding state.

Beginning in privileged EXEC mode, follow these steps to override the default link-type setting. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to configure, and enter interface configuration mode. Valid interfaces include physical ports, VLANs, and port channels. Valid VLAN IDs are 1 to 4094. Valid port-channel numbers are 1 to 64.
Step 3	**spanning-tree link-type point-to-point**	Specify that the link type of a port is point-to-point.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show spanning-tree mst interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default setting, use the **no spanning-tree link-type** interface configuration command.

Designating the Neighbor Type

A topology could contain both prestandard and IEEE 802.1s standard compliant devices. By default, ports can automatically detect prestandard devices, but they can still receive both standard and prestandard BPDUs. When there is a mismatch between a device and its neighbor, only the CIST runs on the interface.

You can choose to set a port to send only prestandard BPDUs. The prestandard flag appears in all the show commands, even if the port is in STP compatibility mode.

Beginning in privileged EXEC mode, follow these steps to override the default link-type setting. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode. Valid interfaces include physical ports.
Step 3	**spanning-tree mst pre-standard**	Specify that the port can send only prestandard BPDUs.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show spanning-tree mst interface** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the port to its default setting, use the **no spanning-tree mst pre-standard** interface configuration command.

Restarting the Protocol Migration Process

A switch running MSTP supports a built-in protocol migration mechanism that enables it to interoperate with legacy IEEE 802.1D switches. If this switch receives a legacy IEEE 802.1D configuration BPDU (a BPDU with the protocol version set to 0), it sends only IEEE 802.1D BPDUs on that port. An MSTP switch can also detect that a port is at the boundary of a region when it receives a legacy BPDU, an MST BPDU (version 3) associated with a different region, or an RST BPDU (version 2).

However, the switch does not automatically revert to the MSTP mode if it no longer receives IEEE 802.1D BPDUs because it cannot determine whether the legacy switch has been removed from the link unless the legacy switch is the designated switch. A switch also might continue to assign a boundary role to a port when the switch to which it is connected has joined the region.

To restart the protocol migration process (force the renegotiation with neighboring switches) on the switch, use the **clear spanning-tree detected-protocols** privileged EXEC command.

To restart the protocol migration process on a specific interface, use the **clear spanning-tree detected-protocols interface** *interface-id* privileged EXEC command.

Displaying the MST Configuration and Status

To display the spanning-tree status, use one or more of the privileged EXEC commands in Table 16-5:

Table 16-5 Commands for Displaying MST Status

Command	Purpose
show spanning-tree mst configuration	Displays the MST region configuration.
show spanning-tree mst configuration digest	Displays the MD5 digest included in the current MSTCI.
show spanning-tree mst *instance-id*	Displays MST information for the specified instance.
show spanning-tree mst interface *interface-id*	Displays MST information for the specified interface. Valid interfaces include physical ports, VLANs, and port channels. Valid VLAN IDs are 1 to 4094. The valid port-channel range is 1 to 64.

For information about other keywords for the **show spanning-tree** privileged EXEC command, see the command reference for this release.

CHAPTER 17

Configuring Optional Spanning-Tree Features

This chapter describes how to configure optional spanning-tree features on your Catalyst 3550 switch. You can configure all of these features when your switch is running the per-VLAN spanning-tree plus (PVST+). You can configure only the noted features when your switch is running the Multiple Spanning Tree Protocol (MSTP) or the rapid per-VLAN spanning-tree plus (rapid-PVST+) protocol.

For information on configuring the PVST+ and rapid PVST+, see Chapter 15, "Configuring STP." For information about the Multiple Spanning Tree Protocol (MSTP) and how to map multiple VLANs to the same spanning-tree instance, see Chapter 16, "Configuring MSTP."

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding Optional Spanning-Tree Features, page 17-1
- Configuring Optional Spanning-Tree Features, page 17-13
- Displaying the Spanning-Tree Status, page 17-21

Understanding Optional Spanning-Tree Features

These sections describe how the optional spanning-tree features work:

- Understanding Port Fast, page 17-2
- Understanding BPDU Guard, page 17-2
- Understanding BPDU Filtering, page 17-3
- Understanding UplinkFast, page 17-3
- Understanding Cross-Stack UplinkFast, page 17-5
- Understanding BackboneFast, page 17-9
- Understanding EtherChannel Guard, page 17-11
- Understanding Root Guard, page 17-12
- Understanding Loop Guard, page 17-13

Understanding Port Fast

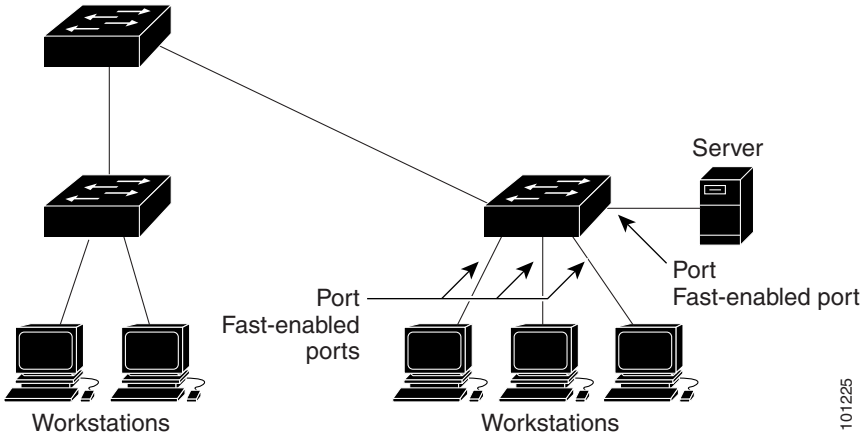
Port Fast immediately brings an interface configured as an access or trunk port from a blocking state to the forwarding state, bypassing the listening and learning states. You can use Port Fast on ports connected to a single workstation or server, as shown in Figure 17-1, to allow those devices to immediately connect to the network, rather than waiting for the spanning tree to converge.

Ports connected to a single workstation or server should not receive bridge protocol data units (BPDUs). A port with Port Fast enabled goes through the normal cycle of spanning-tree status changes when the switch is restarted.

Note Because the purpose of Port Fast is to minimize the time ports must wait for spanning-tree to converge, it is effective only when used on ports connected to end stations. If you enable Port Fast on a port connected to another switch, you risk creating a spanning-tree loop.

You can enable this feature by using the **spanning-tree portfast** interface configuration or the **spanning-tree portfast default** global configuration command.

Figure 17-1 Port Fast-Enabled Ports



Understanding BPDU Guard

The BPDU guard feature can be globally enabled on the switch or can be enabled per interface, but the feature operates with some differences.

At the global level, you can enable BPDU guard on Port Fast-enabled ports by using the **spanning-tree portfast bpduguard default** global configuration command. Spanning tree shuts down ports that are in a Port Fast-operational state if any BPDU is received on those interfaces. In a valid configuration, Port Fast-enabled ports do not receive BPDUs. Receiving a BPDU on a Port Fast-enabled port signals an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the port in the error-disabled state.

At the interface level, you can enable BPDU guard on any port by using the **spanning-tree bpduguard enable** interface configuration command without also enabling the Port Fast feature. When the port receives a BPDU, it is put in the error-disabled state.

The BPDU guard feature provides a secure response to invalid configurations because you must manually put the port back in service. Use the BPDU guard feature in a service-provider network to prevent an access port from participating in the spanning tree.

You can enable the BPDU guard feature for the entire switch or for an interface.

Understanding BPDU Filtering

The BPDU filtering feature can be globally enabled on the switch or can be enabled per interface, but the feature operates with some differences.

At the global level, you can enable BPDU filtering on Port Fast-enabled ports by using the **spanning-tree portfast bpdufilter default** global configuration command. This command prevents ports that are in a Port Fast-operational state from sending or receiving BPDUs. The ports still send a few BPDUs at link-up before the switch begins to filter outbound BPDUs. You should globally enable BPDU filtering on a switch so that hosts connected to these ports do not receive BPDUs. If a BPDU is received on a Port Fast-enabled port, the port loses its Port Fast-operational status, and BPDU filtering is disabled.

At the interface level, you can enable BPDU filtering on any port without also enabling the Port Fast feature by using the **spanning-tree bpdufilter enable** interface configuration command. This command prevents the port from sending or receiving BPDUs.

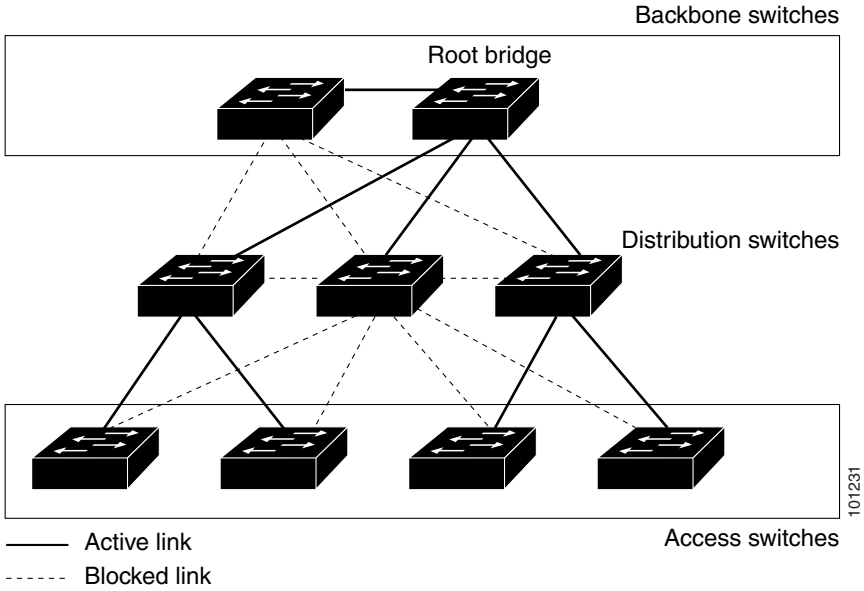
Caution Enabling BPDU filtering on an interface is the same as disabling spanning tree on it and can result in spanning-tree loops.

You can enable the BPDU filtering feature for the entire switch or for an interface.

Understanding UplinkFast

Switches in hierarchical networks can be grouped into backbone switches, distribution switches, and access switches. Figure 17-2 shows a complex network where distribution switches and access switches each have at least one redundant link that spanning tree blocks to prevent loops.

Figure 17-2 Switches in a Hierarchical Network



If a switch loses connectivity, it begins using the alternate paths as soon as the spanning tree selects a new root port. By enabling UplinkFast with the **spanning-tree uplinkfast** global configuration command, you can accelerate the choice of a new root port when a link or switch fails or when the spanning tree reconfigures itself. The root port transitions to the forwarding state immediately without going through the listening and learning states, as it would with the normal spanning-tree procedures.

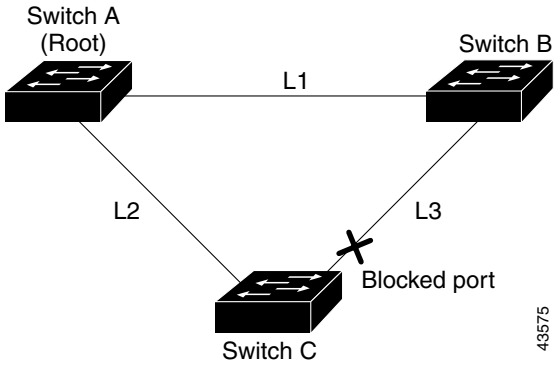
When the spanning tree reconfigures the new root port, other interfaces flood the network with multicast packets, one for each address that was learned on the interface. You can limit these bursts of multicast traffic by reducing the max-update-rate parameter (the default for this parameter is 150 packets per second). However, if you enter zero, station-learning frames are not generated, so the spanning-tree topology converges more slowly after a loss of connectivity.

Note UplinkFast is most useful in wiring-closet switches at the access or edge of the network. It is not appropriate for backbone devices. This feature might not be useful for other types of applications.

UplinkFast provides fast convergence after a direct link failure and achieves load balancing between redundant Layer 2 links using uplink groups. An uplink group is a set of Layer 2 interfaces (per VLAN), only one of which is forwarding at any given time. Specifically, an uplink group consists of the root port (which is forwarding) and a set of blocked ports, except for self-looping ports. The uplink group provides an alternate path in case the currently forwarding link fails.

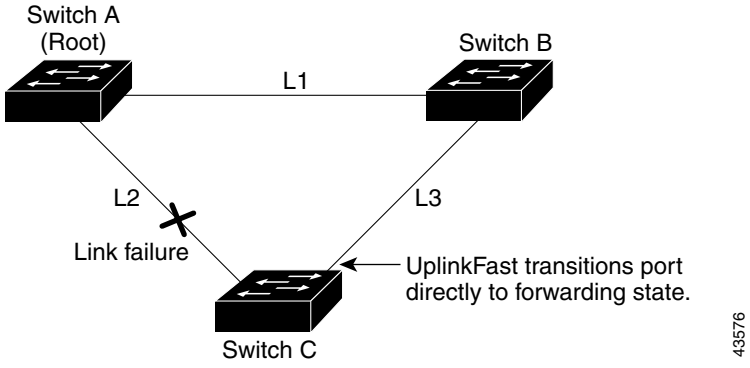
Figure 17-3 shows an example topology with no link failures. Switch A, the root switch, is connected directly to Switch B over link L1 and to Switch C over link L2. The Layer 2 interface on Switch C that is connected directly to Switch B is in a blocking state.

Figure 17-3 UplinkFast Example Before Direct Link Failure



If Switch C detects a link failure on the currently active link L2 on the root port (a *direct* link failure), UplinkFast unblocks the blocked port on Switch C and transitions it to the forwarding state without going through the listening and learning states, as shown in Figure 17-4. This change takes approximately 1 to 5 seconds.

Figure 17-4 UplinkFast Example After Direct Link Failure



Understanding Cross-Stack UplinkFast

Cross-stack UplinkFast (CSUF) provides a fast spanning-tree transition (fast convergence in less than 1 second under normal network conditions) across a stack of switches that use the GigaStack GBIC modules connected in a shared cascaded configuration (multidrop backbone). During the fast transition, an alternate redundant link on the stack of switches is placed in the forwarding state without causing temporary spanning-tree loops or loss of connectivity to the backbone. With this feature, you can have a redundant and resilient network in some configurations. You enable CSUF by using the **spanning-tree stack-port** interface configuration command.

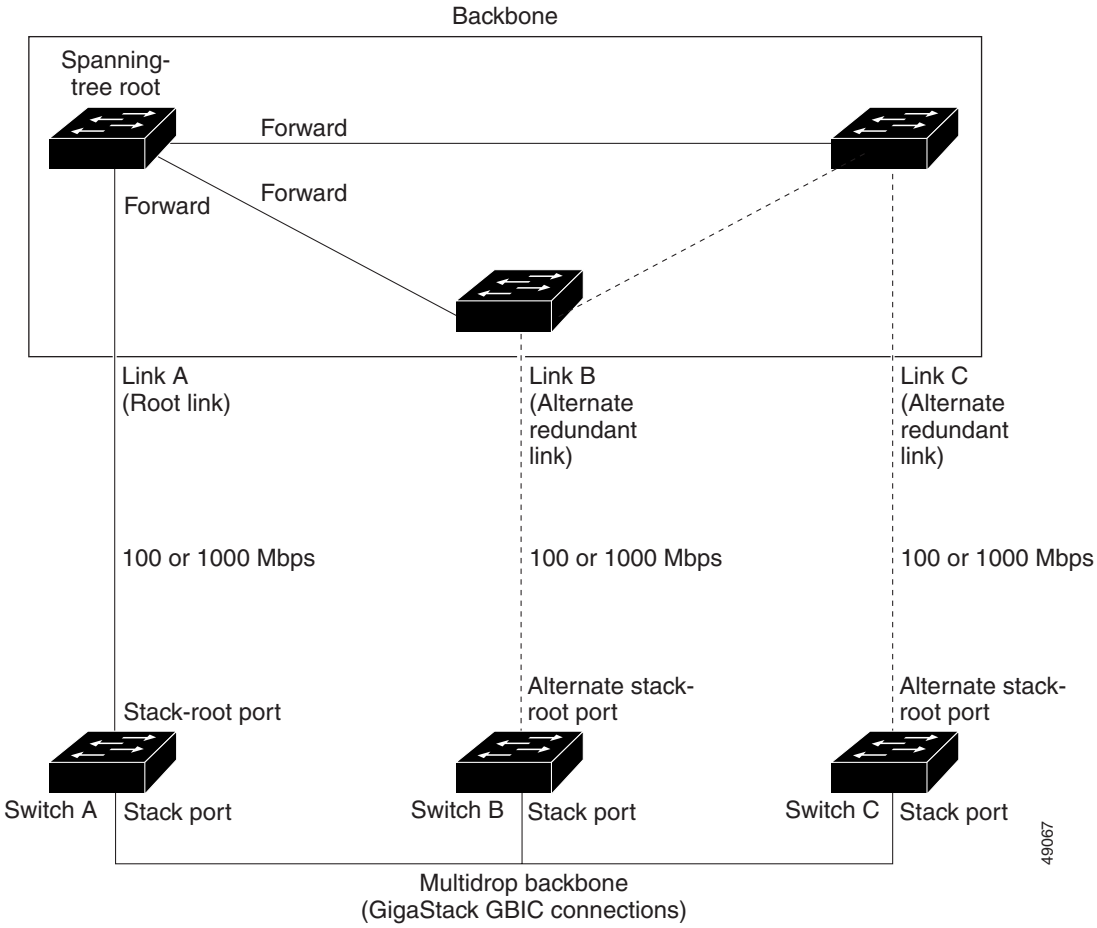
CSUF might not provide a fast transition all the time; in these cases, the normal spanning-tree transition occurs, completing in 30 to 40 seconds. For more information, see the "Events that Cause Fast Convergence" section on page 17-7.

How CSUF Works

CSUF ensures that one link in the stack is elected as the path to the root. As shown in Figure 17-5, Switches A, B, and C are cascaded through the GigaStack GBIC module to form a multidrop backbone, which communicates control and data traffic across the switches at the access layer. The switches in the stack use their stack ports to communicate with each other and to connect to the stack backbone; stack ports are always in the spanning-tree forwarding state. The stack-root port on Switch A provides the path to the root of the spanning tree; the alternate stack-root ports on Switches B and C can provide an alternate path to the spanning-tree root if the current stack-root switch fails or if its link to the spanning-tree root fails.

Link A, the root link, is in the spanning-tree forwarding state; Links B and C are alternate redundant links that are in the spanning-tree blocking state. If Switch A fails, if its stack-root port fails, or if Link A fails, CSUF selects either the Switch B or Switch C alternate stack-root port and puts it into the forwarding state in less than 1 second.

Figure 17-5 Cross-Stack UplinkFast Topology



CSUF uses the Stack Membership Discovery Protocol to build a neighbor list of stack members through the receipt of discovery hello packets. When certain link loss or spanning-tree events occur (described in "Events that Cause Fast Convergence" section on page 17-7), the Fast Uplink Transition Protocol uses the neighbor list to send fast-transition requests on the stack port to stack members.

The switch sending the fast-transition request needs to do a fast transition to the forwarding state of a port that it has chosen as the root port, and it must obtain an acknowledgement from each stack switch before performing the fast transition.

Each switch in the stack determines if the sending switch is a better choice than itself to be the stack root of this spanning-tree instance by comparing the root, cost, and bridge ID. If the sending switch is the best choice as the stack root, each switch in the stack returns an acknowledgement; otherwise, it does not respond to the sending switch (drops the packet). The sending switch then has not received acknowledgements from all stack switches.

When acknowledgements are received from all stack switches, the Fast Uplink Transition Protocol on the sending switch immediately transitions its alternate stack-root port to the forwarding state. If acknowledgements from all stack switches are not obtained by the sending switch, the normal spanning-tree transitions (blocking, listening, learning, and forwarding) take place, and the spanning-tree topology converges at its normal rate (2 * forward-delay time + max-age time).

The Fast Uplink Transition Protocol is implemented on a per-VLAN basis and affects only one spanning-tree instance at a time.

Events that Cause Fast Convergence

Depending on the network event or failure, the CSUF fast convergence might or might not occur.

Fast convergence (less than 1 second under normal network conditions) occurs under these circumstances:

- The stack-root port link fails.

  If two switches in the stack have alternate paths to the root, only one of the switches performs the fast transition.
- The failed link, which connects the stack root to the spanning-tree root, recovers.
- A network reconfiguration causes a new stack-root switch to be selected.
- A network reconfiguration causes a new port on the current stack-root switch to be chosen as the stack-root port.

Note The fast transition might not occur if multiple events occur simultaneously. For example, if a stack member switch is powered off, and at the same time, the link connecting the stack root to the spanning-tree root comes back up, the normal spanning-tree convergence occurs.

Normal spanning-tree convergence (30 to 40 seconds) occurs under these conditions:

- The stack-root switch is powered off, or the software failed.
- The stack-root switch, which was powered off or failed, is powered on.
- A new switch, which might become the stack root, is added to the stack.
- A switch other than the stack root is powered off or failed.
- A link fails between stack ports on the multidrop backbone.

Limitations

These limitations apply to CSUF:

- CSUF uses the GigaStack GBIC module and runs on all Catalyst 3550 switches, all Catalyst 3500 XL switches, Catalyst 2950 switches with GBIC module slots, and only on modular Catalyst 2900 XL switches that have the 1000BASE-X module installed.
- Up to nine stack switches can be connected through their stack ports to the multidrop backbone. Only one stack port per switch is supported.
- Each stack switch can be connected to the spanning-tree backbone through one uplink.
- If the stack consists of a mixture of Catalyst 3550, Catalyst 3500 XL, Catalyst 2950, and Catalyst 2900 XL switches, up to 64 VLANs with spanning tree enabled are supported. If the stack consists of only Catalyst 3550 switches, up to 128 VLANs with spanning tree enabled are supported.

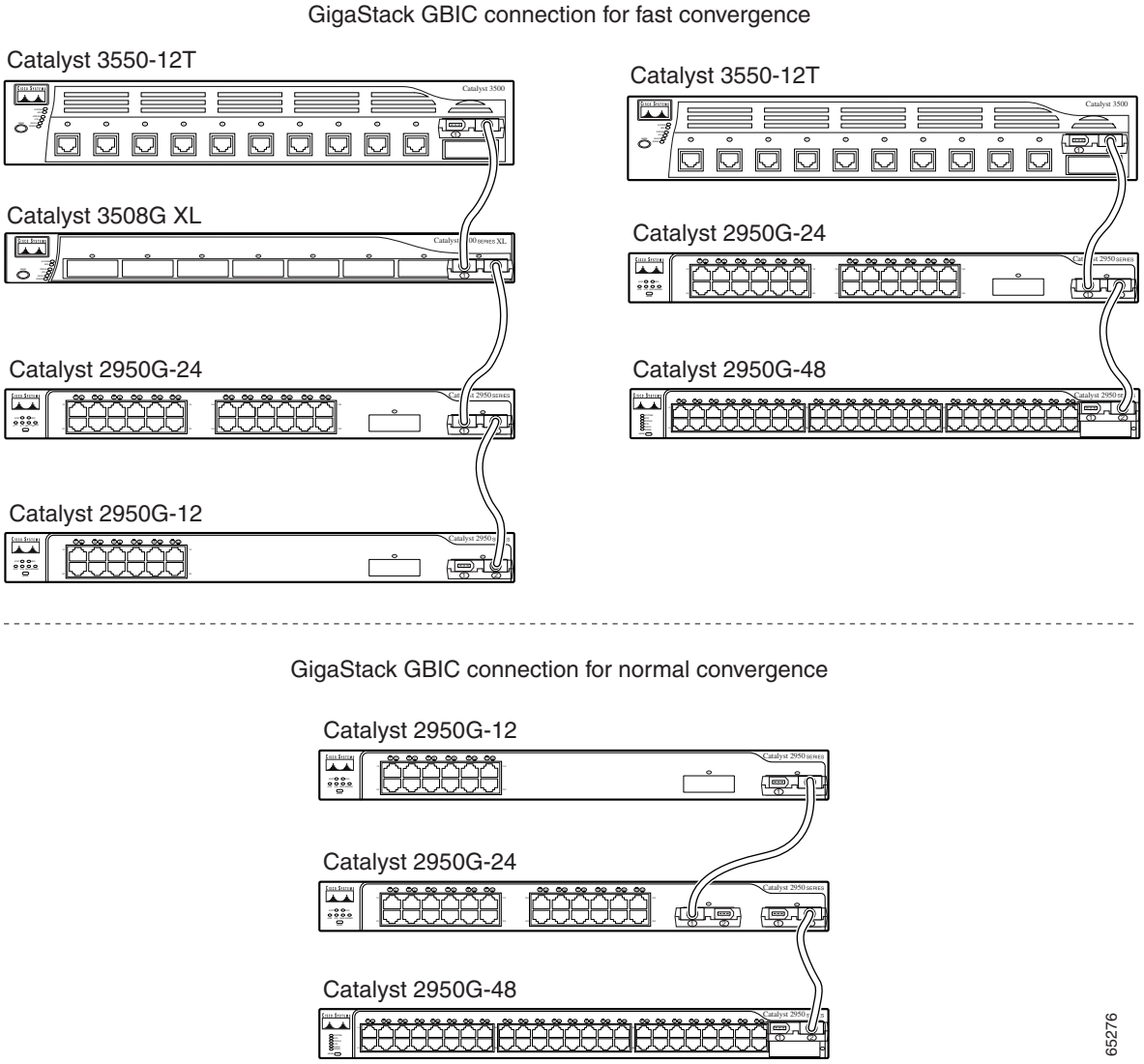
Connecting the Stack Ports

A fast transition occurs across the stack of switches if the multidrop backbone connections are a continuous link from one GigaStack GBIC module to another as shown in the top half of Figure 17-6. The bottom half of Figure 17-6 shows how to connect the GigaStack GBIC module to achieve a normal convergence time.

You should follow these guidelines:

- A switch supports only one stack port.
- Do not connect alternate stack-root ports to stack ports.
- Connect all stack ports on the switch stack to the multidrop backbone.
- You can connect the open ports on the top and bottom GigaStack GBIC modules within the same stack to form a redundant link.

Figure 17-6 GigaStack GBIC Module Connections and Spanning-Tree Convergence



Understanding BackboneFast

BackboneFast detects indirect failures in the core of the backbone. BackboneFast is a complementary technology to the UplinkFast feature, which responds to failures on links directly connected to access switches. BackboneFast optimizes the maximum-age timer, which determines the amount of time the switch stores protocol information received on an interface. When a switch receives an inferior BPDU from the designated port of another switch, the BPDU is a signal that the other switch might have lost its path to the root, and BackboneFast tries to find an alternate path to the root.

BackboneFast, which is enabled by using the **spanning-tree backbonefast** global configuration command, starts when a root port or blocked port on a switch receives inferior BPDUs from its designated switch. An inferior BPDU identifies a switch that declares itself as both the root bridge and the designated switch. When a switch receives an inferior BPDU, it means that a link to which the switch is not directly connected (an *indirect* link) has failed (that is, the designated bridge has lost its connection

to the root switch). Under spanning-tree rules, the switch ignores inferior BPDUs for the configured maximum aging time specified by the **spanning-tree vlan** *vlan-id* **max-age** global configuration command.

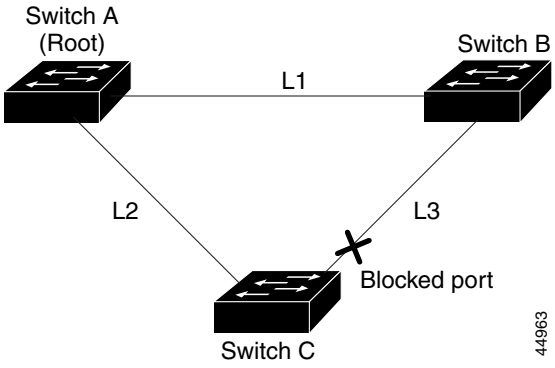
The switch tries to determine if it has an alternate path to the root switch. If the inferior BPDU arrives on a blocked port, the root port and other blocked ports on the switch become alternate paths to the root switch. (Self-looped ports are not considered alternate paths to the root switch.) If the inferior BPDU arrives on the root port, all blocked ports become alternate paths to the root switch. If the inferior BPDU arrives on the root port and there are no blocked ports, the switch assumes that it has lost connectivity to the root switch, causes the maximum aging time on the root port to expire, and becomes the root switch according to normal spanning-tree rules.

If the switch has alternate paths to the root switch, it uses these alternate paths to send a root link query (RLQ) request. The switch sends the RLQ request on all alternate paths to the root switch and waits for an RLQ reply from other switches in the network.

If the switch determines that it still has an alternate path to the root, it expires the maximum aging time on the port that received the inferior BPDU. If all the alternate paths to the root switch indicate that the switch has lost connectivity to the root switch, the switch expires the maximum aging time on the port that received the RLQ reply. If one or more alternate paths can still connect to the root switch, the switch makes all ports on which it received an inferior BPDU its designated ports and moves them from the blocking state (if they were in the blocking state), through the listening and learning states, and into the forwarding state.

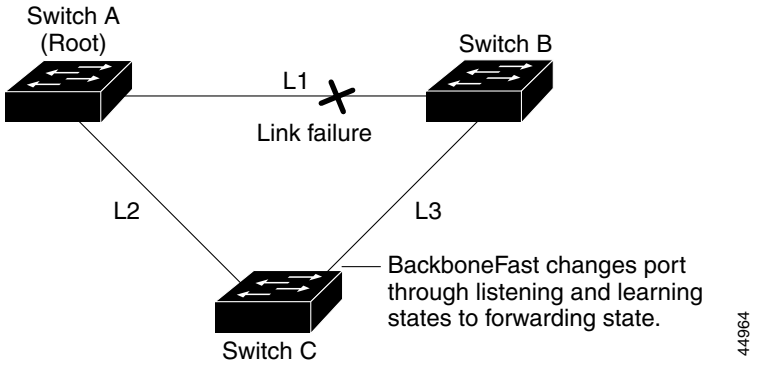
Figure 17-7 shows an example topology with no link failures. Switch A, the root switch, connects directly to Switch B over link L1 and to Switch C over link L2. The Layer 2 interface on Switch C that connects directly to Switch B is in the blocking state.

Figure 17-7 BackboneFast Example Before Indirect Link Failure



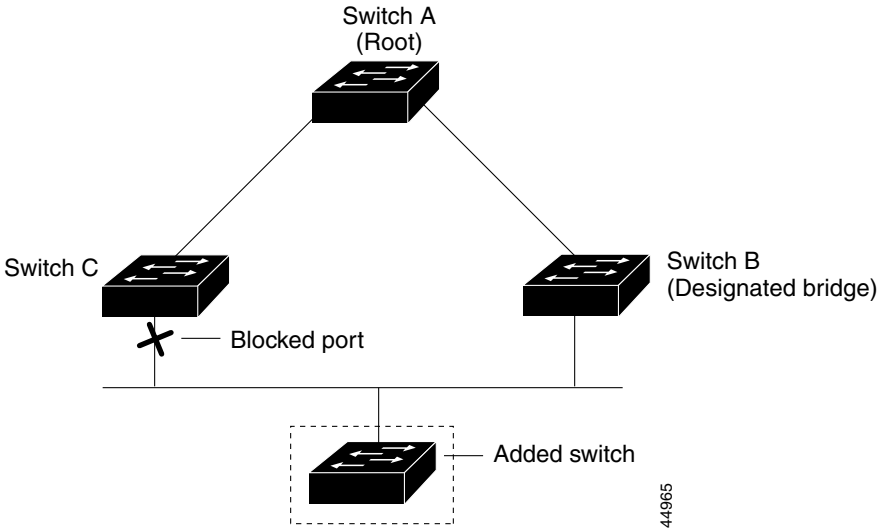
If link L1 fails as shown in Figure 17-8, Switch C cannot detect this failure because it is not connected directly to link L1. However, because Switch B is directly connected to the root switch over L1, it detects the failure, elects itself the root, and begins sending BPDUs to Switch C, identifying itself as the root. When Switch C receives the inferior BPDUs from Switch B, Switch C assumes that an indirect failure has occurred. At that point, BackboneFast allows the blocked port on Switch C to move immediately to the listening state without waiting for the maximum aging time for the port to expire. BackboneFast then transitions the Layer 2 interface on Switch C to the forwarding state, providing a path from Switch B to Switch A. This switchover takes approximately 30 seconds, twice the Forward Delay time if the default Forward Delay time of 15 seconds is set. Figure 17-8 shows how BackboneFast reconfigures the topology to account for the failure of link L1.

Figure 17-8 BackboneFast Example After Indirect Link Failure



If a new switch is introduced into a shared-medium topology as shown in Figure 17-9, BackboneFast is not activated because the inferior BPDUs did not come from the recognized designated bridge (Switch B). The new switch begins sending inferior BPDUs that indicate it is the root switch. However, the other switches ignore these inferior BPDUs, and the new switch learns that Switch B is the designated bridge to Switch A, the root switch.

Figure 17-9 Adding a Switch in a Shared-Medium Topology



Understanding EtherChannel Guard

You can use EtherChannel guard to detect an EtherChannel misconfiguration between the switch and a connected device. A misconfiguration can occur if the switch interfaces are configured in an EtherChannel, but the interfaces on the other device are not. A misconfiguration can also occur if the channel parameters are not the same at both ends of the EtherChannel. For EtherChannel configuration guidelines, see the "EtherChannel Configuration Guidelines" section on page 30-8.

If the switch detects a misconfiguration on the other device, EtherChannel guard places the switch interfaces in the error-disabled state, and this error message appears:

```
PM-4-ERR_DISABLE: Channel-misconfig error detected on [chars], putting [chars] in
err-disable state.
```

You can enable this feature by using the **spanning-tree etherchannel guard misconfig** global configuration command.

Understanding Root Guard

The Layer 2 network of a service provider (SP) can include many connections to switches that are not owned by the SP. In such a topology, the spanning tree can reconfigure itself and select a *customer switch* as the root switch, as shown in Figure 17-10. You can avoid this situation by enabling root guard on SP switch interfaces that connect to switches in your customer's network. If spanning-tree calculations cause an interface in the customer network to be selected as the root port, root guard then places the interface in the root-inconsistent (blocked) state to prevent the customer's switch from becoming the root switch or being in the path to the root.

If a switch outside the SP network becomes the root switch, the interface is blocked (root-inconsistent state), and spanning tree selects a new root switch. The customer's switch does not become the root switch and is not in the path to the root.

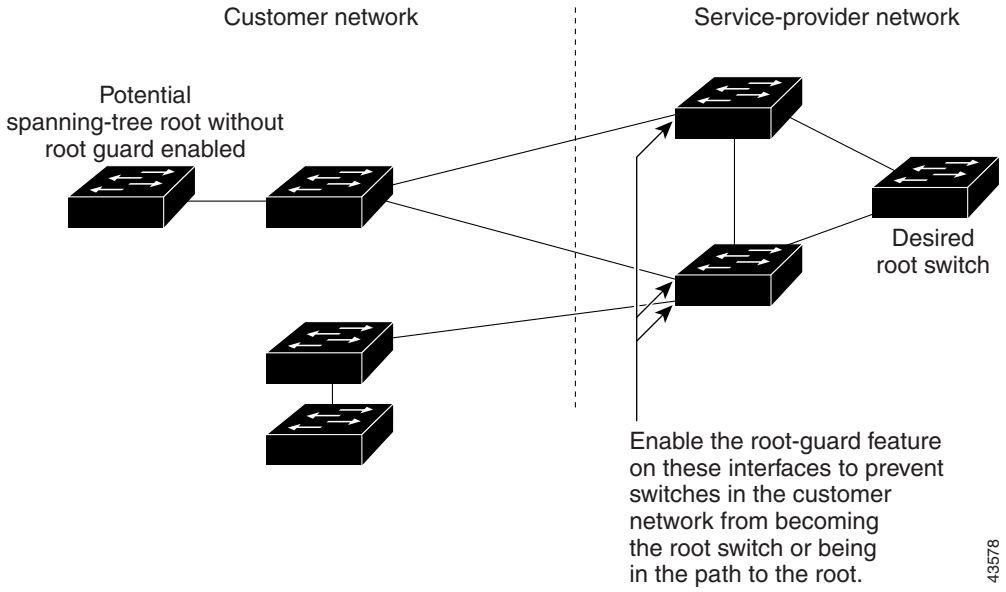
If the switch is operating in multiple spanning-tree (MST) mode, root guard forces the port to be a designated port. If a boundary port is blocked in an internal spanning-tree (IST) instance because of root guard, the port also is blocked in all MST instances. A boundary port is a port that connects to a LAN, the designated switch of which is either an IEEE 802.1D switch or a switch with a different MST region configuration.

Root guard enabled on an interface applies to all the VLANs to which the interface belongs. VLANs can be grouped and mapped to an MST instance.

You can enable this feature by using the **spanning-tree guard root** interface configuration command.

Caution Misuse of the root-guard feature can cause a loss of connectivity.

Figure 17-10 Root Guard in a Service-Provider Network



Understanding Loop Guard

You can use loop guard to prevent alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link. This feature is most effective when it is configured on the entire switched network.

You can enable this feature by using the **spanning-tree loopguard default** global configuration command.

When the switch is operating in PVST+ or rapid-PVST+ mode, loop guard prevents alternate and root ports from becoming designated ports, and spanning tree does not send BPDUs on root or alternate ports.

When the switch is operating in MST mode, BPDUs are not sent on nonboundary ports only if the port is blocked by loop guard in all MST instances. On a boundary port, loop guard blocks the port in all MST instances.

Configuring Optional Spanning-Tree Features

These sections describe how to configure optional spanning-tree features:

- Default Optional Spanning-Tree Configuration, page 17-14
- Optional Spanning-Tree Configuration Guidelines, page 17-14
- Enabling Port Fast, page 17-14 (optional)
- Enabling BPDU Guard, page 17-15 (optional)
- Enabling BPDU Filtering, page 17-16 (optional)
- Enabling UplinkFast for Use with Redundant Links, page 17-17 (optional)
- Enabling Cross-Stack UplinkFast, page 17-18 (optional)

- Enabling BackboneFast, page 17-19 (optional)
- Enabling EtherChannel Guard, page 17-20 (optional)
- Enabling Root Guard, page 17-20 (optional)
- Enabling Loop Guard, page 17-21 (optional)

Default Optional Spanning-Tree Configuration

Table 17-1 shows the default optional spanning-tree configuration.

Table 17-1 Default Optional Spanning-Tree Configuration

Feature	Default Setting
Port Fast, BPDU filtering, BPDU guard	Globally disabled (unless they are individually configured per interface).
UplinkFast	Globally disabled.
CSUF	Disabled on all interfaces.
BackboneFast	Globally disabled.
EtherChannel guard	Globally enabled.
Root guard	Disabled on all interfaces.
Loop guard	Disabled on all interfaces.

Optional Spanning-Tree Configuration Guidelines

You can configure PortFast, BPDU guard, BPDU filtering, EtherChannel guard, root guard, or loop guard if your switch is running PVST+, rapid PVST+, or MSTP.

You can configure the UplinkFast, the BackboneFast, or the cross-stack UplinkFast feature for rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

Enabling Port Fast

A port with the Port Fast feature enabled is moved directly to the spanning-tree forwarding state without waiting for the standard forward-time delay.

Caution Use Port Fast *only* when connecting a single end station to an access or trunk port. Enabling this feature on a port connected to a switch or hub could prevent spanning tree from detecting and disabling loops in your network, which could cause broadcast storms and address-learning problems.

If you enable the voice VLAN feature, the Port Fast feature is automatically enabled. When you disable voice VLAN, the Port Fast feature is not automatically disabled. For more information, see Chapter 13, “Configuring Voice VLAN.”

You can enable this feature if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to enable Port Fast. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode.
Step 3	**spanning-tree portfast** [**trunk**]	Enable Port Fast on an access port connected to a single workstation or server. By specifying the **trunk** keyword, you can enable Port Fast on a trunk port. **Note** To enable Port Fast on trunk ports, you must use the **spanning-tree portfast trunk** interface configuration command. The **spanning-tree portfast** command will not work on trunk ports. **Caution** Make sure that there are no loops in the network between the trunk port and the workstation or server before you enable Port Fast on a trunk port. By default, Port Fast is disabled on all ports.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show spanning-tree interface** *interface-id* **portfast**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note You can use the **spanning-tree portfast default** global configuration command to globally enable the Port Fast feature on all nontrunking ports.

To disable the Port Fast feature, use the **spanning-tree portfast disable** interface configuration command.

Enabling BPDU Guard

When you globally enable BPDU guard on ports that are Port Fast-enabled (the ports are in a Port Fast-operational state), spanning tree continues to run on the ports. They remain up unless they receive a BPDU.

In a valid configuration, Port Fast-enabled ports do not receive BPDUs. Receiving a BPDU on a Port Fast-enabled port signals an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the port in the error-disabled state. The BPDU guard feature provides a secure response to invalid configurations because you must manually put the port back in service. Use the BPDU guard feature in a service-provider network to prevent an access port from participating in the spanning tree.

Caution Configure Port Fast only on ports that connect to end stations; otherwise, an accidental topology loop could cause a data packet loop and disrupt switch and network operation.

You can also use the **spanning-tree bpduguard enable** interface configuration command to enable BPDU guard on any port without also enabling the Port Fast feature. When the port receives a BPDU, it is put in the error-disabled state.

You can enable the BPDU guard feature if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to globally enable the BPDU guard feature. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree portfast bpduguard default**	Globally enable BPDU guard. By default, BPDU guard is disabled.
Step 3	**interface** *interface-id*	Specify the interface connected to an end station, and enter interface configuration mode.
Step 4	**spanning-tree portfast**	Enable the Port Fast feature.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable BPDU guard, use the **no spanning-tree portfast bpduguard default** global configuration command.

You can override the setting of the **no spanning-tree portfast bpduguard default** global configuration command by using the **spanning-tree bpduguard enable** interface configuration command.

Enabling BPDU Filtering

When you globally enable BPDU filtering on Port Fast-enabled ports, it prevents ports that are in a Port Fast-operational state from sending or receiving BPDUs. The ports still send a few BPDUs at link-up before the switch begins to filter outbound BPDUs. You should globally enable BPDU filtering on a switch so that hosts connected to these ports do not receive BPDUs. If a BPDU is received on a Port Fast-enabled port, the port loses its Port Fast-operational status, and BPDU filtering is disabled.

Caution Configure Port Fast only on ports that connect to end stations; otherwise, an accidental topology loop could cause a data packet loop and disrupt switch and network operation.

You can also use the **spanning-tree bpdufilter enable** interface configuration command to enable BPDU filtering on any port without also enabling the Port Fast feature. This command prevents the port from sending or receiving BPDUs.

Caution Enabling BPDU filtering on an interface is the same as disabling spanning tree on it and can result in spanning-tree loops.

You can enable the BPDU filtering feature if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to globally enable the BPDU filtering feature. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree portfast bpdufilter default**	Globally enable BPDU filtering. By default, BPDU filtering is disabled.
Step 3	**interface** *interface-id*	Specify the interface connected to an end station, and enter interface configuration mode.
Step 4	**spanning-tree portfast**	Enable the Port Fast feature.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable BPDU filtering, use the **no spanning-tree portfast bpdufilter default** global configuration command.

You can override the setting of the **no spanning-tree portfast bpdufilter default** global configuration command by using the **spanning-tree bpdufilter enable** interface configuration command.

Enabling UplinkFast for Use with Redundant Links

UplinkFast cannot be enabled on VLANs that have been configured for switch priority. To enable UplinkFast on a VLAN with switch priority configured, first restore the switch priority on the VLAN to the default value by using the **no spanning-tree vlan** *vlan-id* **priority** global configuration command.

Note When you enable UplinkFast, it affects all VLANs on the switch. You cannot configure UplinkFast on an individual VLAN.

You can enable the UplinkFast feature for rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

Beginning in privileged EXEC mode, follow these steps to enable UplinkFast. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree uplinkfast** [**max-update-rate** *pkts-per-second*]	Enable UplinkFast. (Optional) For *pkts-per-second*, the range is 0 to 32000 packets per second; the default is 150. If you set the rate to 0, station-learning frames are not generated, and the spanning-tree topology converges more slowly after a loss of connectivity.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree summary**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

When UplinkFast is enabled, the switch priority of all VLANs is set to 49152. If you change the path cost to a value less than 3000 and you enable UplinkFast or UplinkFast is already enabled, the path cost of all interfaces and VLAN trunks is increased by 3000 (if you change the path cost to 3000 or above, the path cost is not altered). The changes to the switch priority and the path cost reduces the chance that the switch will become the root switch.

When UplinkFast is disabled, the switch priorities of all VLANs and path costs of all interfaces are set to default values if you did not modify them from their defaults.

To return the update packet rate to the default setting, use the **no spanning-tree uplinkfast max-update-rate** global configuration command. To disable UplinkFast, use the **no spanning-tree uplinkfast** command.

Enabling Cross-Stack UplinkFast

Before enabling CSUF, make sure your stack switches are properly connected. For more information, see the "Connecting the Stack Ports" section on page 17-8.

You can enable the CSUF feature for rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

Beginning in privileged EXEC mode, follow these steps to enable CSUF. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree uplinkfast** [**max-update-rate** *pkts-per-second*]	Enable UplinkFast on the switch. (Optional) For **max-update-rate** *pkts-per-second*, specify the number of packets per second at which update packets are sent. The range is 0 to 65535; the default is 150 packets per second.
Step 3	**interface** *interface-id*	Specify the GBIC module interface on which to enable CSUF, and enter interface configuration mode.

	Command	Purpose
Step 4	**spanning-tree stack-port**	Enable CSUF on only one stack-port GBIC interface. The stack port connects to the GigaStack GBIC module multidrop backbone. If you try to enable CSUF on a Fast Ethernet or a Gigabit-capable Ethernet port, you receive an error message. If CSUF is already enabled on an interface and you try to enable it on another interface, you receive an error message. You must disable CSUF on the first interface before enabling it on a new interface. Use this command only on access switches.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable CSUF on an interface, use the **no spanning-tree stack-port** interface configuration command. To disable UplinkFast on the switch and all its VLANs, use the **no spanning-tree uplinkfast** global configuration command.

Enabling BackboneFast

You can enable BackboneFast to detect indirect link failures and to start the spanning-tree reconfiguration sooner.

Note If you use BackboneFast, you must enable it on all switches in the network. BackboneFast is not supported on Token Ring VLANs. This feature is supported for use with third-party switches.

You can enable the BackboneFast feature for rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

Beginning in privileged EXEC mode, follow these steps to enable BackboneFast. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree backbonefast**	Enable BackboneFast.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree summary**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the BackboneFast feature, use the **no spanning-tree backbonefast** global configuration command.

Enabling EtherChannel Guard

You can enable EtherChannel guard to detect an EtherChannel misconfiguration if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to enable EtherChannel guard. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**spanning-tree etherchannel guard misconfig**	Enable EtherChannel guard.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show spanning-tree summary**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the EtherChannel guard feature, use the **no spanning-tree etherchannel guard misconfig** global configuration command.

You can use the **show interfaces status err-disabled** privileged EXEC command to show which switch ports are disabled because of an EtherChannel misconfiguration. On the remote device, you can enter the **show etherchannel summary** privileged EXEC command to verify the EtherChannel configuration.

After the configuration is corrected, enter the **shutdown** and **no shutdown** interface configuration commands on the port-channel interfaces that were misconfigured.

Enabling Root Guard

Root guard enabled on an interface applies to all the VLANs to which the interface belongs.

Do not enable the root guard on interfaces to be used by the UplinkFast feature. With UplinkFast, the backup interfaces (in the blocked state) replace the root port in the case of a failure. However, if root guard is also enabled, all the backup interfaces used by the UplinkFast feature are placed in the root-inconsistent state (blocked) and are prevented from reaching the forwarding state.

Note You cannot enable both root guard and loop guard at the same time.

You can enable this feature if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to enable root guard on an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify an interface to configure, and enter interface configuration mode.
Step 3	**spanning-tree guard root**	Enable root guard on the interface. By default, root guard is disabled on all interfaces.

	Command	Purpose
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable root guard, use the **no spanning-tree guard** interface configuration command.

Enabling Loop Guard

You can use loop guard to prevent alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link. This feature is most effective when it is configured on the entire switched network. Loop guard operates only on ports that are considered point-to-point by the spanning tree.

Note You cannot enable both loop guard and root guard at the same time.

You can enable this feature if your switch is running PVST+, rapid PVST+, or MSTP.

Beginning in privileged EXEC mode, follow these steps to enable loop guard. This procedure is optional.

	Command	Purpose
Step 1	**show spanning-tree active** or **show spanning-tree mst**	Determine which ports are alternate or root ports.
Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**spanning-tree loopguard default**	Enable loop guard. By default, loop guard is disabled.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To globally disable loop guard, use the **no spanning-tree loopguard default** global configuration command. You can override the setting of the **no spanning-tree loopguard default** global configuration command by using the **spanning-tree guard loop** interface configuration command.

Displaying the Spanning-Tree Status

To display the spanning-tree status, use one or more of the privileged EXEC commands in Table 17-2:

Table 17-2 Commands for Displaying the Spanning-Tree Status

Command	Purpose
show spanning-tree active	Displays spanning-tree information on active interfaces only.
show spanning-tree detail	Displays a detailed summary of interface information.
show spanning-tree interface *interface-id*	Displays spanning-tree information for the specified interface.
show spanning-tree mst interface *interface-id*	Displays MST information for the specified interface.
show spanning-tree summary [**totals**]	Displays a summary of port states or displays the total lines of the spanning-tree state section.

You can clear spanning-tree counters by using the **clear spanning-tree** [**interface** *interface-id*] privileged EXEC command.

For information about other keywords for the **show spanning-tree** privileged EXEC command, see the command reference for this release.

CHAPTER 18

Configuring Flex Links and the MAC Address-Table Move Update Feature

This chapter describes how to configure Flex Links, a pair of interfaces on the Catalyst 3550 switch that provide a mutual backup. It also describes how to configure the MAC address-table move update feature, also referred to as the Flex Links bidirectional fast convergence feature. Unless otherwise noted, the term *switch* refers to a standalone switch.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

The chapter consists of these sections:

- Understanding Flex Links and the MAC Address-Table Move Update, page 18-1
- Configuring Flex Links and MAC Address-Table Move Update, page 18-4
- Monitoring Flex Links and the MAC Address-Table Move Update, page 18-9

Understanding Flex Links and the MAC Address-Table Move Update

This section contains this information:

- Flex Links, page 18-1
- MAC Address-Table Move Update, page 18-3

Flex Links

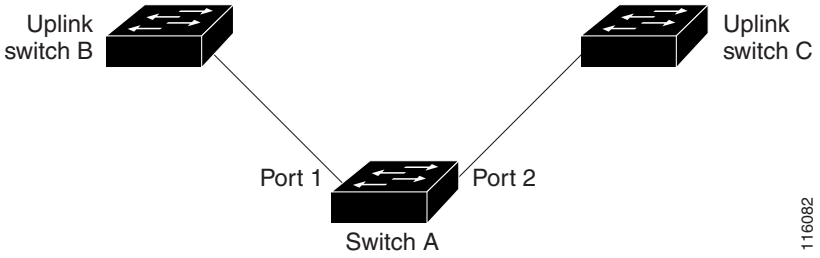
Flex Links are a pair of a Layer 2 interfaces (switch ports or port channels) where one interface is configured to act as a backup to the other. The feature provides an alternative solution to the Spanning Tree Protocol (STP). Users can disable STP and still retain basic link redundancy. Flex Links are typically configured in service provider or enterprise networks where customers do not want to run STP on the switch. If the switch is running STP, Flex Links is not necessary because STP already provides link-level redundancy or backup.

You configure Flex Links on one Layer 2 interface (the active link) by assigning another Layer 2 interface as the Flex Link or backup link. The Flex Link can be on the same switch. When one of the links is up and forwarding traffic, the other link is in standby mode, ready to begin forwarding traffic if the other link shuts down. At any given time, only one of the interfaces is in the linkup state and forwarding traffic. If the primary link shuts down, the standby link starts forwarding traffic. When the active link comes back up, it goes into standby mode and does not forward traffic. STP is disabled on Flex Link interfaces.

In Figure 18-1, ports 1 and 2 on switch A are connected to uplink switches B and C. Because they are configured as Flex Links, only one of the interfaces is forwarding traffic; the other is in standby mode. If port 1 is the active link, it begins forwarding traffic between port 1 and switch B; the link between port 2 (the backup link) and switch C is not forwarding traffic. If port 1 goes down, port 2 comes up and starts forwarding traffic to switch C. When port 1 comes back up, it goes into standby mode and does not forward traffic; port 2 continues forwarding traffic.

Optionally, you can configure a preemption mechanism, specifying the preferred port for forwarding traffic. For example, you can configure the above flexlink pair with preemption mode so that once port 1 comes back up in the above scenario, if it has greater bandwidth than port 2, port 1 will go forwarding after 60 seconds and port 2 will become standby. This is done by entering the preemption mode bandwidth and delay commands.

Figure 18-1 Flex Links Configuration Example



If a primary (forwarding) link goes down, a trap notifies the network management stations. If the standby link goes down, a trap notifies the users.

Flex Links are supported only on Layer 2 ports and port channels, not on VLANs or on Layer 3 ports.

MAC Address-Table Move Update

The MAC address-table move update feature allows the switch to provide rapid bidirectional convergence when a primary (forwarding) link goes down and the standby link begins forwarding traffic.

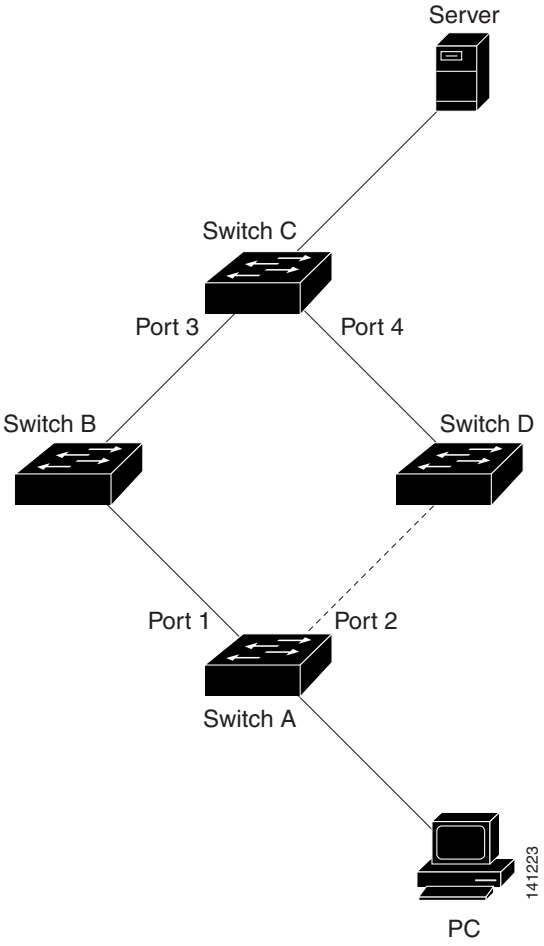
In Figure 18-2, switch A is an access switch, and ports 1 and 2 on switch A are connected to uplink switches B and D through a Flex Link pair. Port 1 is forwarding traffic, and port 2 is in the backup state. Traffic from the PC to the server is forwarded from port 1 to port 3. The MAC address of the PC has been learned on port 3 of switch C. Traffic from the server to the PC is forwarded from port 3 to port 1.

If the MAC address-table move update feature is not configured and port 1 goes down, port 2 starts forwarding traffic. However, for a short time, switch C keeps forwarding traffic from the server to the PC through port 3, and the PC does not get the traffic because port 1 is down. If switch C removes the MAC address of the PC on port 3 and relearns it on port 4, traffic can then be forwarded from the server to the PC through port 2.

If the MAC address-table move update feature is configured and enabled on the switches in Figure 18-2 and port 1 goes down, port 2 starts forwarding traffic from the PC to the server. The switch sends a MAC address-table move update packet from port 2. Switch C gets this packet on port 4 and immediately learns the MAC address of the PC on port 4, which reduces the reconvergence time.

You can configure the access switch, switch A, to *send* MAC address-table move update messages. You can also configure the uplink switches B, C, and D to *get* and process the MAC address-table move update messages. When switch C gets a MAC address-table move update message from switch A, switch C learns the MAC address of the PC on port 4. Switch C updates the MAC address table, including the forwarding table entry for the PC. Switch A does not need to wait for the MAC address-table update. The switch detects a failure on port 1 and immediately starts forwarding server traffic from port 2, the new forwarding port. This change occurs in 100 milliseconds (ms). The PC is directly connected to switch A, and the connection status does not change. Switch A does not need to update the PC entry in the MAC address table.

Figure 18-2 MAC Address-Table Move Update Example



Configuring Flex Links and MAC Address-Table Move Update

These sections contain this information:

- Configuration Guidelines, page 18-4
- Default Configuration, page 18-5

Configuration Guidelines

Follow these guidelines to configure Flex Links:

- You can configure only one Flex Link backup link for any active link, and it must be a different interface from the active interface.
- An interface can belong to only one Flex Link pair. An interface can be a backup link for only one active link. An active link cannot belong to another Flex Link pair.

- Neither of the links can be a port that belongs to an EtherChannel. However, you can configure two port channels (EtherChannel logical interfaces) as Flex Links, and you can configure a port channel and a physical interface as Flex Links, with either the port channel or the physical interface as the active link.
- A backup link does not have to be the same type (Fast Ethernet, Gigabit Ethernet, or port channel) as the active link. However, you should configure both Flex Links with similar characteristics so that there are no loops or changes in behavior if the standby link begins to forward traffic.
- STP is disabled on Flex Link ports. A Flex Link port does not participate in STP, even if the VLANs present on the port are configured for STP. When STP is not enabled, be sure that there are no loops in the configured topology.

Follow these guidelines to configure MAC address-table move update feature:

- You can enable and configure this feature on the access switch to *send* the MAC address-table move updates.
- You can enable and configure this feature on the uplink switches to *get* the MAC address-table move updates.

Default Configuration

The Flex Links are not configured, and there are no backup interfaces defined.

The preemption mode is OFF.

The preemption delay is 35 seconds.

The MAC address-table move update feature is not configured on the switch.

Configuring Flex Links and MAC Address-Table Move Update

This section contains this information:

- Configuring Flex Links, page 18-5
- Configuring the MAC Address-Table Move Update Feature, page 18-7

Configuring Flex Links

Beginning in privileged EXEC mode, follow these steps to configure a pair of Flex Links:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface, and enter interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 48.
Step 3	**switchport backup interface** *interface-id*	Configure a physical Layer 2 interface (or port channel) as part of a Flex Link pair with the interface. When one link is forwarding traffic, the other interface is in standby mode.

	Command	Purpose
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interface** [*interface-id*] **switchport backup**	Verify the configuration.
Step 6	**copy running-config startup config**	(Optional) Save your entries in the switch startup configuration file.

This example shows how to configure an interface with a backup interface and to verify the configuration:

```
Switch# configure terminal
Switch(conf)# interface gigabitethernet0/21
Switch(conf-if)# switchport backup interface gigabitethernet0/22
Switch(conf-if)# end
Switch# show interface switchport backup
Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------------------
FastEthernet1/0/1         FastEthernet1/0/2         Active Up/Backup Standby
FastEthernet1/0/3         FastEthernet2/0/4         Active Up/Backup Standby
Port-channel1           GigabitEthernet7/0/1      Active Up/Backup Standby
GigabitEthernet0/21      GigabitEthernet0/22     Active Up/Backup Standby
GigabitEthernet0/3      GigabitEthernet0/4     Active Up/Backup Standby
Port-channel1           GigabitEthernet0/5     Active Up/Backup Standby
```

Beginning in Interface Configuration mode, follow these steps to configure a preemption scheme for a pair of Flex Links:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface, and enter interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 48.
Step 3	**switchport backup interface** *interface-id*	Configure a physical Layer 2 interface (or port channel) as part of a Flex Link pair with the interface. When one link is forwarding traffic, the other interface is in standby mode.
Step 4	**switchport backup interface** *interface-id* **preemption mode** [**forced** | **bandwidth** | **off**]	Configure a preemption mechanism and delay for a Flex link interface pair. You can configure the preemption as: • forced - the active interface always preempts the backup • bandwidth - the interface with higher bandwidth always acts as the active interface • off - no preemption happens from active to backup

	Command	Purpose
Step 5	**switchport backup interface** *interface-id* **preemption delay** *delay-time*	Configure the delay time until a port preempts another port. **Note** Setting a delay time only works with forced and bandwidth modes.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interface** [*interface-id*] **switchport backup**	Verify the configuration.
Step 8	**copy running-config startup config**	(Optional) Save your entries in the switch startup configuration file.

This example shows how to configure preemption mode as bandwidth, for a backup interface pair and to verify the configuration:

```
Switch# configure terminal
Switch(conf)# interface gigabitethernet0/1
Switch(conf-if)# switchport backup interface gigabitethernet0/2 preemption mode forced
Switch(conf-if)# switchport backup interface gigabitethernet0/2 preemption delay 50
Switch(conf-if)# end

Switch# show interface switchport backup detail
Active Interface Backup Interface State
------------------------------------------------------------------------
GigabitEthernet0/21 GigabitEthernet0/2 Active Up/Backup Standby
Interface Pair : Gi0/1, Gi0/2
Preemption Mode : forced
Preemption Delay : 50 seconds
Bandwidth : 100000 Kbit (Gi0/1), 100000 Kbit (Gi0/2)
Mac Address Move Update Vlan : auto
```

Configuring the MAC Address-Table Move Update Feature

This section contains this information:

- Configuring a switch to send MAC address-table move updates
- Configuring a switch to get MAC address-table move updates

Beginning in privileged EXEC mode, follow these steps to configure an access switch to send MAC address-table move updates:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface, and enter interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 48.

	Command	Purpose
Step 3	**switchport backup interface** *interface-id* or **switchport backup interface** *interface-id* **mmu primary vlan** *vlan-id*	Configure a physical Layer 2 interface (or port channel), as part of a Flex Link pair with the interface. The MAC address-table move update VLAN is the lowest VLAN ID on the interface. Configure a physical Layer 2 interface (or port channel) and specify the VLAN ID on the interface, which is used for sending the MAC address-table move update. When one link is forwarding traffic, the other interface is in standby mode.
Step 4	**end**	Return to global configuration mode.
Step 5	**mac address-table move update transmit**	Enable the access switch to send MAC address-table move updates to other switches in the network if the primary link goes down and the switch starts forwarding traffic through the standby link.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show mac address-table move update**	Verify the configuration.
Step 8	**copy running-config startup config**	(Optional) Save your entries in the switch startup configuration file.

To disable the MAC address-table move update feature on the access switch, use the **no mac address-table move update transmit** interface configuration command. To display the MAC address-table move update information, use the **show mac address-table move update** privileged EXEC command.

This example shows how to configure an access switch to send MAC address-table move update messages:

```
Switch# configure terminal
Switch(conf)# interface gigabitethernet0/21
Switch(conf-if)# switchport backup interface fastethernet1/0/2
Switch(conf-if)# switchport backup interface gigabitethernet0/22 mmu primary vlan 2
Switch(conf-if)# end
Switch(conf)# mac address-table move update transmit
Switch(conf)# end
```

Verify the configuration as shown in the following example:

```
Switch# show mac-address-table move update
Switch-ID : 01d0.2bfc.3180
Dst mac-address : 0180.c200.0010
Vlans/Macs supported : 1023/8320
Default/Current settings: Rcv Off/Off, Xmt Off/Off
Max packets per min : Rcv 40, Xmt 60
Rcv packet count : 0
Rcv conforming packet count : 0
Rcv invalid packet count : 0
Rcv packet count this min : 0
Rcv threshold exceed count : 0
Rcv last sequence# this min : 0
Rcv last interface : None
Rcv last src-mac-address : 0000.0000.0000
Rcv last switch-ID : 0000.0000.0000
Xmt packet count : 0
Xmt packet count this min : 0
```

```
Xmt threshold exceed count : 0
Xmt pak buf unavail cnt : 0
Xmt last interface : None
```

Beginning in privileged EXEC mode, follow these steps to configure a switch to get and process MAC address-table move update messages:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mac address-table move update receive**	Enable the switch to get and process the MAC address-table move updates.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mac address-table move update**	Verify the configuration.
Step 5	**copy running-config startup config**	(Optional) Save your entries in the switch startup configuration file.

To disable the MAC address-table move update feature on the access switch, use the **no mac address-table move update receive** configuration command. To display the MAC address-table move update information, use the **show mac address-table move update** privileged EXEC command.

This example shows how to configure a switch to get and process MAC address-table move update messages:

```
Switch# configure terminal
Switch(conf)# mac address-table move update receive
Switch(conf)# end
```

Monitoring Flex Links and the MAC Address-Table Move Update

Table 18-1 shows the privileged EXEC commands for monitoring the Flex Links configuration and the MAC address-table move update information.

Table 18-1 Flex Links and MAC Address-Table Move Update Monitoring Commands

Command	Purpose
show interface [*interface-id*] **switchport backup**	Displays the Flex Link backup interface configured for an interface or all the configured Flex Links and the state of each active and backup interface (up or standby mode).
show mac address-table move update	Displays the MAC address-table move update information on the switch.

CHAPTER 18

Configuring DHCP Features and IP Source Guard

This chapter describes how to configure DHCP snooping and the option-82 data insertion features on the Catalyst 3550 switch. It also describes how to configure the IP source guard feature.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release, and see the "DHCP Commands" section in the *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2*.

This chapter consists of these sections:

- Understanding DHCP Features, page 18-1
- Configuring DHCP Features, page 18-8
- Displaying DHCP Information, page 18-18
- Understanding IP Source Guard, page 18-18
- Configuring IP Source Guard, page 18-19
- Displaying IP Source Guard Information, page 18-22

Understanding DHCP Features

DHCP is widely used in LAN environments to dynamically assign host IP addresses from a centralized server, which significantly reduces the overhead of administration of IP addresses. DHCP also helps conserve the limited IP address space because IP addresses no longer need to be permanently assigned to hosts; only those hosts that are connected to the network consume IP addresses.

The switch supports these DHCP features:

- DHCP Server, page 18-2
- DHCP Relay Agent, page 18-2
- DHCP Snooping, page 18-2
- Option-82 Data Insertion, page 18-3
- Cisco IOS DHCP Server Database, page 18-7
- DHCP Snooping Binding Database, page 18-7

For information about the DHCP client, see the "*Configuring DHCP*" section of the "*IP Addressing and Services*" section of the *Cisco IOS IP Configuration Guide, Release 12.2*.

DHCP Server

The DHCP server assigns IP addresses from specified address pools on a switch or router to DHCP clients and manages them. If the DHCP server cannot give the DHCP client the requested configuration parameters from its database, it forwards the request to one or more secondary DHCP servers defined by the network administrator.

DHCP Relay Agent

A DHCP relay agent is a Layer 3 device that forwards DHCP packets between clients and servers. Relay agents forward requests and replies between clients and servers when they are not on the same physical subnet. Relay agent forwarding is different from the normal Layer 2 forwarding, in which IP datagrams are switched transparently between networks. Relay agents receive DHCP messages and generate new DHCP messages to send on egress interfaces.

DHCP Snooping

DHCP snooping is a DHCP security feature that provides network security by filtering untrusted DHCP messages and by building and maintaining a DHCP snooping binding database, which is also referred to as a DHCP snooping binding table.

DHCP snooping acts like a firewall between untrusted hosts and DHCP servers. You can use DHCP snooping to differentiate between untrusted interfaces connected to the end user and trusted interfaces connected to the DHCP server or another switch.

Note For DHCP snooping to function properly, all DHCP servers must be connected to the switch through trusted interfaces.

An untrusted message is a message that is received from outside the network or firewall. When you use DHCP snooping in a service-provider environment, an untrusted message is sent from a device that is not in the service-provider network, such as a customer's switch. Messages from unknown devices are untrusted because they can be sources of traffic attacks.

The DHCP snooping binding database contains the MAC address, the IP address, the lease time, the binding type, the VLAN number, and the interface information that corresponds to the local untrusted interfaces of a switch. It does not contain information regarding hosts interconnected with a trusted interface.

In a service-provider network, a trusted interface is connected to a port on a device in the same network. An untrusted interface is connected to an untrusted interface in the network or to an interface on a device that is not in the network.

When a switch receives a packet on an untrusted interface and the interface belongs to a VLAN in which DHCP snooping is enabled, the switch compares the source MAC address and the DHCP client hardware address. If addresses match (the default), the switch forwards the packet. If the addresses do not match, the switch drops the packet.

The switch drops a DHCP packet when one of these situations occurs:

- A packet from a DHCP server, such as a DHCPOFFER, DHCPACK, DHCPNAK, or DHCPLEASEQUERY packet, is received from outside the network or firewall.
- A packet is received on an untrusted interface, and the source MAC address and the DHCP client hardware address do not match.
- The switch receives a DHCPRELEASE or DHCPDECLINE broadcast message that contains a MAC address in the DHCP snooping binding table, but the interface information in the binding table does not match the interface on which the message was received.
- A DHCP relay agent forwards a DHCP packet that includes an relay-agent IP address that is not 0.0.0.0, or the relay agent forwards a packet that includes option-82 information to an untrusted port.

If the switch is an aggregation switch supporting DHCP snooping and is connected to an edge switch that is inserting DHCP option-82 information, the switch drops packets with option-82 information when packets are received on an untrusted interface. If DHCP snooping is enabled and packets are received on a trusted port, the aggregation switch does not learn the DHCP snooping bindings for connected devices and cannot build a complete DHCP snooping binding database.

When option-82 information is inserted by an edge switch in software releases earlier than Cisco IOS Release 12.1(22)EA3 or in Cisco IOS Release 12.2(25)SEA or later, you cannot configure DHCP snooping on an aggregation switch because the DHCP snooping bindings database is not properly populated. You also cannot configure IP source guard and dynamic Address Resolution Protocol (ARP) inspection on the switch unless you use static bindings or ARP access control lists (ACLs).

In Cisco IOS Release 12.1(22)EA3 and in Cisco IOS Release 12.2(25)SEA or later when an aggregation switch can be connected to an edge switch through an untrusted interface and you enter the **ip dhcp snooping information option allow-untrusted** global configuration command, the aggregation switch accepts packets with option-82 information from the edge switch. The aggregation switch learns the bindings for hosts connected through an untrusted switch interface. The DHCP security features, such as dynamic ARP or IP source guard, can still be enabled on the aggregation switch while the switch receives packets with option-82 information on ingress untrusted interfaces to which hosts are connected. The port on the edge switch that connects to the aggregation switch must be configured as a trusted interface.

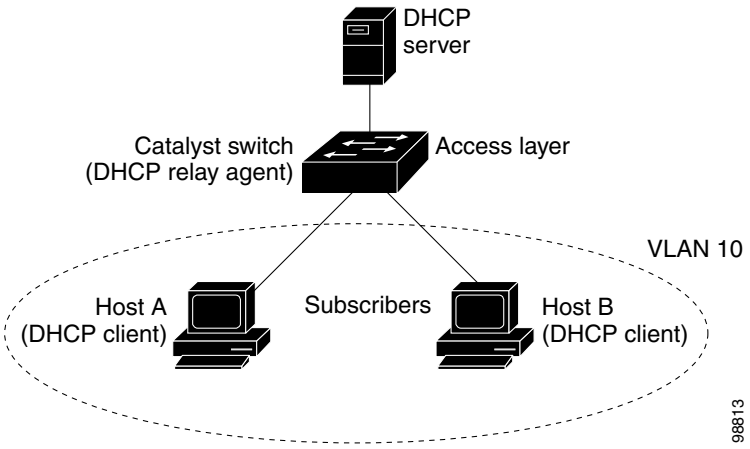
Option-82 Data Insertion

In residential, metropolitan Ethernet-access environments, DHCP can centrally manage the IP address assignments for a large number of subscribers. When the DHCP option-82 feature is enabled on the switch, a subscriber device is identified by the switch port through which it connects to the network (in addition to its MAC address). Multiple hosts on the subscriber LAN can be connected to the same port on the access switch and are uniquely identified.

Note In Cisco IOS Release 12.1(19)EA1 or later, the DHCP option-82 feature is supported when DHCP snooping is enabled globally and on the VLANs to which subscriber devices using this feature are assigned. The switch also supports the DHCP option-82 feature when DHCP is disabled.

Figure 18-1 is an example of a metropolitan Ethernet network in which a centralized DHCP server assigns IP addresses to subscribers connected to the switch at the access layer. Because the DHCP clients and their associated DHCP server do not reside on the same IP network or subnet, a DHCP relay agent (the Catalyst switch) is configured with a helper address to enable broadcast forwarding and to transfer DHCP messages between the clients and the server.

Figure 18-1 DHCP Relay Agent in a Metropolitan Ethernet Network



When you enable the DHCP snooping information option 82 on the switch, this sequence of events occurs:

- The host (DHCP client) generates a DHCP request and broadcasts it on the network.
- When the switch receives the DHCP request, it adds the option-82 information in the packet. By default, the option-82 information contains the switch MAC address (the remote-ID suboption) and the port identifier, **vlan-mod-port** or **snmp-ifindex**, from which the packet is received (the circuit-ID suboption). Beginning with Cisco IOS Release 12.2(25)SEE, you can configure the remote ID and circuit ID. For information on configuring these suboptions, see the"Enabling DHCP Snooping and Option 82" section on page 18-15.
- If the IP address of the relay agent is configured, the switch adds the IP address in the DHCP packet.
- The switch forwards the DHCP request that includes the option-82 field to the DHCP server.
- The DHCP server receives the packet. If the server is option-82 capable, it can use the remote ID, the circuit ID, or both to assign IP addresses and implement policies, such as restricting the number of IP addresses that can be assigned to a single remote ID or circuit ID. Then the DHCP server echoes the option-82 field in the DHCP reply.
- The DHCP server unicasts the reply to the switch if the request was relayed to the server by the switch. When the client and server are on the same subnet, the server broadcasts the reply. The switch verifies that it originally inserted the option-82 data by inspecting the remote ID and possibly the circuit ID fields. The switch removes the option-82 field and forwards the packet to the switch port that connects to the DHCP client that sent the DHCP request.

In the default suboption configuration, when the described sequence of events occurs, the values in these fields of examples 1 and 2 in Figure 18-2 do not change:

- Circuit-ID suboption fields
  - Suboption type
  - Length of the suboption type
  - Circuit-ID type
  - Length of the circuit-ID type
- Remote-ID suboption fields
  - Suboption type
  - Length of the suboption type

  - Remote-ID type
  - Length of the remote-ID type

Example 3 in Figure 18-2 shows the packet formats for user-configured remote-ID and circuit-ID suboptions The switch uses these packet formats when you globally enable DHCP snooping and enter both the **ip dhcp snooping information option format remote-id** global configuration command **and the ip dhcp snooping vlan information option format-type circuit-id string** interface configuration command.

The values for these fields in the packets change from the default values when you configure the remote-ID and circuit- ID suboptions:

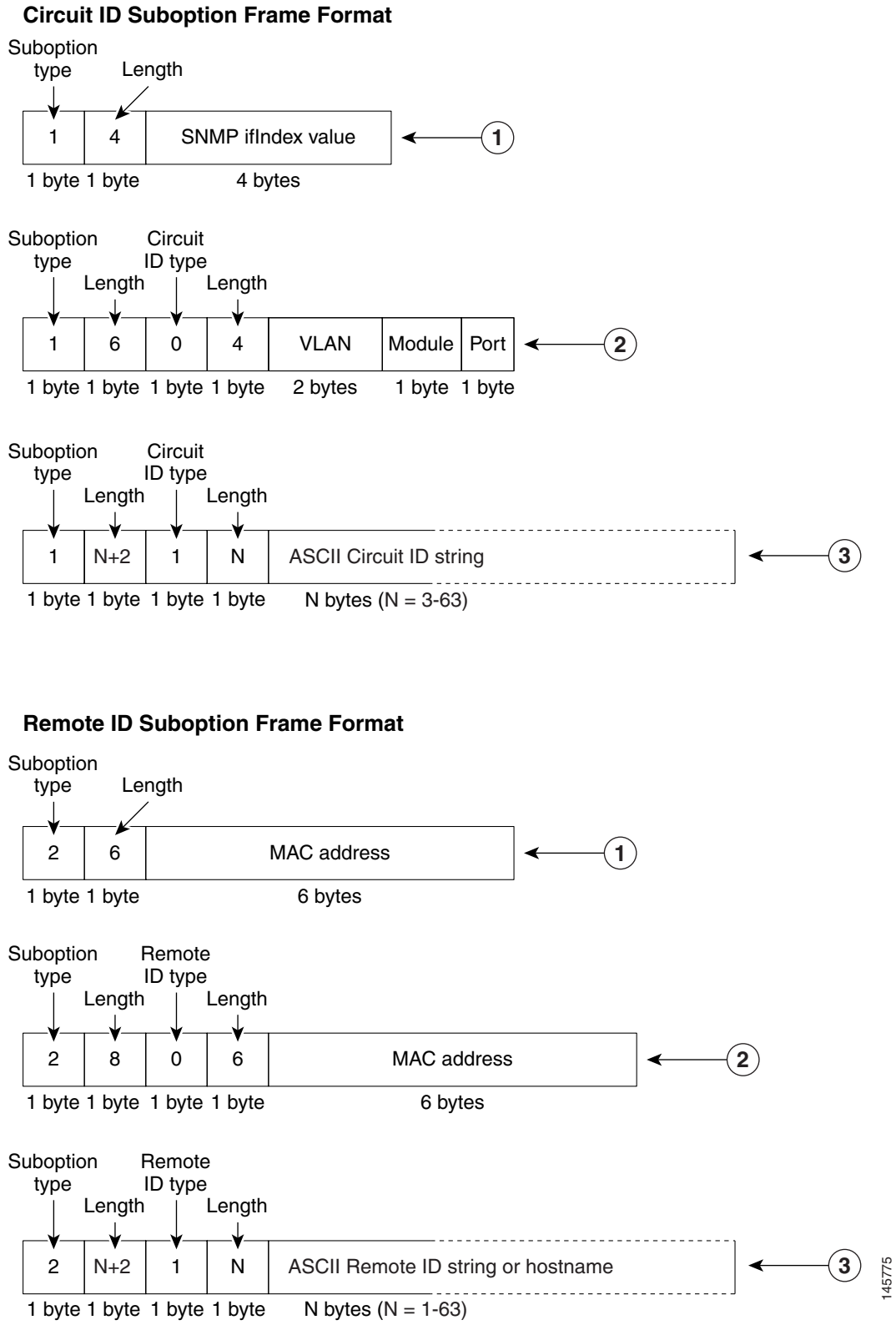
- Circuit-ID suboption fields
  - The circuit-ID type is 1.
  - The length values are variable, depending on the length of the string that you configure.
- Remote-ID suboption fields
  - The remote-ID type is 1.

The length values are variable, depending on the length of the string that you configure.

When you globally enable DHCP snooping, and enter the **ip dhcp snooping information option** global configuration command, and do not configure the SNMP ifIndex format, the port numbers in the port field of the circuit-ID suboption start at 0. For example, on a Catalyst 3550-24 switch, port 0 is the Fast Ethernet 0/1 port, port 1 is the Fast Ethernet 0/2 port, port 2 is the Fast Ethernet 0/3 port, and so on. Port 24 is the Gigabit Interface Converter (GBIC)-based Gigabit module slot 0/1, and port 25 is the GBIC-based Gigabit module slot 0/2.

Figure 18-2 shows the packet formats for the default and user-configured remote-ID suboption and circuit-ID suboption. For the circuit-ID suboption, the module field is always zero.

Figure 18-2 Suboption Packet Formats



1	When you globally enable DHCP snooping, and enter the **ip dhcp relay information option** global configuration command, and enter the **ip dhcp snooping information option format snmp-ifindex** global configuration command, the switch uses these formats.
2	When you globally enable DHCP snooping, and enter the **ip dhcp snooping information option** global configuration command, and the SNMP ifIndex format is not configured, the switch uses these formats.
3	When you globally enable DHCP snooping, and enter the **ip dhcp snooping information option format remote-id** global configuration command, and enter the **ip dhcp snooping vlan information option format-type circuit-id string** interface configuration command, the switch uses these formats.

Cisco IOS DHCP Server Database

During the DHCP-based autoconfiguration process, the designated DHCP server uses the Cisco IOS DHCP server database. It has IP addresses, *address bindings*, and configuration parameters, such as the boot file.

An address binding is a mapping between an IP address and a MAC address of a host in the Cisco IOS DHCP server database. You can manually assign the client IP address, or the DHCP server can allocate an IP address from a DHCP address pool. For more information about manual and automatic address bindings, see the "Configuring DHCP" chapter of the *Cisco IOS IP Configuration Guide, Release 12.2*.

DHCP Snooping Binding Database

When DHCP snooping is enabled, the switch uses the DHCP snooping binding database to store information about untrusted interfaces. The database can have up to 8192 bindings.

Each database entry (*binding*) has an IP address, an associated MAC address, the lease time (in hexadecimal format), the interface to which the binding applies, and the VLAN to which the interface belongs. The database agent stores the bindings in a file at a configured location. At the end of each entry is a checksum that accounts for all the bytes from the start of the file through all the bytes associated with the entry. Each entry is 72 bytes, followed by a space and then the checksum value.

To keep the bindings when the switch reloads, you must use the DHCP snooping database agent. If the agent is disabled, dynamic ARP or IP source guard is enabled, and the DHCP snooping binding database has dynamic bindings, the switch loses its connectivity. If the agent is disabled and only DHCP snooping is enabled, the switch does not lose its connectivity, but DHCP snooping might not prevent DCHP spoofing attacks.

When reloading, the switch reads the binding file to build the DHCP snooping binding database. The switch updates the file when the database changes.

When a switch learns of new bindings or when it loses bindings, the switch immediately updates the entries in the database. The switch also updates the entries in the binding file. The frequency at which the file is updated is based on a configurable delay, and the updates are batched. If the file is not updated in a specified time (set by the write-delay and abort-timeout values), the update stops.

This is the format of the file with bindings:

```
<initial-checksum>
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
<entry-1> <checksum-1>
<entry-2> <checksum-1-2>
...
...
<entry-n> <checksum-1-2-..-n>
END
```

Each entry in the file is tagged with a checksum value that the switch uses to verify the entries when it reads the file. The *initial-checksum* entry on the first line distinguishes entries associated with the latest file update from entries associated with a previous file update.

This is an example of a binding file:

```
2bb4c2a1
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
192.1.168.1 3 0003.47d8.c91f 2BB6488E Fa1/0/4 21ae5fbb
192.1.168.3 3 0003.44d6.c52f 2BB648EB Fa1/0/4 1bdb223f
192.1.168.2 3 0003.47d9.c8f1 2BB648AB Fa1/0/4 584a38f0
END
```

When the switch starts and the calculated checksum value equals the stored checksum value, the switch reads entries from the binding file and adds the bindings to its DHCP snooping binding database. The switch ignores an entry when one of these situations occurs:

- The switch reads the entry and the calculated checksum value does not equal the stored checksum value. The entry and the ones following it are ignored.
- An entry has an expired lease time (the switch might not remove a binding entry when the lease time expires).
- The interface in the entry no longer exists on the system.
- The interface is a routed interface or a DHCP snooping-trusted interface.

Configuring DHCP Features

These sections describe how to configure DHCP snooping and option 82 on your switch:

- Default DHCP Configuration, page 18-9
- DHCP Snooping Configuration Guidelines, page 18-10
- Upgrading from a Previous Software Release, page 18-10
- Configuring the DHCP Server, page 18-11
- Enabling Only the DHCP Relay Agent, page 18-11
- Enabling the DHCP Relay Agent and Option 82, page 18-11
- Validating the Relay Agent Information Option 82, page 18-12
- Configuring the Reforwarding Policy, page 18-12
- Specifying the Packet Forwarding Address, page 18-13
- Enabling DHCP Snooping and Option 82, page 18-15

- Enabling DHCP Snooping on Private VLANs, page 18-16
- Enabling the Cisco IOS DHCP Server Database, page 18-17
- Enabling the DHCP Snooping Binding Database Agent, page 18-17

Default DHCP Configuration

Table 18-1 shows the default DHCP configuration.

Table 18-1 Default DHCP Configuration

Feature	Default Setting
DHCP server	Enabled in Cisco IOS software, requires configuration[1]
DHCP relay agent	Enabled[2]
DHCP packet forwarding address	None configured
Checking the relay agent information	Enabled (invalid messages are dropped)[2]
DHCP relay agent forwarding policy	Replace the existing relay agent information[2]
DHCP snooping enabled globally	Disabled
DHCP snooping information option	Enabled
DHCP snooping option to accept packets on untrusted ingress interfaces[3]	Disabled
DHCP snooping limit rate	None configured
DHCP snooping trust	Untrusted
DHCP snooping VLAN	Disabled
DHCP snooping MAC address verification	Enabled
Cisco IOS DHCP server binding database	Enabled in Cisco IOS software, requires configuration The switch gets network addresses and configuration parameters only from a device configured as a DHCP server.
DHCP snooping binding database agent	Enabled in Cisco IOS software, requires configuration This feature is operational only when a destination is configured.

1. The switch responds to DHCP requests only if it is configured as a DHCP server.
2. The switch relays DHCP packets only if the IP address of the DHCP server is configured on the SVI of the DHCP client.
3. Use this feature when the switch is an aggregation switch that receives packets with option-82 information from an edge switch.

DHCP Snooping Configuration Guidelines

These are the configuration guidelines for DHCP snooping.

- You must globally enable DHCP snooping on the switch.
- DHCP snooping is not active until DHCP snooping is enabled on a VLAN.
- Before globally enabling DHCP snooping on the switch, make sure that the devices acting as the DHCP server and the DHCP relay agent are configured and enabled.
- When you globally enable DHCP snooping on the switch, these Cisco IOS commands are not available until snooping is disabled. If you enter these commands, the switch returns an error message, and the configuration is not applied.
  - **ip dhcp relay information check** global configuration command
  - **ip dhcp relay information policy** global configuration command
  - **ip dhcp relay information trust-all** global configuration command
  - ip dhcp relay information option global configuration command
  - **ip dhcp relay information trusted** interface configuration command
- Before configuring the DHCP snooping information option on your switch, be sure to configure the device that is acting as the DHCP server. For example, you must specify the IP addresses that the DHCP server can assign or exclude, or you must configure DHCP options for these devices.
- Before configuring the DHCP relay agent on your switch, make sure to configure the device that is acting as the DHCP server. For example, you must specify the IP addresses that the DHCP server can assign or exclude, configure DHCP options for devices, or set up the DHCP database agent.
- If the DHCP relay agent is enabled but DHCP snooping is disabled, the DHCP option-82 data insertion feature is not supported.
- If a switch port is connected to a DHCP server, configure a port as trusted by entering the **ip dhcp snooping trust** interface configuration command.
- If a switch port is connected to a DHCP client, configure a port as untrusted by entering the **no ip dhcp snooping trust** interface configuration command.
- Do not enter the **ip dhcp snooping information option allow-untrusted** command on an aggregation switch to which an untrusted device is connected. If you enter this command, an untrusted device might spoof the option-82 information.
- Starting with Cisco IOS Release 12.2(37)SE, you can display DHCP snooping statistics by entering the **show ip dhcp snooping statistics** user EXEC command, and you can clear the snooping statistics counters by entering the **clear ip dhcp snooping statistics** privileged EXEC command.

Upgrading from a Previous Software Release

In Cisco IOS Release 12.1(19)EA1, the implementation for the option 82 Subscriber Identification changed from the previous release. The new option-82 format uses a different circuit-ID and remote-ID suboption, **vlan-mod-port**. The previous version uses the **snmp-ifindex** circuit ID and remote-ID suboption.

If you have option 82 configured on the switch and you upgrade to Cisco IOS Release 12.1(19)EA1 or later, the option 82 configuration is not affected. However, when you globally enable DHCP snooping on the switch by using the **ip dhcp snooping** global configuration command, the previous option 82 configuration is suspended, and the new option 82 format is applied. When you globally disable DHCP snooping on the switch, the previous option 82 configuration is re-enabled.

To provide for backward compatibility, you can select the previous option 82 format by using the **ip dhcp snooping information option format snmp-ifindex** global configuration command when you enable DHCP snooping. When DHCP snooping is globally enabled, option-82 information (in the selected format) is only inserted on snooped VLANs.

To use the previous version of option 82 without enabling DHCP snooping, see the "Enabling the DHCP Relay Agent and Option 82" section on page 18-11 for instructions.

Beginning in Cisco IOS Release 12.2(25)SEE, you can configure a string of ASCII characters for the remote-ID and circuit-ID suboptions. For information on configuring these suboptions, see the "Enabling DHCP Snooping and Option 82" section on page 18-15

Configuring the DHCP Server

The switch can act as a DHCP server. By default, the Cisco IOS DHCP server and relay agent features are enabled on your switch but are not configured. These features are not operational.

For procedures to configure the switch as a DHCP server, see the "Configuring DHCP" section of the "IP addressing and Services" section of the *Cisco IOS IP Configuration Guide, Release 12.2*.

Enabling Only the DHCP Relay Agent

Beginning in privileged EXEC mode, follow these steps to enable the DHCP relay agent on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**service dhcp**	Enable the DHCP server and relay agent on your switch. By default, this feature is enabled.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the DHCP server and relay agent, use the **no service dhcp** global configuration command.

Enabling the DHCP Relay Agent and Option 82

In Cisco IOS Release 12.1(19)EA1, the implementation for the option 82 Subscriber Identification changed from the previous release. For more information about configuring the relay agent and option 82 when using DHCP snooping, see the "Upgrading from a Previous Software Release" section on page 18-10.

Beginning in privileged EXEC mode, follow these steps to enable the DHCP relay agent and option 82 on the switch.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**service dhcp**	Enable the DHCP server and relay agent on your switch. By default, this feature is enabled.
Step 3	**ip dhcp relay information option**	Enable the switch to insert and remove DHCP relay information (option-82 field) in forwarded DHCP request messages to the DHCP server. By default, this feature is disabled.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the DHCP server and relay agent, use the **no service dhcp** global configuration command. To disable the insertion and removal of the option-82 field, use the **no ip dhcp relay information option** global configuration command.

Validating the Relay Agent Information Option 82

By default, the switch verifies that the option-82 field in DHCP reply packet it receives from the DHCP server is valid. If an invalid message is received, the switch drops it. If a valid message is received, the switch removes the option-82 field and forwards the packet.

If you want to disable this feature, use the **no ip dhcp relay information check** global configuration command. When disabled, the switch does not validate the option-82 field for validity, but still removes the option from the packet and forwards it. (This feature is not available when DHCP snooping is enabled on the switch.)

Note If the switch receives a packet that contains the option-82 field from a DHCP client and the information checking feature is enabled, the switch drops the packet because it is invalid. However, in some instances, you might configure a client with the option-82 field. In this situation, you should disable the information-check feature so that the switch does not remove the option-82 field from the packet. You can configure the action that the switch takes when it receives a packet with existing option-82 information by using the **ip dhcp relay information policy** global configuration command. For more information, see the "Configuring the Reforwarding Policy" section on page 18-12. (This feature is not available when DHCP snooping is enabled on the switch.)

Configuring the Reforwarding Policy

By default, the reforwarding policy of the switch is to replace existing relay information in packets received from DHCP clients with switch DHCP relay information. If the default action is not suitable for your network configuration, you can use the **ip dhcp relay information policy** {**drop** | **keep** | **replace**} global configuration command to change it. (This feature is not available when DHCP snooping is enabled on the switch.)

Note To ensure the correct operation of the reforwarding policy, make sure to disable the relay agent information check by using the **no ip dhcp relay information check** global configuration command.

Beginning in privileged EXEC mode, follow these steps to change the action of the reforwarding policy.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip dhcp relay information policy** {**drop** | **keep** | **replace**}	Configure the reforwarding policy. The default is to replace (overwrite) existing information with switch DHCP relay information. • Use the **drop** keyword if you want the switch to discard messages with existing relay information if the option-82 information is also present. • Use the **keep** keyword if you want the switch to retain the existing relay information.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default reforwarding policy, use the **no ip dhcp relay information policy** global configuration command.

Specifying the Packet Forwarding Address

A DHCP relay agent is any device that forwards DHCP packets between clients and servers when they are not on the same physical subnet. Relay agent forwarding is distinct from the normal forwarding of an IP router, where IP datagrams are transparently switched between networks. By contrast, relay agents receive DHCP messages and then generate a new DHCP message to send on another interface.

If the DHCP server and the DHCP clients are on different networks or subnets, you must configure the switch with the **ip helper-address** *address* interface configuration command. The general rule is to configure the command on the Layer 3 interface closest to the client. The address used in the **ip helper-address** command can be a specific DHCP server IP address, or it can be the network address if other DHCP servers are on the destination network segment. Using the network address enables any DHCP server to respond to requests.

Beginning in privileged EXEC mode, follow these steps to specify the packet forwarding address. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface vlan** *vlan-id*	Enter a VLAN ID to create a switch virtual interface, and enter interface configuration mode.
Step 3	**ip address** *ip-address subnet-mask*	Configure the interface with an IP address and an IP subnet.

	Command	Purpose
Step 4	**ip helper-address** *address*	Specify the DHCP packet forwarding address. The helper address can be a specific DHCP server address, or it can be the network address if other DHCP servers are on the destination network segment. Using the network address enables other servers to respond to DHCP requests. If you have multiple servers, you can configure one helper address for each server.
Step 5	**exit**	Return to global configuration mode.
Step 6	**interface range** *port-range* or **interface** *interface-id*	Configure multiple physical ports that are connected to the DHCP clients, and enter interface range configuration mode. or Configure a single physical port that is connected to the DHCP client, and enter interface configuration mode.
Step 7	**switchport mode access**	Define the VLAN membership mode for the port.
Step 8	**switchport access vlan** *vlan-id*	Assign the ports to the same VLAN as configured in Step 2.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show running-config**	Verify your entries.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the DHCP packet forwarding address, use the **no ip helper-address** *address* interface configuration command.

This example shows how to enable the DHCP server, the relay agent, and the insertion and removal of the DHCP relay information (option 82). It creates a switch virtual interface with VLAN ID 10, assigns it an IP address, and specifies the DHCP packet forwarding address of 30.0.0.2 (DHCP server address). Two interfaces (Gigabit Ethernet 0/1 and 0/2) that connect to the DHCP clients are configured as static access ports in VLAN 10 (see Figure 18-1 on page 18-4):

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# service dhcp
Switch(config)# ip dhcp relay information option
Switch(config)# interface vlan 10
Switch(config-if)# ip address 10.0.0.1 255.0.0.0
Switch(config-if)# ip helper-address 30.0.0.2
Switch(config-if)# exit
Switch(config)# interface range gigabitethernet0/1 - 2
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 10
Switch(config-if)# exit
```

Enabling DHCP Snooping and Option 82

Beginning in privileged EXEC mode, follow these steps to enable DHCP snooping on the switch.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip dhcp snooping**	Enable DHCP snooping globally.
Step 3	**ip dhcp snooping vlan** *vlan-range*	Enable DHCP snooping on a VLAN or range of VLANs. The range is 1 to 4094. You can enter a single VLAN ID identified by VLAN ID number, a series of VLAN IDs separated by commas, a range of VLAN IDs separated by hyphens, or a range of VLAN IDs separated by entering the starting and ending VLAN IDs separated by a space.
Step 4	**ip dhcp snooping information option**	Enable the switch to insert and remove DHCP relay information (option-82 field) in forwarded DHCP request messages to the DHCP server. The default is enabled.
Step 5	**ip dhcp snooping information option format snmp-ifindex**	(Optional) Specify **ip dhcp snooping information option format snmp-ifindex** to select an alternate format for the circuit-ID and remote-ID suboption of the option 82 feature. See the "Upgrading from a Previous Software Release" section on page 18-10 for more information. The default setting is **no ip dhcp snooping information option format snmp-ifindex**.
Step 6	**ip dhcp snooping information option format remote-id [string** *ASCII-string* | hostname]	(Optional) Configure the remote-ID suboption. You can configure the remote ID to be: • String of up to 63 ASCII characters (no spaces) • Configured hostname for the switch **Note** If the hostname is longer than 63 characters, it is truncated to 63 characters in the remote-ID configuration. The default remote ID is the switch MAC address.
Step 7	**ip dhcp snooping information option allow-untrusted**	(Optional) If the switch is an aggregation switch connected to an edge switch, enable the switch to accept incoming DHCP snooping packets with option-82 information from the edge switch. The default is disabled. **Note** You must only enter this command on aggregation switches that are connected to trusted devices.
Step 8	interface *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 9	**ip dhcp snooping vlan** *vlan* **information option format-type circuit-id string** *ASCII-string*	(Optional) Configure the circuit-ID suboption for the specified interface. Specify the VLAN and port identifier, using a VLAN ID in the range of 1 to 4094. You can configure the circuit ID to be a string of 3 to 63 ASCII characters (no spaces). The default circuit ID is the port identifier, in the format **vlan-mod-port.**

	Command	Purpose
Step 10	**ip dhcp snooping trust**	(Optional) Configure the interface as trusted or untrusted. You can use the **no** keyword to configure an interface to receive messages from an untrusted client. The default is untrusted.
Step 11	**ip dhcp snooping limit rate** *rate*	(Optional) Configure the number of DHCP packets per second than an interface can receive. The range is 1 to 4294967294. The default is no rate limit configured. **Note** We recommend an untrusted rate limit of not more than 100 packets per second. If you configure rate limiting for trusted interfaces, you might need to increase the rate limit if the port is a trunk port assigned to more than one VLAN on which DHCP snooping is enabled.
Step 12	**exit**	Return to global configuration mode.
Step 13	**ip dhcp snooping verify** *mac-address*	(Optional) Configure the switch to verify that the source MAC address in a DHCP packet that is received on untrusted ports matches the client hardware address in the packet. The default is to verify that the source MAC address matches the client hardware address in the packet.
Step 14	**end**	Return to privileged EXEC mode.
Step 15	**show running-config**	Verify your entries.
Step 16	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable DHCP snooping, use the **no ip dhcp snooping** global configuration command. To disable DHCP snooping on a VLAN or range of VLANs, use the **no ip dhcp snooping vlan** *vlan-id* global configuration command. To disable the insertion and removal of the option-82 field, use the **no ip dhcp snooping information option** global configuration command. To configure an aggregation switch to drop incoming DHCP snooping packets with option-82 information from an edge switch, use the **no ip dhcp snooping information option allow-untrusted** global configuration command.

This example shows how to enable DHCP snooping globally and on VLAN 10 and to configure a rate limit of 100 packets per second on Fast Ethernet port 0/1:

```
Switch(config)# ip dhcp snooping
Switch(config)# ip dhcp snooping vlan 10
Switch(config)# ip dhcp snooping information option
Switch(config)# interface fastethernet0/1
Switch(config-if)# ip dhcp snooping limit rate 100
```

Enabling DHCP Snooping on Private VLANs

You can enable DHCP snooping on private VLANs. If DHCP snooping is enabled, the configuration is propagated to both a primary VLAN and its associated secondary VLANs. If DHCP snooping is enabled on the primary VLAN, it is also configured on the secondary VLANs.

If DHCP snooping is already configured on the primary VLAN and you configure DHCP snooping with different settings on a secondary VLAN, the configuration for the secondary VLAN does not take effect. You must configure DHCP snooping on the primary VLAN. If DHCP snooping is not configured on the primary VLAN, this message appears when you are configuring DHCP snooping on the secondary VLAN, such as VLAN 200:

```
2w5d:%DHCP_SNOOPING-4-DHCP_SNOOPING_PVLAN_WARNING:DHCP Snooping configuration may not take
effect on secondary vlan 200. DHCP Snooping configuration on secondary vlan is derived
from its primary vlan.
```

The **show ip dhcp snooping** privileged EXEC command output shows all VLANs, including primary and secondary private VLANs, on which DHCP snooping is enabled.

Enabling the Cisco IOS DHCP Server Database

For procedures to enable and configure the Cisco IOS DHCP server database, see the "DHCP Configuration Task List" section in the "Configuring DHCP" chapter of the *Cisco IOS IP Configuration Guide, Release 12.2*.

Enabling the DHCP Snooping Binding Database Agent

Beginning in privileged EXEC mode, follow these steps to enable and configure the DHCP snooping binding database agent on the switch.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip dhcp snooping database** {**flash:/***filename* | **ftp://***user***:***password@host/filename* | **http://**[[*username***:***password*]@] {*hostname* | *host-ip*}[*/directory*] */image-name***.tar** | **rcp://***user@host/filename*}	Specify the URL for the database agent or the binding file by using one of these forms: • **flash:/***filename* • **ftp://***user***:***password@host/filename* • **http://**[[*username***:***password*]@]{*hostname* | *host-ip*}[*/directory*]*/image-name***.tar** • **rcp://***user@host/filename* • **tftp://***host/filename*
Step 3	**ip dhcp snooping database timeout** *seconds*	Specify (in seconds) how long to wait for the database transfer to finish before stopping. The range is 0 to 86400. Use 0 for an infinite duration. The default is 300 seconds (5 minutes).
Step 4	**ip dhcp snooping database write-delay** *seconds*	Specify the duration for which the transfer should be delayed after the binding database changes. The range is from 15 to 86400 seconds. The default is 300 seconds (5 minutes).
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**ip dhcp snooping binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* **expiry** *seconds*	(Optional) Add binding entries to the DHCP snooping binding database. The *vlan-id* range is from 1 to 4904. The *seconds* range is from 1 to 4294967295. Enter this command for each entry that you add. **Note** Use this command when you are testing or debugging the switch.
Step 7	**show ip dhcp snooping database** [**detail**]	Display the status and statistics of the DHCP snooping binding database agent.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To stop using the database agent and binding files, use the **no ip dhcp snooping database** global configuration command. To reset the timeout or delay values, use the **ip dhcp snooping database timeout** *seconds* or the **ip dhcp snooping database write-delay** *seconds* global configuration command.

To clear the statistics of the DHCP snooping binding database agent, use the **clear ip dhcp snooping database statistics** privileged EXEC command. To renew the database, use the **renew ip dhcp snooping database** privileged EXEC command.

To delete binding entries from the DHCP snooping binding database, use the **no ip dhcp snooping binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* privileged EXEC command. Enter this command for each entry that you want to delete.

Displaying DHCP Information

To display the DHCP snooping information, use one or more of the privileged EXEC commands in Table 18-2:

Table 18-2 Commands for Displaying DHCP Information

Command	Purpose
show ip dhcp snooping	Displays the DHCP snooping configuration for a switch.
show ip dhcp snooping binding	Displays only the dynamically configured bindings in the DHCP snooping binding database.[1]
show ip dhcp snooping database	Displays the DHCP snooping binding database status and statistics.
show ip dhcp snooping statistics	Displays the DHCP snooping statistics in summary or detail form.
show running-config	Displays the status of the insertion and removal of the DHCP option-82 field on all interfaces.

1. If DHCP snooping is enabled and an interface changes to the down state, the switch does not delete the manually configured bindings.

Understanding IP Source Guard

IP source guard is a security feature that restricts IP traffic on nonrouted, Layer 2 interfaces by filtering traffic based on the DHCP snooping binding database and on manually configured IP source bindings. You can use IP source guard to prevent traffic attacks caused when a host tries to use the IP address of its neighbor.

You can enable IP source guard when DHCP snooping is enabled on an untrusted interface. After IP source guard is enabled on an interface, the switch blocks all IP traffic received on the interface, except for DHCP packets allowed by DHCP snooping. A port access control list (ACL) is applied to the interface. The port ACL allows only IP traffic with a source IP address in the IP source binding table and denies all other traffic.

The IP source binding table has bindings that are learned by DHCP snooping or are manually configured (static IP source bindings). An entry in this table has an IP address, its associated MAC address, and its associated VLAN number. The switch uses the IP source binding table only when IP source guard is enabled.

IP source guard is supported only on Layer 2 ports, including access and trunk ports.You can configure IP source guard with source IP address filtering or with source IP and MAC address filtering.

Source IP Address Filtering

When IP source guard is enabled with this option, IP traffic is filtered based on the source IP address. The switch forwards IP traffic when the source IP address matches an entry in the DHCP snooping binding database or a binding in the IP source binding table.

When a DHCP snooping binding or static IP source binding is added, changed, or deleted on an interface, the switch modifies the port ACL using the IP source binding changes, and re-applies the port ACL to the interface.

If you enable IP source guard on an interface on which IP source bindings (dynamically learned by DHCP snooping or manually configured) are not configured, the switch creates and applies a port ACL that denies all IP traffic on the interface. If you disable IP source guard, the switch removes the port ACL from the interface.

Source IP and MAC Address Filtering

When IP source guard is enabled with this option, IP traffic is filtered based on the source IP and MAC addresses. The switch forwards traffic only when the source IP and MAC addresses match an entry in the IP source binding table.

When IP source guard with source IP and MAC address filtering is enabled, the switch filters IP and non-IP traffic. If the source MAC address of an IP or non-IP packet matches a valid IP source binding, the switch forwards the packet. The switch drops all other types of packets except DHCP packets.

The switch uses port security to filter source MAC addresses. The interface can shut down when a port-security violation occurs.

Configuring IP Source Guard

This section describes how to configure IP source guard on your switch.

- Default IP Source Guard Configuration, page 18-19
- IP Source Guard Configuration Guidelines, page 18-20
- Enabling IP Source Guard, page 18-20
- Displaying IP Source Guard Information, page 18-22

Default IP Source Guard Configuration

By default, IP source guard is disabled.

IP Source Guard Configuration Guidelines

These are the configuration guides for IP source guard:

- You can configure static IP bindings only on nonrouted ports. If you enter the **ip source binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* global configuration command on a routed interface, this error message appears:

  ```
  Static IP source binding can only be configured on switch port.
  ```

- When IP source guard with source IP filtering is enabled on an interface, DHCP snooping must be enabled on the access VLAN to which the interface belongs.
- If you are enabling IP source guard on a trunk interface with multiple VLANs and DHCP snooping is enabled on all the VLANs, the source IP address filter is applied on all the VLANs.

  Note If IP source guard is enabled and you enable or disable DHCP snooping on a VLAN on the trunk interface, the switch might not properly filter traffic.

- When IP source guard with source IP and MAC address filtering is enabled, DHCP snooping and port security must be enabled on the interface.You must also enter the **ip dhcp snooping information option** global configuration command and ensure that the DHCP server supports option 82. When IP source guard is enabled with MAC address filtering, the DHCP host MAC address is not learned until the host is granted a lease. When forwarding packets from the server to the host, DHCP snooping uses the option-82 data to identify the host port.
- When configuring IP source guard on interfaces on which a private VLAN is configured, port security is not supported.
- IP source guard is not supported on EtherChannels.
- You can enable this feature when IEEE 802.1x port-based authentication is enabled.
- If the number of ternary content addressable memory (TCAM) entries exceeds the maximum available, the CPU usage increases.

Enabling IP Source Guard

Beginning in privileged EXEC mode, follow these steps to enable and configure IP source guard on an interface.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.

	Command	Purpose
Step 3	**ip verify source** or **ip verify source port-security**	Enable IP source guard with source IP address filtering. Enable IP source guard with source IP and MAC address filtering. **Note** When you enable both IP Source Guard and Port Security, using the **ip verify source port-security** interface configuration command, there are two caveats: • The DHCP server must support option 82, or the client is not assigned an IP address. • The MAC address in the DHCP packet is not learned as a secure address. The MAC address of the DHCP client is learned as a secure address only when the switch receives non-DHCP data traffic.
Step 4	**exit**	Return to global configuration mode.
Step 5	**ip source binding** *mac-address* **vlan** *vlan-id ip-address* **inteface** *interface-id*	Add a static IP source binding. Enter this command for each static binding.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show ip verify source** [**interface** *interface-id*]	Display the IP source guard configuration for all interfaces or for a specific interface.
Step 8	**show ip source binding** [*ip-address*] [*mac-address*] [**dhcp-snooping** | **static**] [**interface** *interface-id*] [**vlan** *vlan-id*]	Display the IP source bindings on the switch, on a specific VLAN, or on a specific interface.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IP source guard with source IP address filtering, use the **no ip verify source** interface configuration command.

To delete a static IP source binding entry, use the **no ip source global** configuration command.

This example shows how to enable IP source guard with source IP and MAC filtering on VLANs 10 and 11:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip verify source port-security
Switch(config-if)# exit
Switch(config)# ip source binding 0100.0022.0010 vlan 10 10.0.0.2 interface
gigabitethernet1/0/1
Switch(config)# ip source binding 0100.0230.0002 vlan 11 10.0.0.4 interface
gigabitethernet1/0/1
Switch(config)# end
```

Displaying IP Source Guard Information

To display the IP source guard information, use one or more of the privileged EXEC commands in Table 18-3:

Table 18-3 Commands for Displaying IP Source Guard Information

Command	Purpose
show ip source binding	Display the IP source bindings on a switch.
show ip verify source	Display the IP source guard configuration on the switch.

CHAPTER 19

Configuring Dynamic ARP Inspection

This chapter describes how to configure dynamic Address Resolution Protocol inspection (dynamic ARP inspection) on the Catalyst 3550 switch. This feature helps prevent malicious attacks on the switch by not relaying invalid ARP requests and responses to other ports in the same VLAN.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

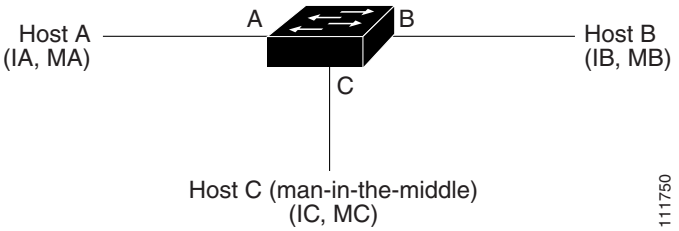
- Understanding Dynamic ARP Inspection, page 19-1
- Configuring Dynamic ARP Inspection, page 19-5
- Displaying Dynamic ARP Inspection Information, page 19-14

Understanding Dynamic ARP Inspection

ARP provides IP communication within a Layer 2 broadcast domain by mapping an IP address to a MAC address. For example, Host B wants to send information to Host A but does not have the MAC address of Host A in its ARP cache. Host B generates a broadcast message for all hosts within the broadcast domain to obtain the MAC address associated with the IP address of Host A. All hosts within the broadcast domain receive the ARP request, and Host A responds with its MAC address. However, because ARP allows a gratuitous reply from a host even if an ARP request was not received, an ARP spoofing attack and the poisoning of ARP caches can occur. After the attack, all traffic from the device under attack flows through the attacker's computer and then to the router, switch, or host.

A malicious user can attack hosts, switches, and routers connected to your Layer 2 network by poisoning the ARP caches of systems connected to the subnet and by intercepting traffic intended for other hosts on the subnet. Figure 19-1 shows an example of ARP cache poisoning.

Figure 19-1 ARP Cache Poisoning



Hosts A, B, and C are connected to the switch on interfaces A, B and C, all of which are on the same subnet. Their IP and MAC addresses are shown in parentheses; for example, Host A uses IP address IA and MAC address MA. When Host A needs to communicate to Host B at the IP layer, it broadcasts an ARP request for the MAC address associated with IP address IB. When the switch and Host B receive the ARP request, they populate their ARP caches with an ARP binding for a host with the IP address IA and a MAC address MA; for example, IP address IA is bound to MAC address MA. When Host B responds, the switch and Host A populate their ARP caches with a binding for a host with the IP address IB and the MAC address MB.

Host C can poison the ARP caches of the switch, Host A, and Host B by broadcasting forged ARP responses with bindings for a host with an IP address of IA (or IB) and a MAC address of MC. Hosts with poisoned ARP caches use the MAC address MC as the destination MAC address for traffic intended for IA or IB. This means that Host C intercepts that traffic. Because Host C knows the true MAC addresses associated with IA and IB, it can forward the intercepted traffic to those hosts by using the correct MAC address as the destination. Host C has inserted itself into the traffic stream from Host A to Host B, the classic *man-in-the middle* attack.

Dynamic ARP inspection is a security feature that validates ARP packets in a network. It intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. This capability protects the network from certain man-in-the-middle attacks.

Dynamic ARP inspection ensures that only valid ARP requests and responses are relayed. The switch performs these activities:

- Intercepts all ARP requests and responses on untrusted ports
- Verifies that each of these intercepted packets has a valid IP-to-MAC address binding before updating the local ARP cache or before forwarding the packet to the appropriate destination
- Drops invalid ARP packets

Dynamic ARP inspection determines the validity of an ARP packet based on valid IP-to-MAC address bindings stored in a trusted database, the DHCP snooping binding database. This database is built by DHCP snooping if DHCP snooping is enabled on the VLANs and on the switch. If the ARP packet is received on a trusted interface, the switch forwards the packet without any checks. On untrusted interfaces, the switch forwards the packet only if it is valid.

You enable dynamic ARP inspection on a per-VLAN basis by using the **ip arp inspection vlan** *vlan-range* global configuration command. For configuration information, see the "Configuring Dynamic ARP Inspection in DHCP Environments" section on page 19-7.

In non-DHCP environments, dynamic ARP inspection can validate ARP packets against user-configured ARP access control lists (ACLs) for hosts with statically configured IP addresses. You define an ARP ACL by using the **arp access-list** *acl-name* global configuration command. For configuration information, see the "Configuring ARP ACLs for Non-DHCP Environments" section on page 19-8. The switch logs dropped packets. For more information about the log buffer, see the "Logging of Dropped Packets" section on page 19-4.

You can configure dynamic ARP inspection to drop ARP packets when the IP addresses in the packets are invalid or when the MAC addresses in the body of the ARP packets do not match the addresses specified in the Ethernet header. Use the **ip arp inspection validate** {[**src-mac**] [**dst-mac**] [**ip**]} global configuration command. For more information, see the "Performing Validation Checks" section on page 19-11.

Interface Trust States and Network Security

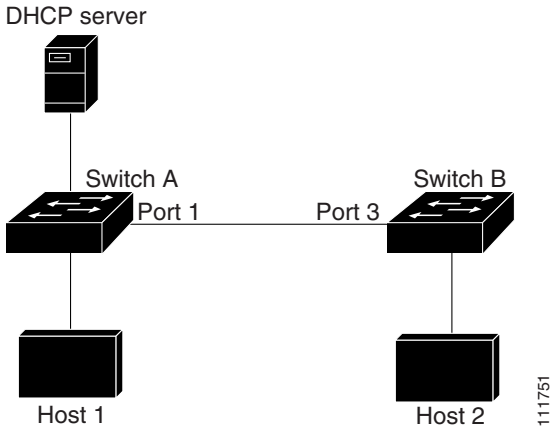
Dynamic ARP inspection associates a trust state with each interface on the switch. Packets arriving on trusted interfaces bypass all dynamic ARP inspection validation checks, and those arriving on untrusted interfaces undergo the dynamic ARP inspection validation process.

In a typical network configuration, you configure all switch ports connected to host ports as untrusted and configure all switch ports connected to switches as trusted. With this configuration, all ARP packets entering the network from a given switch bypass the security check. No other validation is needed at any other place in the VLAN or in the network. You configure the trust setting by using the **ip arp inspection trust** interface configuration command.

Caution Use the trust state configuration carefully. Configuring interfaces as untrusted when they should be trusted can result in a loss of connectivity.

In Figure 19-2, assume that both Switch A and Switch B are running dynamic ARP inspection on the VLAN that includes Host 1 and Host 2. If Host 1 and Host 2 acquire their IP addresses from the DHCP server connected to Switch A, only Switch A binds the IP-to-MAC address of Host 1. Therefore, if the interface between Switch A and Switch B is untrusted, the ARP packets from Host 1 are dropped by Switch B. Connectivity between Host 1 and Host 2 is lost.

Figure 19-2 ARP Packet Validation on a VLAN Enabled for Dynamic ARP Inspection



Configuring interfaces to be trusted when they are actually untrusted leaves a security hole in the network. If Switch A is not running dynamic ARP inspection, Host 1 can easily poison the ARP cache of Switch B (and Host 2, if the link between the switches is configured as trusted). This condition can occur even though Switch B is running dynamic ARP inspection.

Dynamic ARP inspection ensures that hosts (on untrusted interfaces) connected to a switch running dynamic ARP inspection do not poison the ARP caches of other hosts in the network. However, dynamic ARP inspection does not prevent hosts in other portions of the network from poisoning the caches of the hosts that are connected to a switch running dynamic ARP inspection.

In cases in which some switches in a VLAN run dynamic ARP inspection and other switches do not, configure the interfaces connecting such switches as untrusted. However, to validate the bindings of packets from nondynamic ARP inspection switches, configure the switch running dynamic ARP inspection with ARP ACLs. When you cannot determine such bindings, at Layer 3, isolate switches running dynamic ARP inspection from switches not running dynamic ARP inspection switches. For configuration information, see the "Configuring ARP ACLs for Non-DHCP Environments" section on page 19-8.

Note Depending on the setup of the DHCP server and the network, it might not be possible to validate a given ARP packet on all switches in the VLAN.

Rate Limiting of ARP Packets

The switch CPU performs dynamic ARP inspection validation checks; therefore, the number of incoming ARP packets is rate-limited to prevent a denial-of-service attack. By default, the rate for untrusted interfaces is 15 packets per second (pps). Trusted interfaces are not rate-limited. You can change this setting by using the **ip arp inspection limit** interface configuration command.

When the rate of incoming ARP packets exceeds the configured limit, the switch places the port in the error-disabled state. The port remains in that state until you intervene. You can use the **errdisable recovery** global configuration command to enable error disable recovery so that ports automatically emerge from this state after a specified timeout period.

For configuration information, see the "Limiting the Rate of Incoming ARP Packets" section on page 19-10.

Relative Priority of ARP ACLs and DHCP Snooping Entries

Dynamic ARP inspection uses the DHCP snooping binding database for the list of valid IP-to-MAC address bindings.

ARP ACLs take precedence over entries in the DHCP snooping binding database. The switch uses ACLs only if you configure them by using the **ip arp inspection filter vlan** global configuration command. The switch first compares ARP packets to user-configured ARP ACLs. If the ARP ACL denies the ARP packet, the switch also denies the packet even if a valid binding exists in the database populated by DHCP snooping.

Logging of Dropped Packets

When the switch drops a packet, it places an entry in the log buffer and then generates system messages on a rate-controlled basis. After the message is generated, the switch clears the entry from the log buffer. Each log entry contains flow information, such as the receiving VLAN, the port number, the source and destination IP addresses, and the source and destination MAC addresses.

You use the **ip arp inspection log-buffer** global configuration command to configure the number of entries in the buffer and the number of entries needed in the specified interval to generate system messages. You specify the type of packets that are logged by using the **ip arp inspection vlan logging** global configuration command. For configuration information, see the "Configuring the Log Buffer" section on page 19-12.

Configuring Dynamic ARP Inspection

These sections describe how to configure dynamic ARP inspection on your switch:

- Default Dynamic ARP Inspection Configuration, page 19-5
- Dynamic ARP Inspection Configuration Guidelines, page 19-6
- Configuring Dynamic ARP Inspection in DHCP Environments, page 19-7 (required in DHCP environments)
- Configuring ARP ACLs for Non-DHCP Environments, page 19-8 (required in non-DHCP environments)
- Limiting the Rate of Incoming ARP Packets, page 19-10 (optional)
- Performing Validation Checks, page 19-11 (optional)
- Configuring the Log Buffer, page 19-12 (optional)

Default Dynamic ARP Inspection Configuration

Table 19-1 shows the default dynamic ARP inspection configuration.

Table 19-1 Default Dynamic ARP Inspection Configuration

Feature	Default Setting
Dynamic ARP inspection	Disabled on all VLANs.
Interface trust state	All interfaces are untrusted.
Rate limit of incoming ARP packets	The rate is 15 pps on untrusted interfaces, assuming that the network is a switched network with a host connecting to as many as 15 new hosts per second. The rate is unlimited on all trusted interfaces. The burst interval is 1 second.
ARP ACLs for non-DHCP environments	No ARP ACLs are defined.
Validation checks	No checks are performed.
Log buffer	When dynamic ARP inspection is enabled, all denied or dropped ARP packets are logged. The number of entries in the log is 32. The number of system messages is limited to 5 per second. The logging-rate interval is 1 second.
Per-VLAN logging	All denied or dropped ARP packets are logged.

Dynamic ARP Inspection Configuration Guidelines

These are the dynamic ARP inspection configuration guidelines:

- Dynamic ARP inspection is an ingress security feature; it does not perform any egress checking.
- Dynamic ARP inspection is not effective for hosts connected to switches that do not support dynamic ARP inspection or that do not have this feature enabled. Because man-in-the-middle attacks are limited to a single Layer 2 broadcast domain, separate the domain with dynamic ARP inspection checks from the one with no checking. This action secures the ARP caches of hosts in the domain enabled for dynamic ARP inspection.
- Dynamic ARP inspection depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically assigned IP addresses. For configuration information, see Chapter 18, "Configuring DHCP Features and IP Source Guard."

  When DHCP snooping is disabled or in non-DHCP environments, use ARP ACLs to permit or to deny packets.
- Dynamic ARP inspection is supported on access ports, trunk ports, EtherChannel ports, and private VLAN ports.
- A physical port can join an EtherChannel port channel only when the trust state of the physical port and the channel port match. Otherwise, the physical port remains suspended in the port channel. A port channel inherits its trust state from the first physical port that joins the channel. Consequently, the trust state of the first physical port need not match the trust state of the channel.

  Conversely, when you change the trust state on the port channel, the switch configures a new trust state on all the physical ports that comprise the channel.
- The rate limit is calculated separately on each switch in a switch stack. For a cross-stack EtherChannel, this means that the actual rate limit might be higher than the configured value. For example, if you set the rate limit to 30 pps on an EtherChannel that has one port on switch 1 and one port on switch 2, each port can receive packets at 29 pps without causing the EtherChannel to become error-disabled.
- The operating rate for the port channel is cumulative across all the physical ports within the channel. For example, if you configure the port channel with an ARP rate-limit of 400 pps, all the interfaces combined on the channel receive an aggregate 400 pps. The rate of incoming ARP packets on EtherChannel ports is equal to the sum of the incoming rate of packets from all the channel members. Configure the rate limit for EtherChannel ports only after examining the rate of incoming ARP packets on the channel-port members.

  The rate of incoming packets on a physical port is checked against the port-channel configuration rather than the physical-ports configuration. The rate-limit configuration on a port channel is independent of the configuration on its physical ports.

  If the EtherChannel receives more ARP packets than the configured rate, the channel (including all physical ports) is placed in the error-disabled state.
- Make sure to limit the rate of ARP packets on incoming trunk ports. Configure trunk ports with higher rates to reflect their aggregation and to handle packets across multiple dynamic ARP inspection-enabled VLANs. You also can use the **ip arp inspection limit none** interface configuration command to make the rate unlimited. A high rate-limit on one VLAN can cause a denial-of-service attack to other VLANs when the software places the port in the error-disabled state.

Configuring Dynamic ARP Inspection in DHCP Environments

This procedure shows how to configure dynamic ARP inspection when two switches support this feature. Host 1 is connected to Switch A, and Host 2 is connected to Switch B as shown in Figure 19-2 on page 19-3. Both switches are running dynamic ARP inspection on VLAN 1 where the hosts are located. A DHCP server is connected to Switch A. Both hosts acquire their IP addresses from the same DHCP server. Therefore, Switch A has the bindings for Host 1 and Host 2, and Switch B has the binding for Host 2.

Note Dynamic ARP inspection depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically assigned IP addresses. For configuration information, see Chapter 18, "Configuring DHCP Features and IP Source Guard."

For information on how to configure dynamic ARP inspection when only one switch supports the feature, see the "Configuring ARP ACLs for Non-DHCP Environments" section on page 19-8.

Beginning in privileged EXEC mode, follow these steps to configure dynamic ARP inspection. You must perform this procedure on both switches. This procedure is required.

	Command	Purpose
Step 1	**show cdp neighbors**	Verify the connection between the switches.
Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**ip arp inspection *vlan*** *vlan-range*	Enable dynamic ARP inspection on a per-VLAN basis. By default, dynamic ARP inspection is disabled on all VLANs. For *vlan-range*, specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. Specify the same VLAN ID for both switches.
Step 4	**interface** *interface-id*	Specify the interface connected to the other switch, and enter interface configuration mode.
Step 5	**ip arp inspection trust**	Configure the connection between the switches as trusted. By default, all interfaces are untrusted. The switch does not check ARP packets that it receives from the other switch on the trusted interface. It simply forwards the packets. For untrusted interfaces, the switch intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The switch drops invalid packets and logs them in the log buffer according to the logging configuration specified with the **ip arp inspection vlan logging** global configuration command. For more information, see the "Configuring the Log Buffer" section on page 19-12.
Step 6	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 7	**show ip arp inspection interfaces** **show ip arp inspection vlan** *vlan-range*	Verify the dynamic ARP inspection configuration.
Step 8	**show ip dhcp snooping binding**	Verify the DHCP bindings.
Step 9	**show ip arp inspection statistics vlan** *vlan-range*	Check the dynamic ARP inspection statistics.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable dynamic ARP inspection, use the **no ip arp inspection vlan** *vlan-range* global configuration command. To return the interfaces to an untrusted state, use the **no ip arp inspection trust** interface configuration command.

This example shows how to configure dynamic ARP inspection on Switch A in VLAN 1. You would perform a similar procedure on Switch B:

```
Switch(config)# ip arp inspection vlan 1
Switch(config)# interface gigabitethernet 0/1
Switch(config-if)# ip arp inspection trust
```

Configuring ARP ACLs for Non-DHCP Environments

This procedure shows how to configure dynamic ARP inspection when Switch B shown in Figure 19-2 on page 19-3 does not support dynamic ARP inspection or DHCP snooping.

If you configure port 1 on Switch A as trusted, a security hole is created because both Switch A and Host 1 could be attacked by either Switch B or Host 2. To prevent this possibility, you must configure port 1 on Switch A as untrusted. To permit ARP packets from Host 2, you must set up an ARP ACL and apply it to VLAN 1. If the IP address of Host 2 is not static (it is impossible to apply the ACL configuration on Switch A) you must separate Switch A from Switch B at Layer 3 and use a router to route packets between them.

Beginning in privileged EXEC mode, follow these steps to configure an ARP ACL on Switch A. This procedure is required in non-DHCP environments.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**arp access-list** *acl-name*	Define an ARP ACL, and enter ARP access-list configuration mode. By default, no ARP access lists are defined. **Note** At the end of the ARP access list, there is an implicit **deny ip any mac any** command.

	Command	Purpose
Step 3	**permit ip host** *sender-ip* **mac host** *sender-mac* [**log**]	Permit ARP packets from the specified host (Host 2). • For *sender-ip*, enter the IP address of Host 2. • For *sender-mac*, enter the MAC address of Host 2. • (Optional) Specify **log** to log a packet in the log buffer when it matches the access control entry (ACE). Matches are logged if you also configure the **matchlog** keyword in the **ip arp inspection vlan logging** global configuration command. For more information, see the "Configuring the Log Buffer" section on page 19-12.
Step 4	**exit**	Return to global configuration mode.
Step 5	**ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* [**static**]	Apply the ARP ACL to the VLAN. By default, no defined ARP ACLs are applied to any VLAN. • For *arp-acl-name*, specify the name of the ACL created in Step 2. • For *vlan-range*, specify the VLAN that the switches and hosts are in. You can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • (Optional) Specify **static** to treat implicit denies in the ARP ACL as explicit denies and to drop packets that do not match any previous clauses in the ACL. DHCP bindings are not used. If you do not specify this keyword, it means that there is no explicit deny in the ACL that denies the packet, and DHCP bindings determine whether a packet is permitted or denied if the packet does not match any clauses in the ACL. ARP packets containing only IP-to-MAC address bindings are compared against the ACL. Packets are permitted only if the access list permits them.
Step 6	**interface** *interface-id*	Specify the Switch A interface that is connected to Switch B, and enter interface configuration mode.

	Command	Purpose
Step 7	**no ip arp inspection trust**	Configure the Switch A interface that is connected to Switch B as untrusted. By default, all interfaces are untrusted. For untrusted interfaces, the switch intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The switch drops invalid packets and logs them in the log buffer according to the logging configuration specified with the **ip arp inspection vlan logging** global configuration command. For more information, see the "Configuring the Log Buffer" section on page 19-12.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show arp access-list** [*acl-name*] **show ip arp inspection vlan** *vlan-range* **show ip arp inspection interfaces**	Verify your entries.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the ARP ACL, use the **no arp access-list** global configuration command. To remove the ARP ACL attached to a VLAN, use the **no ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* global configuration command.

This example shows how to configure an ARP ACL called *host2* on Switch A, to permit ARP packets from Host 2 (IP address 1.1.1.1 and MAC address 0001.0001.0001), to apply the ACL to VLAN 1, and to configure port 1 on Switch A as untrusted:

```
Switch(config)# arp access-list host2
Switch(config-arp-acl)# permit ip host 1.1.1.1 mac host 1.1.1
Switch(config-arp-acl)# exit
Switch(config)# ip arp inspection filter host2 vlan 1
Switch(config)# interface gigabitethernet 0/1
Switch(config-if)# no ip arp inspection trust
```

Limiting the Rate of Incoming ARP Packets

The switch CPU performs dynamic ARP inspection validation checks; therefore, the number of incoming ARP packets is rate-limited to prevent a denial-of-service attack.

When the rate of incoming ARP packets exceeds the configured limit, the switch places the port in the error-disabled state. The port remains in that state until you enable error-disabled recovery so that ports automatically emerge from this state after a specified timeout period.

Note Unless you configure a rate limit on an interface, changing the trust state of the interface also changes its rate limit to the default value for that trust state. After you configure the rate limit, the interface retains the rate limit even when its trust state is changed. If you enter the **no ip arp inspection limit** interface configuration command, the interface reverts to its default rate limit.

For configuration guidelines for rate limiting trunk ports and EtherChannel ports, see the "Dynamic ARP Inspection Configuration Guidelines" section on page 19-6.

Beginning in privileged EXEC mode, follow these steps to limit the rate of incoming ARP packets. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be rate-limited, and enter interface configuration mode.
Step 3	**ip arp inspection limit** {**rate** *pps* [**burst interval** *seconds*] | **none**}	Limit the rate of incoming ARP requests and responses on the interface. The default rate is 15 pps on untrusted interfaces and unlimited on trusted interfaces. The burst interval is 1 second. The keywords have these meanings: • For **rate** *pps*, specify an upper limit for the number of incoming packets processed per second. The range is 0 to 2048 pps. • (Optional) For **burst interval** *seconds*, specify the consecutive interval in seconds, over which the interface is monitored for a high rate of ARP packets.The range is 1 to 15. • For **rate none**, specify no upper limit for the rate of incoming ARP packets that can be processed.
Step 4	**exit**	Return to global configuration mode.
Step 5	**errdisable recovery cause arp-inspection interval** *interval*	(Optional) Enable error recovery from the dynamic ARP inspection error-disable state. By default, recovery is disabled, and the recovery interval is 300 seconds. For **interval** *interval*, specify the time in seconds to recover from the error-disable state. The range is 30 to 86400.
Step 6	**exit**	Return to privileged EXEC mode.
Step 7	**show ip arp inspection interfaces** **show errdisable recovery**	Verify your settings.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default rate-limit configuration, use the **no ip arp inspection limit** interface configuration command. To disable error recovery for dynamic ARP inspection, use the **no errdisable recovery cause arp-inspection** global configuration command.

Performing Validation Checks

Dynamic ARP inspection intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. You can configure the switch to perform additional checks on the destination MAC address, the sender and target IP addresses, and the source MAC address.

Beginning in privileged EXEC mode, follow these steps to perform specific checks on incoming ARP packets. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip arp inspection validate** {[**src-mac**] [**dst-mac**] [**ip**]}	Perform a specific check on incoming ARP packets. By default, no checks are performed. The keywords have these meanings: • For **src-mac**, check the source MAC address in the Ethernet header against the sender MAC address in the ARP body. This check is performed on both ARP requests and responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped. • For **dst-mac**, check the destination MAC address in the Ethernet header against the target MAC address in ARP body. This check is performed for ARP responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped. • For **ip**, check the ARP body for invalid and unexpected IP addresses. Addresses include 0.0.0.0, 255.255.255.255, and all IP multicast addresses. Sender IP addresses are checked in all ARP requests and responses, and target IP addresses are checked only in ARP responses. You must specify at least one of the keywords. Each command overrides the configuration of the previous command; that is, if a command enables **src** and **dst mac** validations, and a second command enables IP validation only, the **src** and **dst mac** validations are disabled as a result of the second command.
Step 3	**exit**	Return to privileged EXEC mode.
Step 4	**show ip arp inspection vlan** *vlan-range*	Verify your settings.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable checking, use the **no ip arp inspection validate** [**src-mac**] [**dst-mac**] [**ip**] global configuration command. To display statistics for forwarded, dropped, and MAC and IP validation failure packets, use the **show ip arp inspection statistics** privileged EXEC command.

Configuring the Log Buffer

When the switch drops a packet, it places an entry in the log buffer and then generates system messages on a rate-controlled basis. After the message is generated, the switch clears the entry from the log buffer. Each log entry contains flow information, such as the receiving VLAN, the port number, the source and destination IP addresses, and the source and destination MAC addresses.

A log-buffer entry can represent more than one packet. For example, if an interface receives many packets on the same VLAN with the same ARP parameters, the switch combines the packets as one entry in the log buffer and generates a single system message for the entry.

If the log buffer overflows, it means that a log event does not fit into the log buffer, and the display for the **show ip arp inspection log** privileged EXEC command is affected. A -- in the display appears in place of all data except the packet count and the time. No other statistics are provided for the entry. If you see this entry in the display, increase the number of entries in the log buffer or increase the logging rate.

Beginning in privileged EXEC mode, follow these steps to configure the log buffer. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip arp inspection log-buffer** {**entries** *number* | **logs** *number* **interval** *seconds*}	Configure the dynamic ARP inspection logging buffer. By default, when dynamic ARP inspection is enabled, denied or dropped ARP packets are logged. The number of log entries is 32. The number of system messages is limited to 5 per second. The logging-rate interval is 1 second. The keywords have these meanings: • For **entries** *number*, specify the number of entries to be logged in the buffer. The range is 0 to 1024. • For **logs** *number* **interval** *seconds*, specify the number of entries to generate system messages in the specified interval. For **logs** *number*, the range is 0 to 1024. A 0 value means that the entry is placed in the log buffer, but a system message is not generated. For **interval** *seconds*, the range is 0 to 86400 seconds (1 day). A 0 value means that a system message is immediately generated (and the log buffer is always empty). An interval setting of 0 overrides a log setting of 0. The **logs** and **interval** settings interact. If the **logs** *number* X is greater than **interval** *seconds* Y, X divided by Y (X/Y) system messages are sent every second. Otherwise, one system message is sent every Y divided by X (Y/X) seconds.
Step 3	**ip arp inspection vlan** *vlan-range* **logging** {**acl-match** {**matchlog** | **none**} | **dhcp-bindings** {**all** | **none** | **permit**}}	Control the type of packets that are logged per VLAN. By default, all denied or all dropped packets are logged. The term *logged* means the entry is placed in the log buffer and a system message is generated. The keywords have these meanings: • For *vlan-range*, specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For **acl-match matchlog**, log packets based on the ACE logging configuration. If you specify the **matchlog** keyword in this command and the **log** keyword in the **permit** or **deny** ARP access-list configuration command, ARP packets permitted or denied by the ACL are logged. • For **acl-match none**, do not log packets that match ACLs. • For **dhcp-bindings all**, log all packets that match DHCP bindings. • For **dhcp-bindings none**, do not log packets that match DHCP bindings. • For **dhcp-bindings permit**, log DHCP-binding permitted packets.
Step 4	**exit**	Return to privileged EXEC mode.

	Command	Purpose
Step 5	**show ip arp inspection log**	Verify your settings.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default log buffer settings, use the **no ip arp inspection log-buffer** {**entries** | **logs**} global configuration command. To return to the default VLAN log settings, use the **no ip arp inspection vlan** *vlan-range* **logging** {**acl-match** | **dhcp-bindings**} global configuration command. To clear the log buffer, use the **clear ip arp inspection log** privileged EXEC command.

Displaying Dynamic ARP Inspection Information

To display dynamic ARP inspection information, use the privileged EXEC commands described in Table 19-2:

Table 19-2 Commands for Displaying Dynamic ARP Inspection Information

Command	Description
show arp access-list [*acl-name*]	Displays detailed information about ARP ACLs.
show ip arp inspection interfaces [*interface-id*]	Displays the trust state and the rate limit of ARP packets for the specified interface or all interfaces.
show ip arp inspection vlan *vlan-range*	Displays the configuration and the operating state of dynamic ARP inspection for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with dynamic ARP inspection enabled (active).

To clear or display dynamic ARP inspection statistics, use the privileged EXEC commands in Table 19-3:

Table 19-3 Commands for Clearing or Displaying Dynamic ARP Inspection Statistics

Command	Description
clear ip arp inspection statistics	Clears dynamic ARP inspection statistics.
show ip arp inspection statistics [**vlan** *vlan-range*]	Displays statistics for forwarded, dropped, MAC validation failure, IP validation failure, ACL permitted and denied, and DHCP permitted and denied packets for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with dynamic ARP inspection enabled (active).

For the **show ip arp inspection statistics** command, the switch increments the number of forwarded packets for each ARP request and response packet on a trusted dynamic ARP inspection port. The switch increments the number of ACL or DHCP permitted packets for each packet that is denied by source MAC, destination MAC, or IP validation checks, and the switch increments the appropriate failure count.

To clear or display dynamic ARP inspection logging information, use the privileged EXEC commands in Table 19-4:

Table 19-4 Commands for Clearing or Displaying Dynamic ARP Inspection Logging Information

Command	Description
clear ip arp inspection log	Clears the dynamic ARP inspection log buffer.
show ip arp inspection log	Displays the configuration and contents of the dynamic ARP inspection log buffer.

For more information about these commands, see the command reference for this release.

CHAPTER 20

Configuring IGMP Snooping and MVR

This chapter describes how to configure Internet Group Management Protocol (IGMP) snooping on your Catalyst 3550 switch, including an application of local IGMP snooping, Multicast VLAN Registration (MVR). It also includes procedures for controlling multicast group membership by using IGMP filtering and procedures for configuring the IGMP throttling action.

Note For complete syntax and usage information for the commands used in this chapter, see the switch command reference for this release and the "IP Multicast Routing Commands" section in the *Cisco IOS IP Command Reference, Volume 3 of 3:Multicast, Release 12.2*.

This chapter consists of these sections:

- Understanding IGMP Snooping, page 20-1
- Configuring IGMP Snooping, page 20-6
- Displaying IGMP Snooping Information, page 20-14
- Understanding Multicast VLAN Registration, page 20-15
- Configuring MVR, page 20-18
- Displaying MVR Information, page 20-21
- Configuring IGMP Filtering and Throttling, page 20-22
- Displaying IGMP Filtering and Throttling Configuration, page 20-27

Note For MAC addresses that map to IP multicast groups, you can either manage them through features such as IGMP snooping and MVR, or you can use static MAC addresses. However, you cannot use both methods simultaneously. Therefore, before using IGMP snooping or MVR, you should remove all statically configured MAC addresses that map to IP multicast groups.

Understanding IGMP Snooping

Layer 2 switches can use IGMP snooping to constrain the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices. As the name implies, IGMP snooping requires the LAN switch to snoop on the IGMP transmissions between the host and the router and to keep track of multicast groups and member ports. When the switch receives an IGMP report from a host for a particular multicast group,

the switch adds the host port number to the forwarding table entry; when it receives an IGMP Leave Group message from a host, it removes the host port from the table entry. It also periodically deletes entries if it does not receive IGMP membership reports from the multicast clients.

Note For more information on IP multicast and IGMP, see RFC 1112 and RFC 2236.

The multicast router (which could be a Catalyst 3550 switch with the enhanced multilayer software image) sends out periodic general queries to all VLANs. All hosts interested in this multicast traffic send join requests and are added to the forwarding table entry. The switch forwards only one join request per IP multicast group to the multicast router. It creates one entry per VLAN in the Layer 2 forwarding table for each MAC group from which it receives an IGMP join request.

Layer 2 multicast groups learned through IGMP snooping are dynamic. However, you can statically configure MAC multicast groups by using the **ip igmp snooping vlan static** global configuration command. If you specify group membership for a multicast group address statically, your setting supersedes any automatic manipulation by IGMP snooping. Multicast group membership lists can consist of both user-defined and IGMP snooping-learned settings.

If a port spanning-tree, a port group, or a VLAN ID change occurs, the IGMP snooping-learned multicast groups from this port on the VLAN are deleted.

These sections describe characteristics of IGMP snooping on the switch:

- IGMP Versions, page 20-2
- Joining a Multicast Group, page 20-3
- Leaving a Multicast Group, page 20-4
- Immediate-Leave Processing, page 20-5
- IGMP Configurable-Leave Timer, page 20-5
- IGMP Report Suppression, page 20-5
- Source-Only Networks, page 20-6

IGMP Versions

The switch supports IGMP version 1, IGMP version 2, and IGMP version 3. These versions are interoperable on the switch. For example, if IGMP snooping is enabled on an IGMPv2 switch and the switch receives an IGMPv3 report from a host, the switch can forward the IGMPv3 report to the multicast router.

Note The switch supports IGMPv3 snooping based only on the destination multicast MAC address. It does not support snooping based on the source MAC address or on proxy reports.

An IGMPv3 switch supports Basic IGMPv3 Snooping Support (BISS), which includes support for the snooping features on IGMPv1 and IGMPv2 switches and for IGMPv3 membership report messages. BISS constrains the flooding of multicast traffic when your network includes IGMPv3 hosts. It constrains traffic to approximately the same set of ports as the IGMP snooping feature on IGMPv2 or IGMPv1 hosts.

Note IGMPv3 join and leave messages are not supported on switches running IGMP filtering or MVR.

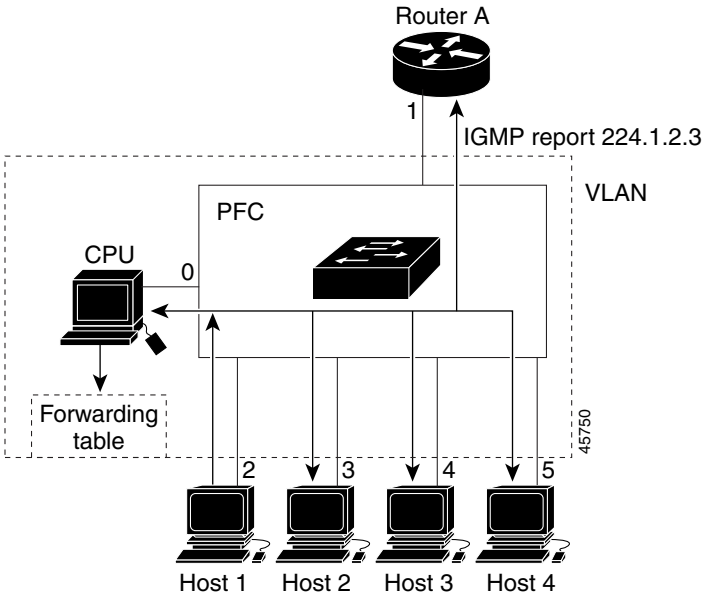
An IGMPv3 switch can receive messages from and forward messages to a device running the Source Specific Multicast (SSM) feature. For more information, see the "Configuring IP Multicast Layer 3 Switching" chapter in the *Catalyst 4500 Series Switch Cisco IOS Software Configuration Guide, Cisco IOS Release 12.1(12c)EW* at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/lan/cat4000/12_1_12/config/mcastmls.htm

Joining a Multicast Group

When a host connected to the switch wants to join an IP multicast group, it sends an unsolicited IGMP join message, specifying the IP multicast group to join. Alternatively, when the switch receives a general query from the router, it forwards the query to all ports in the VLAN. Hosts wanting to join the multicast group respond by sending a join message to the switch. The switch CPU creates a multicast forwarding-table entry for the group if it is not already present. The CPU also adds the interface where the join message was received to the forwarding-table entry. The host associated with that interface receives multicast traffic for that multicast group. See Figure 20-1.

Figure 20-1 Initial IGMP Join Message



Router A sends a general query to the switch, which forwards the query to ports 2 through 5, all members of the same VLAN. Host 1 wants to join multicast group 224.1.2.3 and multicasts an IGMP membership report (IGMP join message) to the group with the equivalent MAC destination address of 0x0100.5E01.0203. When the CPU receives the IGMP report multicast by Host 1, the CPU uses the information in the IGMP report to set up a forwarding-table entry, as shown in Table 20-1, that includes the port numbers of Host 1, the router, and the switch internal CPU.

Table 20-1 IGMP Snooping Forwarding Table

Destination Address	Type of Packet	Ports
0100.5exx.xxxx	IGMP	0
0100.5e01.0203	!IGMP	1, 2

Note that the switch hardware can distinguish IGMP information packets from other packets for the multicast group.

- The first entry in the table tells the switching engine to send IGMP packets to only the switch CPU. This prevents the CPU from becoming overloaded with multicast frames.
- The second entry tells the switching engine to send frames addressed to the 0x0100.5E01.0203 multicast MAC address that are not IGMP packets (!IGMP) to the router and to the host that has joined the group.

If another host (for example, Host 4) sends an unsolicited IGMP join message for the same group (Figure 20-2), the CPU receives that message and adds the port number of Host 4 to the forwarding table as shown in Table 20-2. Note that because the forwarding table directs IGMP messages only to the CPU, the message is not flooded to other ports on the switch. Any known multicast traffic is forwarded to the group and not to the CPU.

Figure 20-2 Second Host Joining a Multicast Group

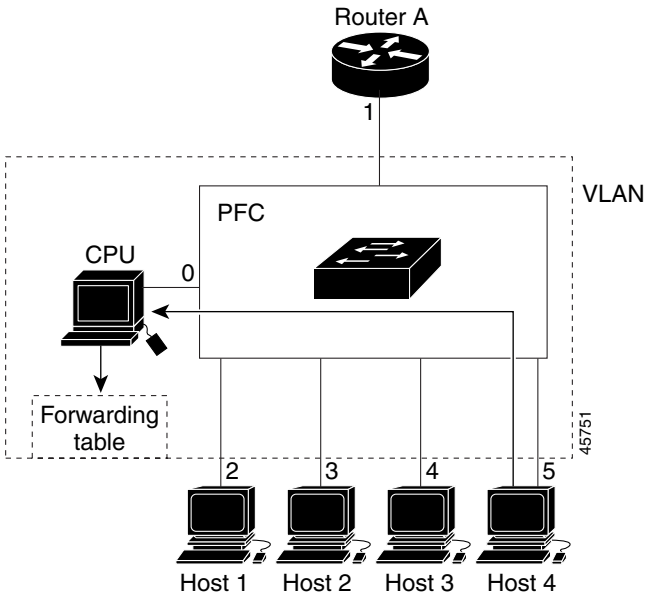


Table 20-2 Updated IGMP Snooping Forwarding Table

Destination Address	Type of Packet	Ports
0100.5exx.xxxx	IGMP	0
0100.5e01.0203	!IGMP	1, 2, 5

Leaving a Multicast Group

The router sends periodic multicast general queries and the switch forwards these queries through all ports in the VLAN. Interested hosts respond to the queries. If at least one host in the VLAN wishes to receive multicast traffic, the router continues forwarding the multicast traffic to the VLAN. The switch forwards multicast group traffic to only those hosts listed in the forwarding table for that Layer 2 multicast group.

When hosts want to leave a multicast group, they can either silently leave, or they can send a leave message. When the switch receives a leave message from a host, it sends a group-specific query to determine if any other devices connected to that interface are interested in traffic for the specific multicast group. The switch then updates the forwarding table for that MAC group so that only those hosts interested in receiving multicast traffic for the group are listed in the forwarding table. If the router receives no reports from a VLAN, it removes the group for the VLAN from its IGMP cache.

Immediate-Leave Processing

Immediate Leave is only supported with IGMP version 2 hosts.

The switch uses IGMP snooping Immediate-Leave processing to remove from the forwarding table an interface that sends a leave message without the switch sending group-specific queries to the interface. The VLAN interface is pruned from the multicast tree for the multicast group specified in the original leave message. Immediate-Leave processing ensures optimal bandwidth management for all hosts on a switched network, even when multiple multicast groups are simultaneously in use.

Note You should only use the Immediate-Leave processing feature on VLANs where a single host is connected to each port. If Immediate Leave is enabled in VLANs where more than one host is connected to a port, some hosts might inadvertently be dropped.

IGMP Configurable-Leave Timer

In Cisco IOS Release 12.2(25)SEA and earlier, the IGMP snooping leave time was fixed at 5 seconds. If membership reports were not received by the switch before the query response time of the query expired, a port was removed from the multicast group membership. However, some applications require a leave latency of less than 5 seconds.

In Cisco IOS Release 12.2(25)SEB and later, you can configure the time that the switch waits after sending a group-specific query to determine if hosts are still interested in a specific multicast group. The IGMP leave response time can be configured from 100 to 5000 milliseconds. The timer can be set either globally or on a per-VLAN basis. The VLAN configuration of the leave time overrides the global configuration.

IGMP Report Suppression

Note IGMP report suppression is supported only when the multicast query has IGMPv1 and IGMPv2 reports. This feature is not supported when the query includes IGMPv3 reports.

The switch uses IGMP report suppression to forward only one IGMP report per multicast router query to multicast devices. When IGMP router suppression is enabled (the default), the switch sends the first IGMP report from all hosts for a group to all the multicast routers. The switch does not send the remaining IGMP reports for the group to the multicast routers. This feature prevents duplicate reports from being sent to the multicast devices.

If the multicast router query includes requests only for IGMPv1 and IGMPv2 reports, the switch forwards only the first IGMPv1 or IGMPv2 report from all hosts for a group to all the multicast routers.

If the multicast router query also includes requests for IGMPv3 reports, the switch forwards all IGMPv1, IGMPv2, and IGMPv3 reports for a group to the multicast devices.

If you disable IGMP report suppression, all IGMP reports are forwarded to the multicast routers.

Source-Only Networks

In a source-only network, switch ports are connected to multicast source ports and multicast router ports. The switch ports are not connected to hosts that send IGMP join or leave messages.

The switch learns about IP multicast groups that alias with reserved, destination, multicast IP addresses (224.0.0.x) from the IP multicast data stream by using the source-only learning method. The switch forwards traffic that aliases with these multicast addresses only to the multicast router ports.

The default learning method for traffic that aliases with reserved, destination, multicast IP addresses is IP multicast-source-only learning.Traffic that does not alias with these multicast addresses is forwarded to both the multicast source ports and multicast router ports. You cannot disable IP multicast-source-only learning for the traffic with reserved, destination, multicast IP addresses.

By default, the switch ages out forwarding-table entries that were learned by the source-only learning method and that are not in use. If the aging time is too long or is disabled, the forwarding table is filled with unused entries that the switch learned by using source-only learning or by using the IGMP join messages. When the switch receives traffic for new IP multicast groups, it floods the packet to all ports in the same VLAN. This unnecessary flooding can impact switch performance.

If aging is disabled and you want to delete multicast addresses that the switch learned by using source-only learning, re-enable aging of the forwarding-table entries. The switch can now age out the multicast addresses that were learned by the source-only learning method and are not in use.

Configuring IGMP Snooping

IGMP snooping allows switches to examine IGMP packets and make forwarding decisions based on their content. To enable IGMP snooping on the switch to discover external multicast routers, the Layer 3 interfaces on the routers in the VLAN must already have been configured for multicast routing. For more information, see Chapter 3, “Configuring IP Multicast Routing.”

These sections describe how to configure IGMP snooping:

- Default IGMP Snooping Configuration, page 20-7
- Enabling or Disabling IGMP Snooping, page 20-7
- Setting the Snooping Method, page 20-8
- Configuring a Multicast Router Port, page 20-9
- Configuring a Host Statically to Join a Group, page 20-9
- Enabling IGMP Immediate-Leave Processing, page 20-10
- Configuring the IGMP Leave Timer, page 20-11
- Configuring TCN-Related Commands, page 20-11
- Disabling IGMP Report Suppression, page 20-13
- Configuring the Aging Time, page 20-14
- Displaying IGMP Snooping Information, page 20-14

Default IGMP Snooping Configuration

Table 20-3 shows the default IGMP snooping configuration.

Table 20-3 Default IGMP Snooping Configuration

Feature	Default Setting
IGMP snooping	Enabled globally and per VLAN.
Multicast routers	None configured.
Multicast router learning (snooping) method	PIM-DVMRP.
IGMP snooping Immediate Leave	Disabled.
Static groups	None configured.
Topology change notification flood query count	2
Topology change notification query solicitation	Disabled
Aging forward-table entries (for traffic that aliases with reserved, destination, multicast IP addresses)	Enabled. The default is 600 seconds (10 minutes).
IGMP report suppression	Enabled.

Enabling or Disabling IGMP Snooping

By default, IGMP snooping is globally enabled on the switch. When globally enabled or disabled, it is also enabled or disabled in all existing VLAN interfaces. IGMP snooping is by default enabled on all VLANs, but can be enabled and disabled on a per-VLAN basis. After you configure the VLAN interface for multicast routing, no configuration is needed for the switch to dynamically access external multicast routers by using IGMP snooping.

Global IGMP snooping overrides the VLAN IGMP snooping. If global snooping is disabled, you cannot enable VLAN snooping. If global snooping is enabled, you can enable or disable VLAN snooping.

Beginning in privileged EXEC mode, follow these steps to globally enable IGMP snooping on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp snooping**	Globally enable IGMP snooping in all existing VLAN interfaces.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To globally disable IGMP snooping on all VLAN interfaces, use the **no ip igmp snooping** global configuration command.

Beginning in privileged EXEC mode, follow these steps to enable IGMP snooping on a VLAN interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp snooping vlan** *vlan-id*	Enable IGMP snooping on the VLAN interface.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IGMP snooping on a VLAN interface, use the **no ip igmp snooping vlan** *vlan-id* global configuration command for the specified VLAN number**.**

Setting the Snooping Method

Multicast-capable router ports are added to the forwarding table for every Layer 2 multicast entry. The switch learns of such ports through one of these methods:

- Snooping on IGMP queries, Protocol Independent Multicast (PIM) packets, and Distance Vector Multicast Routing Protocol (DVMRP) packets
- Listening to Cisco Group Management Protocol (CGMP) packets from other routers
- Statically connecting to a multicast router port with the **ip igmp snooping mrouter** global configuration command

You can configure the switch either to snoop on IGMP queries and PIM/DVMRP packets or to listen to CGMP self-join or proxy-join packets. By default, the switch snoops on PIM/DVMRP packets on all VLANs. To learn of multicast router ports through only CGMP packets, use the **ip igmp snooping vlan** *vlan-id* **mrouter learn cgmp** global configuration command. When this command is entered, the router listens to only CGMP self-join and CGMP proxy-join packets and no other CGMP packets. To learn of multicast router ports through only PIM-DVMRP packets, use the **ip igmp snooping vlan** *vlan-id* **mrouter learn pim-dvmrp** global configuration command.

Note If you want to use CGMP as the learning method and no multicast routers in the VLAN are CGMP proxy-enabled, you must enter the **ip cgmp router-only** command to dynamically access the router. For more information, see Chapter 3, “Configuring IP Multicast Routing.”

Beginning in privileged EXEC mode, follow these steps to alter the method in which a VLAN interface dynamically accesses a multicast router:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp snooping vlan** *vlan-id* **mrouter learn** {**cgmp** | **pim-dvmrp**}	Enable IGMP snooping on a VLAN. The VLAN ID range is 1 to 4094. Specify the multicast router learning method: • **cgmp**—Listen for CGMP packets. This method is useful for reducing control traffic. • **pim-dvmrp**—Snoop on IGMP queries and PIM-DVMRP packets. This is the default.
Step 3	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 4	**show ip igmp snooping**	Verify the configuration.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to configure IGMP snooping to use CGMP packets as the learning method:

```
Switch# configure terminal
Switch(config)# ip igmp snooping vlan 1 mrouter learn cgmp
Switch(config)# end
```

To return to the default learning method, use the **no ip igmp snooping vlan** *vlan-id* **mrouter learn cgmp** global configuration command.

Configuring a Multicast Router Port

To add a multicast router port (add a static connection to a multicast router), use the **ip igmp snooping vlan mrouter** global configuration command on the switch.

Note Static connections to multicast routers are supported only on switch ports.

Beginning in privileged EXEC mode, follow these steps to enable a static connection to a multicast router:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp snooping vlan** *vlan-id* **mrouter interface** *interface-id*	Specify the multicast router VLAN ID and specify the interface to the multicast router. The VLAN ID range is 1 to 4094.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip igmp snooping mrouter** [**vlan** *vlan-id*]	Verify that IGMP snooping is enabled on the VLAN interface.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a multicast router port from the VLAN, use the **no ip igmp snooping vlan** *vlan-id* **mrouter interface** *interface-id* global configuration command.

This example shows how to enable a static connection to a multicast router and verify the configuration:

```
Switch# configure terminal
Switch(config)# ip igmp snooping vlan 200 mrouter interface gigabitethernet0/1
Switch(config)# end
```

Configuring a Host Statically to Join a Group

Hosts or Layer 2 ports normally join multicast groups dynamically, but you can also statically configure a host on an interface.

Beginning in privileged EXEC mode, follow these steps to add a Layer 2 port as a member of a multicast group:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**ip igmp snooping vlan** *vlan-id* **static** *mac-address* **interface** *interface-id*	Statically configure a Layer 2 port as a member of a multicast group: • *vlan-id* is the multicast group VLAN ID. • *mac-address* is the group MAC address. • *interface-id* is the member port.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip igmp snooping groups**	Verify the member port and the IP address.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the Layer 2 port from the multicast group, use the **no ip igmp snooping vlan** *vlan-id* **static** *mac-address* **interface** *interface-id* global configuration command.

This example shows how to statically configure a host on an interface and verify the configuration:

```
Switch# configure terminal
Switch(config)# ip igmp snooping vlan 1 static 0100.5e00.0203 interface gigabitethernet0/1
Switch(config)# end
```

Enabling IGMP Immediate-Leave Processing

When you enable IGMP Immediate-Leave processing, the switch immediately removes a port when it detects an IGMP version 2 leave message on that port. You should use the Immediate-Leave feature only when there is a single receiver present on every port in the VLAN.

Note Immediate Leave is supported with only IGMP version 2 hosts.

Beginning in privileged EXEC mode, follow these steps to enable IGMP Immediate-Leave processing:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**ip igmp snooping vlan** *vlan-id* **immediate-leave**	Enable IGMP Immediate-Leave processing on the VLAN interface.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip igmp snooping vlan** *vlan-id*	Verify that Immediate Leave is enabled on the VLAN.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IGMP Immediate-Leave on a VLAN, use the **no ip igmp snooping vlan** *vlan-id* **immediate-leave** global configuration command.

This example shows how to enable IGMP immediate-leave processing on VLAN 130:

```
Switch# configure terminal
Switch(config)# ip igmp snooping vlan 130 immediate-leave
Switch(config)# end
```

Configuring the IGMP Leave Timer

Follows these guidelines when configuring the IGMP leave timer:

- You can configure the leave time globally or on a per-VLAN basis.
- Configuring the leave time on a VLAN overrides the global setting.
- The default leave time is 1000 milliseconds.
- The IGMP configurable leave time is only supported on hosts running IGMP Version 2.
- The actual leave latency in the network is usually the configured leave time. However, the leave time *might* vary around the configured time, depending on real-time CPU load conditions, network delays and the amount of traffic sent through the interface.

Beginning in privileged EXEC mode, follow these steps to enable the IGMP configurable-leave timer:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp snooping last-member-query-interval** *time*	Configure the IGMP leave timer globally. The range is 100 to 5000 milliseconds.
Step 3	**ip igmp snooping vlan** *vlan-id* **last-member-query-interval** *time*	(Optional) Configure the IGMP leave time on the VLAN interface. The range is 100 to 5000 milliseconds. **Note** Configuring the leave time on a VLAN overrides the globally configured timer.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip igmp snooping**	(Optional) Display the configured IGMP leave time.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip igmp snooping last-member-query-interval** global configuration command to globally reset the IGMP leave timer to the default setting (1000 milliseconds).

Use the **no ip igmp snooping vlan** *vlan-id* **last-member-query-interval global configuration command to** remove the configured IGMP leave-time setting from the specified VLAN.

Configuring TCN-Related Commands

These sections describe how to control flooded multicast traffic during a TCN event:

- Controlling the Multicast Flooding Time After a TCN Event, page 20-12
- Recovering from Flood Mode, page 20-12
- Disabling Multicast Flooding During a TCN Event, page 20-13

Controlling the Multicast Flooding Time After a TCN Event

You can control the time that multicast traffic is flooded after a TCN event by using the **ip igmp snooping tcn flood query count** global configuration command. This command configures the number of general queries for which multicast data traffic is flooded after a TCN event. Some examples of TCN events are when the client changed its location and the receiver is on same port that was blocked but is now forwarding, and when a port went down without sending a leave message.

If you set the TCN flood query count to 1 by using the **ip igmp snooping tcn flood query count** command, the flooding stops after receiving one general query. If you set the count to 7, the flooding of multicast traffic due to the TCN event lasts until seven general queries are received. Groups are relearned based on the general queries received during the TCN event.

Beginning in privileged EXEC mode, follow these steps to configure the TCN flood query count:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp snooping tcn flood query count** *count*	Specify the number of IGMP general queries for which the multicast traffic is flooded. The range is 1 to 10. By default, the flooding query count is 2.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip igmp snooping**	Verify the TCN settings.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default flooding query count, use the **no ip igmp snooping tcn flood query count** global configuration command.

Recovering from Flood Mode

When a topology change occurs, the spanning-tree root sends a special IGMP leave message (also known as global leave) with the group multicast address 0.0.0.0. However, when you enable the **ip igmp snooping tcn query solicit** global configuration command, the switch sends the global leave message whether or not it is the spanning-tree root. When the router receives this special leave, it immediately sends general queries, which expedite the process of recovering from the flood mode during the TCN event. Leaves are always sent if the switch is the spanning-tree root regardless of this configuration command. By default, query solicitation is disabled.

Beginning in privileged EXEC mode, follow these steps to enable the switch to send the global leave message whether or not it is the spanning-tree root:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp snooping tcn query solicit**	Send an IGMP leave message (global leave) to speed the process of recovering from the flood mode caused during a TCN event. By default, query solicitation is disabled.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip igmp snooping**	Verify the TCN settings.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default query solicitation, use the **no ip igmp snooping tcn query solicit** global configuration command.

Disabling Multicast Flooding During a TCN Event

When the switch receives a TCN, multicast traffic is flooded to all the ports until two general queries are received. If the switch has many ports with attached hosts that are subscribed to different multicast groups, this flooding might exceed the capacity of the link and cause packet loss. You can use the **ip igmp snooping tcn flood** interface configuration command to control this behavior.

Beginning in privileged EXEC mode, follow these steps to disable multicast flooding on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**no ip igmp snooping tcn flood**	Disable the flooding of multicast traffic during a spanning-tree TCN event. By default, multicast flooding is enabled on an interface.
Step 4	**exit**	Return to privileged EXEC mode.
Step 5	**show ip igmp snooping**	Verify the TCN settings.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To re-enable multicast flooding on an interface, use the **ip igmp snooping tcn flood** interface configuration command.

Disabling IGMP Report Suppression

IGMP report suppression is enabled by default. When it is enabled, the switch forwards only one IGMP report per multicast router query. When report suppression is disabled, all IGMP reports are forwarded to the multicast routers.

Note IGMP report suppression is supported only when the multicast query has IGMPv1 and IGMPv2 reports. This feature is not supported when the query includes IGMPv3 reports.

Beginning in privileged EXEC mode, follow these steps to disable IGMP report suppression:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no ip igmp snooping report-suppression**	Disable IGMP report suppression.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip igmp snooping**	Verify that IGMP report suppression is disabled.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To re-enable IGMP report suppression, use the **ip igmp snooping report-suppression** global configuration command.

Configuring the Aging Time

You can set the aging time for forwarding-table entries that the switch learns by using the IP multicast-source-only learning method.

Beginning in privileged EXEC mode, follow these steps to configure the aging time:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**ip igmp snooping source-only learning age-timer** *time*	Set the aging time. The range is 0 to 2880 seconds. The default is 600 seconds (10 minutes).
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config | include source-only-learning**	Verify the aging time.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the aging of the forwarding table entries, enter the **ip igmp snooping source-only-learning age-timer 0** global configuration command.

Displaying IGMP Snooping Information

You can display IGMP snooping information for dynamically learned and statically configured router ports and VLAN interfaces. You can also display MAC address multicast entries for a VLAN configured for IGMP snooping.

To display IGMP snooping information, use one or more of the privileged EXEC commands in Table 20-4.

Table 20-4 Commands for Displaying IGMP Snooping Information

Command	Purpose
show ip igmp snooping [**vlan** *vlan-id*]	Display the snooping configuration information for all VLANs on the switch or for a specified VLAN. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN.
show ip igmp snooping groups [**count** | **vlan** *vlan-id* [*ip_address* | **count**]]	Display multicast table information for the switch, for a multicast VLAN, or about a specific parameter: • **count**—Display the total number of entries for the specified command options instead of the actual entries. • *ip_address*—Display information for multicast group with the specified group IP address.
show ip igmp snooping mrouter [**vlan** *vlan-id*]	Display information on dynamically learned and manually configured multicast router interfaces. **Note** When you enable IGMP snooping, the switch automatically learns the interface to which a multicast router is connected. These are dynamically learned interfaces. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN.
show mac address-table multicast [**vlan** *vlan-id*] [**user** | **igmp-snooping**] [**count**]	Display the Layer 2 MAC address table entries for a VLAN. The keywords are all optional and limit the display as shown: • **vlan** *vlan-id*—Displays only the specified multicast group VLAN. • **user**—Displays only the user-configured multicast entries. • **igmp-snooping**—Displays only entries learned through IGMP snooping. • **count**—Displays only the total number of entries for the selected criteria, not the actual entries.

For more information about the keywords and options in these commands, see the command reference for this release.

For examples of output from the commands in Table 20-4, see the command reference for this release.

Understanding Multicast VLAN Registration

Multicast VLAN Registration (MVR) is designed for applications using wide-scale deployment of multicast traffic across an Ethernet ring-based service provider network (for example, the broadcast of multiple television channels over a service-provider network). MVR allows a subscriber on a port to subscribe and unsubscribe to a multicast stream on the network-wide multicast VLAN. It allows the single multicast VLAN to be shared in the network while subscribers remain in separate VLANs. MVR provides the ability to continuously send multicast streams in the multicast VLAN, but to isolate the streams from the subscriber VLANs for bandwidth and security reasons.

MVR assumes that subscriber ports subscribe and unsubscribe (join and leave) these multicast streams by sending out IGMP join and leave messages. These messages can originate from an IGMP version-2-compatible host with an Ethernet connection. Although MVR operates on the underlying mechanism of IGMP snooping, the two features operate independently of each other. One can be enabled

or disabled without affecting the behavior of the other feature. However, if IGMP snooping and MVR are both enabled, MVR reacts only to join and leave messages from multicast groups configured under MVR. Join and leave messages from all other multicast groups are managed by IGMP snooping.

The switch CPU identifies the MVR IP multicast streams and their associated MAC addresses in the switch forwarding table, intercepts the IGMP messages, and modifies the forwarding table to include or remove the subscriber as a receiver of the multicast stream, even though the receivers might be in a different VLAN from the source. This forwarding behavior selectively allows traffic to cross between different VLANs.

The switch has these modes of MVR operation: dynamic and compatible.

- When operating in MVR dynamic mode, the switch performs standard IGMP snooping. IGMP information packets are sent to the switch CPU, but multicast data packets are not sent to the CPU. Dynamic mode allows the multicast router to run normally because the switch sends the IGMP join messages to the router, and the router forwards multicast streams for a particular group to an interface only if it has received a join message from the interface for the group. Receiver ports are treated as members of the multicast VLAN for MVR multicast control and data traffic. IGMP reports for MVR groups are sent out source ports in the multicast VLAN.
- When in MVR compatible mode, MVR on the Catalyst 3550 switch interoperates with MVR on Catalyst 3500 XL and Catalyst 2900 XL switches. It works the same as dynamic mode for all multicast data packets and IGMP query and leave packets. However, received IGMP report packets for MVR groups are not sent out on the multicast VLAN source ports. In contrast to dynamic mode, the switch does not send join messages to the router. The router must be statically configured for the interface to receive the multicast stream. Therefore, in this mode, MVR does not support dynamic membership joins on source ports.

Note IGMPv3 join and leave messages are not supported on switches running MVR.

Using MVR in a Multicast Television Application

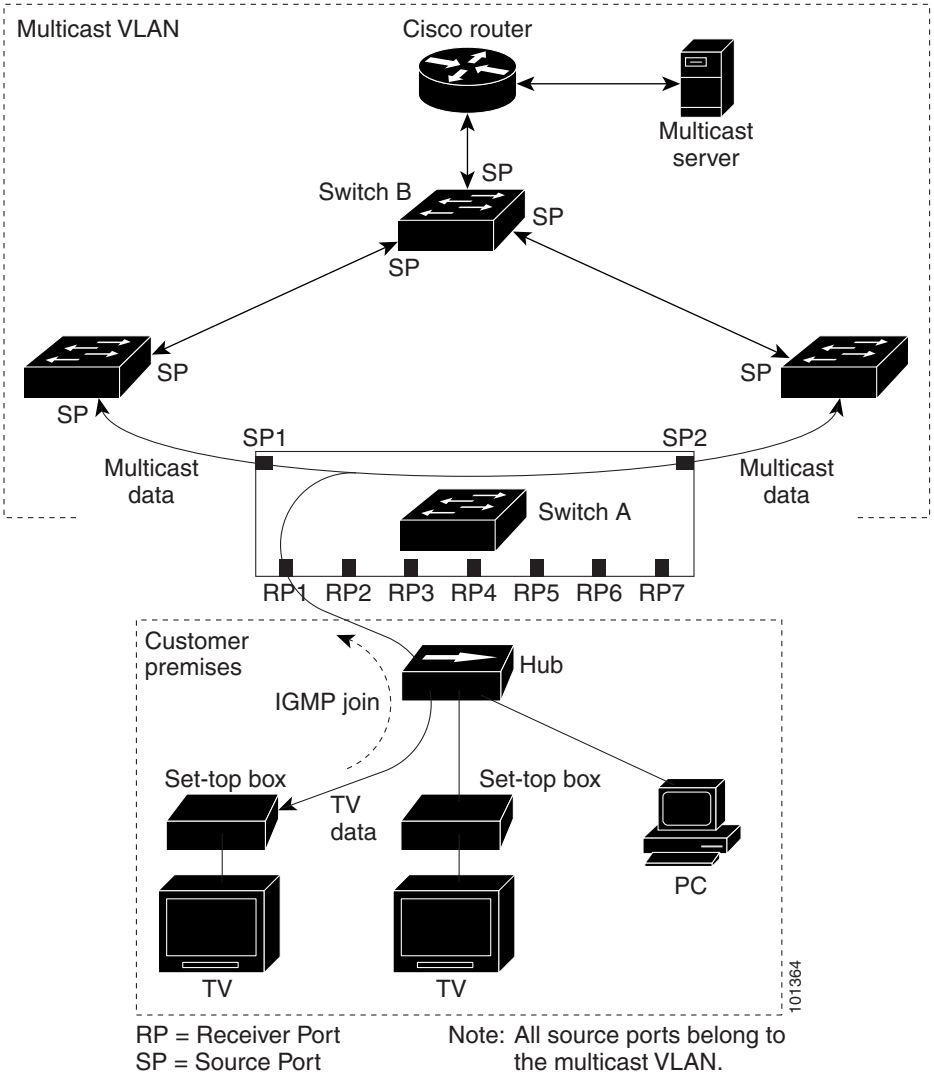
In a multicast television application, a PC or a television with a set-top box can receive the multicast stream. Multiple set-top boxes or PCs can be connected to one subscriber port, which is a switch port configured as an MVR receiver port. Figure 20-3 is an example configuration. DHCP assigns an IP address to the set-top box or the PC. When a subscriber selects a channel, the set-top box or PC sends an IGMP report to Switch A to join the appropriate multicast. If the IGMP report matches one of the configured multicast MAC addresses, the switch CPU modifies the hardware address table to include this receiver port and VLAN as a forwarding destination of the specified multicast stream when it is received from the multicast VLAN. Uplink ports that send and receive multicast data to and from the multicast VLAN are called MVR source ports.

When a subscriber changes channels or turns off the television, the set-top box sends an IGMP leave message for the multicast stream. The switch CPU sends an IGMP group-specific query through the receiver port VLAN. If there is another set-top box in the VLAN still subscribing to this group, that set-top box must respond within the maximum response time. If the CPU does not receive a response, it eliminates the receiver port as a forwarding destination for this group.

Without Immediate Leave, when the switch receives an IGMP leave message from a subscriber on a receiver port, it sends out an IGMP query on that port and waits for IGMP group membership reports. If no reports are received in a configured time period, the receiver port is removed from multicast group membership. With Immediate Leave, an IGMP query is not sent from the receiver port on which the

IGMP leave was received. As soon as the leave message is received, the receiver port is removed from multicast group membership, which speeds up leave latency. Enable the Immediate Leave feature only on receiver ports to which a single receiver device is connected.

Figure 20-3 Multicast VLAN Registration Example



MVR eliminates the need to duplicate television-channel multicast traffic for subscribers in each VLAN. Multicast traffic for all channels is only sent around the VLAN trunk once—only on the multicast VLAN. Although the IGMP leave and join message in the VLAN to which the subscriber port is assigned. These messages dynamically register for streams of multicast traffic in the multicast VLAN on the Layer 3 device. The access layer switch (Switch A) modifies the forwarding behavior to allow the traffic to be forwarded from the multicast VLAN to the subscriber port in a different VLAN, selectively allowing traffic to cross between two VLANs.

IGMP reports are sent to the same MAC addresses as the multicast data. The Switch A CPU must capture all IGMP join and leave messages from receiver ports and forward them to the multicast VLAN of the source (uplink) port.

Configuring MVR

These sections include basic MVR configuration information:

- Default MVR Configuration, page 20-18
- MVR Configuration Guidelines and Limitations, page 20-18
- Configuring MVR Global Parameters, page 20-19
- Configuring MVR Interfaces, page 20-20

Default MVR Configuration

Table 20-5 shows the default MVR configuration.

Table 20-5 Default MVR Configuration

Feature	Default Setting
MVR	Disabled globally and per interface
Multicast addresses	None configured
Query response time	0.5 second
Multicast VLAN	VLAN 1
Mode	Compatible
Interface (per port) default	Neither a receiver nor a source port
Immediate Leave	Disabled on all ports

MVR Configuration Guidelines and Limitations

Follow these guidelines when configuring MVR:

- Receiver ports cannot be trunk ports. Receiver ports on a switch can be in different VLANs, but should not belong to the multicast VLAN.
- The maximum number of multicast entries that can be configured on a switch (that is, the maximum number of television channels that can be received) is 256.
- Each channel is one multicast stream destined for a unique IP multicast address. These IP addresses cannot alias between themselves or with the reserved IP multicast addresses (in the range 224.0.0.xxx).
- Multicast routing and MVR cannot coexist on a switch. If you enable multicast routing and a multicast routing protocol while MVR is enabled, MVR is disabled, and you receive a warning message. If you try to enable MVR while multicast routing and a multicast routing protocol are enabled, the operation to enable MVR is cancelled, and you receive an error message.
- MVR does not support IGMPv3 messages.

Note For complete syntax and usage information for the commands used in this section, see the command reference for this release.

Configuring MVR Global Parameters

You do not need to set the optional MVR parameters if you choose to use the default settings. If you do want to change the default parameters (except for the MVR VLAN), you must first enable MVR.

Beginning in privileged EXEC mode, follow these steps to configure MVR parameters:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mvr**	Enable MVR on the switch.
Step 3	**mvr group** *ip-address* [*count*]	Configure an IP multicast address on the switch or use the *count* parameter to configure a contiguous series of MVR group addresses (the range for *count* is 1 to 256; the default is 1). Any multicast data sent to this address is sent to all source ports on the switch and all receiver ports that have elected to receive data on that multicast address. Each multicast address would correspond to one television channel. **Note** Each IP address translates to a multicast 48-bit MAC address. If an IP address being configured translates (aliases) to a previously configured MAC address or to any reserved multicast MAC addresses, the command fails.
Step 4	**mvr querytime** *value*	(Optional) Define the maximum time to wait for IGMP report memberships on a receiver port before removing the port from multicast group membership. The value is in units of tenths of a second. The range is from 1 to 100 and the default is 5 tenths or one-half second.
Step 5	**mvr vlan** *vlan-id*	(Optional) Specify the VLAN in which multicast data is received; all source ports must belong to this VLAN. The VLAN ID range is 1 to 4094. The default is VLAN 1.
Step 6	**mvr mode** {**dynamic** | **compatible**}	(Optional) Specify the MVR mode of operation: • **dynamic**—Allows dynamic MVR membership on source ports. • **compatible**—Is compatible with Catalyst 3500 XL and Catalyst 2900 XL switches and does not support IGMP dynamic joins on source ports. The default is **compatible** mode.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	show mvr or **show mvr members**	Verify the configuration.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the switch to its default settings, use the **no mvr** [**mode** | **group** *ip-address* | **querytime** | **vlan**] global configuration commands.

This example shows how to enable MVR, configure the MVR group address, set the query time to 1 second (10 tenths), specify the MVR multicast VLAN as VLAN 22, set the MVR mode as dynamic, and verify the results:

```
Switch(config)# mvr
Switch(config)# mvr group 228.1.23.4
```

```
Switch(config)# mvr querytime 10
Switch(config)# mvr vlan 22
Switch(config)# mvr mode dynamic
Switch(config)# end
Switch# show mvr
MVR Running: TRUE
MVR multicast vlan: 22
MVR Max Multicast Groups: 256
MVR Current multicast groups: 1
MVR Global query response time: 10 (tenths of sec)
MVR Mode: dynamic
```

You can use the **show mvr members** privileged EXEC command to verify the MVR multicast group addresses on the switch.

Configuring MVR Interfaces

Beginning in privileged EXEC mode, follow these steps to configure Layer 2 MVR interfaces:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mvr**	Enable MVR on the switch.
Step 3	**interface** *interface-id*	Enter the Layer 2 port to configure and enter interface configuration mode.
Step 4	**mvr type** {**source** | **receiver**}	Configure an MVR port as one of these: • **source**—Configure uplink ports that receive and send multicast data as source ports. Subscribers cannot be directly connected to source ports. All source ports on a switch belong to the single multicast VLAN. • **receiver**—Configure a port as a receiver port if it is a subscriber port and should only receive multicast data. It does not receive data unless it becomes a member of the multicast group, either statically or by using IGMP leave and join messages. Receiver ports cannot belong to the multicast VLAN. The default configuration is as a non-MVR port. If you attempt to configure a non-MVR port with MVR characteristics, the operation fails.
Step 5	**mvr vlan** *vlan-id* **group** *ip-address*	(Optional) Statically configure a port to receive multicast traffic sent to the multicast VLAN and the IP multicast address. A port statically configured as a member of a group remains a member of the group until statically removed. **Note** In compatible mode, this command applies to only receiver ports. In dynamic mode, it applies to receiver ports and source ports. Receiver ports can also dynamically join multicast groups by using IGMP join and leave messages.
Step 6	**mvr immediate**	(Optional) Enable the Immediate Leave feature of MVR on the port. **Note** This command applies to only receiver ports and should only be enabled on receiver ports to which a single receiver device is connected.
Step 7	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 8	show mvr show mvr interface or show mvr **members**	Verify the configuration.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the interface to its default settings, use the **no mvr** [**type** | **immediate** | **vlan** *vlan-id* | **group**] interface configuration commands.

This example shows how to configure a port as a receiver port, statically configure the port to receive multicast traffic sent to the multicast group address, configure Immediate Leave on the interface, and verify the results.

```
Switch(config)# mvr
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# mvr type receiver
Switch(config-if)# mvr vlan 22 group 228.1.23.4
Switch(config-if)# mvr immediate
Switch(config)# end
Switch# show mvr interface gigabitethernet0/1
Type: RECEIVER Status: ACTIVE Immediate Leave: ENABLED
```

Displaying MVR Information

You can display MVR information for the switch or for a specified interface. Beginning in privileged EXEC mode, use the commands in Table 20-6 to display MVR configuration:

Table 20-6 Commands for Displaying MVR Information

show mvr	Displays MVR status and values for the switch—whether MVR is enabled or disabled, the multicast VLAN, the maximum (256) and current (0 through 256) number of multicast groups, the query response time, and the MVR mode.
show mvr interface [*interface-id*] [**members** [**vlan** *vlan-**id***]]	Displays all MVR interfaces and their MVR configurations. When a specific interface is entered, displays this information: • Type—Receiver or Source • Status—One of these: – Active means the port is part of a VLAN. – Up/Down means that the port is forwarding or nonforwarding. – Inactive means that the port is not part of any VLAN. • Immediate Leave—Enabled or Disabled If the **members** keyword is entered, displays all multicast group members on this port or, if a VLAN identification is entered, all multicast group members on the VLAN. The VLAN ID range is 1 to 4094.
show mvr members [*ip-address*]	Displays all receiver and source ports that are members of any IP multicast group or the specified IP multicast group IP address.

Configuring IGMP Filtering and Throttling

In some environments, for example, metropolitan or multiple-dwelling unit (MDU) installations, you might want to control the set of multicast groups to which a user on a switch port can belong. You can control the distribution of multicast services, such as IP/TV, based on some type of subscription or service plan. You might also want to limit the number of multicast groups to which a user on a switch port can belong.

With the IGMP filtering feature, you can filter multicast joins on a per-port basis by configuring IP multicast profiles and associating them with individual switch ports. An IGMP profile can contain one or more multicast groups and specifies whether access to the group is permitted or denied. If an IGMP profile denying access to a multicast group is applied to a switch port, the IGMP join report requesting the stream of IP multicast traffic is dropped, and the port is not allowed to receive IP multicast traffic from that group. If the filtering action permits access to the multicast group, the IGMP report from the port is forwarded for normal processing.

IGMP filtering controls only group specific query and membership reports, including join and leave reports. It does not control general IGMP queries. IGMP filtering has no relationship with the function that directs the forwarding of IP multicast traffic. The filtering feature operates in the same manner whether CGMP or MVR is used to forward the multicast traffic.

You can also set the maximum number of IGMP groups that a Layer 2 interface can join.

With the IGMP throttling feature, you can also set the maximum number of IGMP groups that a Layer 2 interface can join. If the maximum number of IGMP groups is set, the IGMP snooping forwarding table contains the maximum number of entries, and the interface receives an IGMP join report, you can configure an interface to drop the IGMP report or to replace the randomly selected multicast entry with the received IGMP report.

Note IGMPv3 join and leave messages are not supported on switches running IGMP filtering.

These sections describe how to configure IGMP filtering and throttling:

- Default IGMP Filtering and Throttling Configuration, page 20-22
- Configuring IGMP Profiles, page 20-23 (optional)
- Applying IGMP Profiles, page 20-24 (optional)
- Setting the Maximum Number of IGMP Groups, page 20-25 (optional)
- Configuring the IGMP Throttling Action, page 20-25 (optional)

Default IGMP Filtering and Throttling Configuration

Table 20-7 shows the default IGMP filtering configuration.

Table 20-7 Default IGMP Filtering Configuration

Feature	Default Setting
IGMP filters	None applied
IGMP Maximum number of IGMP groups	No maximum set
IGMP profiles	None defined
IGMP profile action	Deny the range addresses

When the maximum number of groups is in forwarding table, the default IGMP throttling action is to deny the IGMP report. For configuration guidelines, see the "Configuring the IGMP Throttling Action" section on page 20-25.

Configuring IGMP Profiles

To configure an IGMP profile, use the **ip igmp profile** global configuration command with a profile number to create an IGMP profile and to enter IGMP profile configuration mode. From this mode, you can specify the parameters of the IGMP profile to be used for filtering IGMP join requests from a port. When you are in IGMP profile configuration mode, you can create the profile by using these commands:

- **deny**: Specifies that matching addresses are denied; this is the default condition.
- **exit**: Exits from igmp-profile configuration mode.
- **no**: Negates a command or sets its defaults.
- **permit**: Specifies that matching addresses are permitted.
- **range**: Specifies a range of IP addresses for the profile. You can enter a single IP address or a range with a start and an end address.

The default is for the switch to have no IGMP profiles configured. When a profile is configured, if neither the **permit** nor **deny** keyword is included, the default is to deny access to the range of IP addresses.

Beginning in privileged EXEC mode, follow these steps to create an IGMP profile:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip igmp profile** *profile number*	Enter IGMP profile configuration mode, and assign a number to the profile you are configuring. The range is 1 to 4294967295.
Step 3	**permit | deny**	(Optional) Set the action to permit or deny access to the IP multicast address. If no action is configured, the default for the profile is to deny access.
Step 4	**range** *ip multicast address*	Enter the IP multicast address or range of IP multicast addresses to which access is being controlled. If entering a range, enter the low IP multicast address, a space, and the high IP multicast address. You can use the **range** command multiple times to enter multiple addresses or ranges of addresses.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip igmp profile** *profile number*	Verify the profile configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete a profile, use the **no ip igmp profile** *profile number* global configuration command.

To delete an IP multicast address or range of IP multicast addresses, use the **no range** *ip multicast address* IGMP profile configuration command.

This example shows how to create IGMP profile 4 allowing access to the single IP multicast address and how to verify the configuration. If the action was to deny (the default), it would not appear in the **show ip igmp profile** output display.

```
Switch(config)# ip igmp profile 4
Switch(config-igmp-profile)# permit
Switch(config-igmp-profile)# range 229.9.9.0
Switch(config-igmp-profile)# end
Switch# show ip igmp profile 4
IGMP Profile 4
    permit
    range 229.9.9.0 229.9.9.0
```

Applying IGMP Profiles

To control access as defined in an IGMP profile, use the **ip igmp filter** interface configuration command to apply the profile to the appropriate interfaces. You can apply IGMP profiles to Layer 2 ports only; you cannot apply IGMP profiles to routed ports or SVIs. You cannot apply profiles to ports that belong to an EtherChannel port group. You can apply a profile to multiple interfaces, but each interface can only have one profile applied to it.

Beginning in privileged EXEC mode, follow these steps to apply an IGMP profile to a switch port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode by entering the physical interface to configure. The interface must be a Layer 2 port that does not belong to an EtherChannel port group.
Step 3	**ip igmp filter** *profile number*	Apply the specified IGMP profile to the interface. The profile number can be 1 to 4294967295.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running configuration interface** *interface-id*	Verify the configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a profile from an interface, use the **no ip igmp filter** *profile number* interface configuration command.

This example shows how to apply IGMP profile 4 to a port and verify the configuration.

```
Switch(config)# interface fastethernet0/2
Switch(config-if)# ip igmp filter 4
Switch(config-if)# end
Switch# show running-config interface fastethernet0/2
Building configuration...
```

```
Current configuration : 123 bytes
!
interface fastethernet0/2
 no ip address
 shutdown
 snmp trap link-status
 ip igmp max-groups 25
 ip igmp filter 4
end
```

Setting the Maximum Number of IGMP Groups

You can set the maximum number of IGMP groups that a Layer 2 interface can join by using the **ip igmp max-groups** interface configuration command. Use the **no** form of this command to set the maximum back to the default, which is no limit.

This restriction can be applied to Layer 2 ports only; you cannot set a maximum number of IGMP groups on routed ports or SVIs. You can use this command on an logical EtherChannel interface but cannot use it on ports that belong to an EtherChannel port group.

Beginning in privileged EXEC mode, follow these steps to set the maximum number of IGMP groups in the forwarding table:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode by entering the physical interface to configure. The interface can be a Layer 2 port that does not belong to an EtherChannel group or a EtherChannel interface.
Step 3	**ip igmp max-groups** *number*	Set the maximum number of IGMP groups that the interface can join. The range is 0 to 4294967294. The default is to have no maximum set.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-configuration interface** *interface-id*	Verify the configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the maximum group limitation and return to the default of no maximum, use the **no ip igmp max-groups** interface configuration command.

This example shows how to limit to 25 the number of IGMP groups that an interface can join.

```
Switch(config)# interface fastethernet0/2
Switch(config-if)# ip igmp max-groups 25
Switch(config-if)# end
```

Configuring the IGMP Throttling Action

After you set the maximum number of IGMP groups that a Layer 2 interface can join, you can configure an interface to replace the existing group with the new group for which the IGMP report was received by using the **ip igmp max-groups action replace** interface configuration command. Use the **no** form of this command to return to the default, which is to drop the IGMP join report.

Follow these guidelines when configuring the IGMP throttling action:

- This restriction can be applied to Layer 2 ports only; you can use this command on a logical EtherChannel interface but cannot use it on ports that belong to an EtherChannel port group.
- When the maximum group limitation is set to the default (no maximum), entering the **ip igmp max-groups action** {**deny** | **replace**} command has no effect.
- If you configure the throttling action and set the maximum group limitation after an interface has added multicast entries to the forwarding table, the forwarding-table entries are either aged out or removed, depending on the throttling action.
  - If you configure the throttling action as **deny**, the entries that were previously in the forwarding table are not removed but are aged out. After these entries are aged out and the maximum number of entries is in the forwarding table, the switch drops the next IGMP report received on the interface.
  - If you configure the throttling action as **replace**, the entries that were previously in the forwarding table are removed. When the maximum number of entries is in the forwarding table, the switch the switch replaces a randomly selected entry with the received IGMP report.

  To prevent the switch from removing the forwarding-table entries, you can configure the IGMP throttling action before an interface adds entries to the forwarding table.

Beginning in privileged EXEC mode, follow these steps to configure the throttling action when the maximum number of entries is in the forwarding table:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the physical interface to configure. The interface can be a Layer 2 port that does not belong to an EtherChannel group or an EtherChannel interface. The interface cannot be a trunk port.
Step 3	**ip igmp max-groups action** {**deny** | **replace**}	When an interface receives an IGMP report and the maximum number of entries is in the forwarding table, specify the action that the interface takes: • **deny**—Drop the report. • **replace**—Replace the existing group with the new group for which the IGMP report was received.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config interface** *interface-id*	Verify the configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default action of dropping the report, use the **no ip igmp max-groups action** interface configuration command.

Displaying IGMP Filtering and Throttling Configuration

You can display IGMP profile characteristics, and you can display the IGMP profile and maximum group configuration for all interfaces on the switch or for a specified interface. You can also display the IGMP throttling configuration for all interfaces on the switch or for a specified interface.

Use the privileged EXEC commands in Table 20-8 to display IGMP filtering and throttling configuration:

Table 20-8 Commands for Displaying IGMP Filtering and Throttling Configuration

Command	Purpose
how ip igmp profile [*profile number*]	Displays the specified IGMP profile or all the IGMP profiles defined on the switch.
show running-configuration [**interface** *interface-id*]	Displays the configuration of the specified interface or the configuration of all interfaces on the switch, including (if configured) the maximum number of IGMP groups to which an interface can belong and the IGMP profile applied to the interface.

CHAPTER 21

Configuring Port-Based Traffic Control

This chapter describes how to configure the port-based traffic control features on your Catalyst 3550 switch.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Configuring Storm Control, page 21-1
- Configuring Protected Ports, page 21-5
- Configuring Port Blocking, page 21-6
- Configuring Port Security, page 21-7
- Displaying Port-Based Traffic Control Settings, page 21-17

Configuring Storm Control

These sections include storm control configuration information and procedures:

- Understanding Storm Control, page 21-1
- Default Storm Control Configuration, page 21-3
- Configuring Storm Control and Threshold Levels, page 21-3

Understanding Storm Control

Storm control prevents traffic on a LAN from being disrupted by a broadcast, a multicast, or a unicast storm on one of the physical interfaces. A LAN storm occurs when packets flood the LAN, creating excessive traffic and degrading network performance. Errors in the protocol-stack implementation, mistakes in the network configuration, or users issuing a denial-of-service attack can cause a storm.

Storm control (or traffic suppression) monitors packets passing from an interface to the switching bus and determines if the packet is unicast, multicast, or broadcast. The switch counts the number of packets of a specified type received within the 1-second time interval and compares the measurement with a predefined suppression-level threshold.

Storm control uses one of these methods to measure traffic activity:

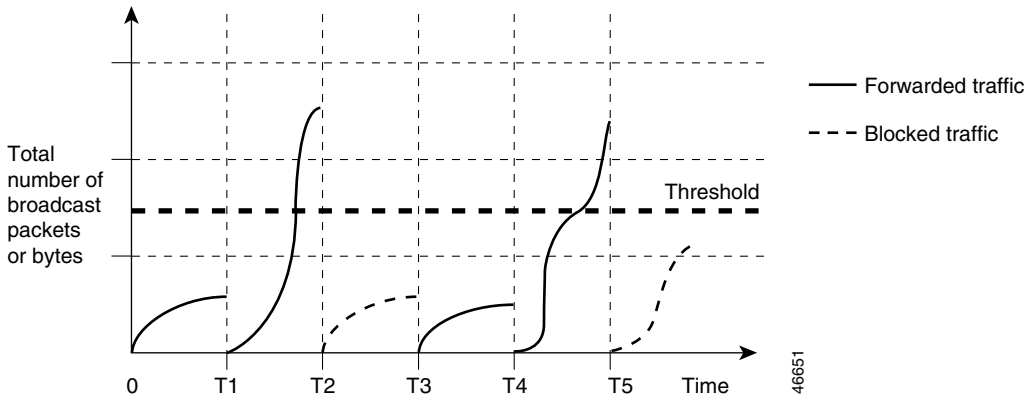
- Bandwidth as a percentage of the total available bandwidth of the port that can be used by the broadcast, multicast, or unicast traffic
- Traffic rate in packets per second at which broadcast, multicast, or unicast packets are received (Cisco IOS Release 12.1(22)EA1 or later)

With either method, the port blocks traffic when the rising threshold is reached. The port remains blocked until the traffic rate drops below the falling threshold (if one is specified) and then resumes normal forwarding. If the falling suppression level is not specified, the switch blocks all traffic until the traffic rate drops below the rising suppression level. In general, the higher the level, the less effective the protection against broadcast storms.

Note When the rate of multicast traffic exceeds a set threshold, all incoming traffic (broadcast, multicast, and unicast) is dropped until the level drops below the threshold level. Only spanning-tree packets are forwarded. When broadcast and unicast thresholds are exceeded, traffic is blocked for only the type of traffic that exceeded the threshold.

The graph in Figure 21-1 shows broadcast traffic patterns on an interface over a given period of time. The example can also be applied to multicast and unicast traffic. In this example, the broadcast traffic being forwarded exceeded the configured threshold between time intervals T1 and T2 and between T4 and T5. When the amount of specified traffic exceeds the threshold, all traffic of that kind is dropped for the next time period. Therefore, broadcast traffic is blocked during the intervals following T2 and T5. At the next time interval (for example, T3), if broadcast traffic does not exceed the threshold, it is again forwarded.

Figure 21-1 Broadcast Storm Control Example



The combination of the storm-control suppression level and the 1-second time interval control the way the storm control algorithm works. A higher threshold allows more packets to pass through.

Note Because packets do not arrive at uniform intervals, the 1-second time interval during which traffic activity is measured can affect the behavior of storm control.

You use the **storm-control** interface configuration commands to set the threshold value for each traffic type.

Note Before Cisco IOS Release 12.1(8)EA1, you set up storm control threshold values by using the **switchport broadcast**, **switchport multicast**, and **switchport unicast** interface configuration commands. These commands are now obsolete, replaced by the **storm-control** interface configuration commands.

Default Storm Control Configuration

By default, unicast, broadcast, and multicast storm control are disabled on the switch: that is, the suppression level is 100 percent (no limit is placed on the traffic).

Configuring Storm Control and Threshold Levels

You configure storm control on a port and enter the threshold level that you want to be used by a particular type of traffic.

However, because of hardware limitations and the way in which packets of different sizes are counted, threshold percentages are approximations. Depending on the sizes of the packets making up the incoming traffic, the actual enforced threshold might differ from the configured level by several percentage points.

Note Storm control is supported on physical interfaces. You can also configure storm control on an EtherChannel. When storm control is configured on an EtherChannel, the storm control settings propagate to the EtherChannel physical interfaces.

Beginning in privileged EXEC mode, follow these steps to configure storm control and threshold levels:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the physical interface to configure, and enter interface configuration mode.

	Command	Purpose
Step 3	**storm-control** {**broadcast** | **multicast** | **unicast**} **level** {*level* [*level-low*] | **pps** *pps* [*pps-low*]}	Configure broadcast, multicast, or unicast storm control. By default, storm control is disabled. The keywords have these meanings: • For *level*, specify the rising threshold level for broadcast, multicast, or unicast traffic as a percentage (up to two decimal places) of the bandwidth. The port blocks traffic when the rising threshold is reached. The range is **0.00 to 100.00**. • (Optional) For *level-low*, specify the falling threshold level as a percentage (up to two decimal places) of the bandwidth. **This value must be less than or equal to the rising suppression value.** The port forwards traffic when traffic drops below this level. If you do not configure a falling suppression level, it is set to the rising suppression level. The range is **0.00 to 100.00**. If you set the threshold to the maximum value (100 percent), no limit is placed on the traffic. If you set the threshold to 0 0, all broadcast, multicast, and unicast traffic on that port is blocked. • For **pps** *pps*, specify the rising threshold level for broadcast, multicast, or unicast traffic in packets per second (up to one decimal place). The port blocks traffic when the rising threshold is reached. The range is **0.0 to** 10000000000.0. • (Optional) For *pps-low*, specify the falling threshold level in packets per second (up to one decimal place). It can be less than or equal to the rising threshold level. The port forwards traffic when traffic drops below this level. The range is **0.0 to** 10000000000.0. For PPS settings, you can use metric suffixes such as k, m, and g for large number thresholds.
Step 4	**storm-control action** {**shutdown** | **trap**}	Specify the action to be taken when a storm is detected. The default is to filter out the traffic and to not send traps. • Select the **shutdown** keyword to error-disable the port during a storm. • Select the **trap** keyword to generate an SNMP trap when a storm is detected.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show storm-control** [*interface-id*] [**broadcast** | **multicast** | **unicast**]	Verify the storm control suppression levels set on the interface for the specified traffic type. If you do not enter a traffic type, broadcast storm control settings appear.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable storm control, use the **no storm-control** {**broadcast** | **multicast** | **unicast**} **level** interface configuration command.

This example shows how to enable unicast storm control on a port with an 87-percent rising suppression level and a 65-percent falling suppression level:

```
Switch# configure terminal
Switch(config)# interface fastethernet0/1
Switch(config-if)# storm-control unicast level 87 65
```

This example shows how to enable broadcast address storm control on a port to a level of 20 percent. When the broadcast traffic exceeds the configured level of 20 percent of the total available bandwidth of the port within the traffic-storm-control interval, the switch drops all broadcast traffic until the end of the traffic-storm-control interval:

```
Switch# configure terminal
Switch(config)# interface fastethernet0/1
Switch(config-if)# storm-control broadcast level 20
```

Configuring Protected Ports

Some applications require that no traffic be forwarded between ports on the same switch so that one neighbor does not see the traffic generated by another neighbor. In such an environment, the use of protected ports ensures that there is no exchange of unicast, broadcast, or multicast traffic between these ports on the switch.

Protected ports have these features:

- A protected port does not forward any traffic (unicast, multicast, or broadcast) to any other port that is also a protected port. Data traffic cannot be forwarded between protected ports at Layer 2; only control traffic, such as PIM packets, is forwarded because these packets are processed by the CPU and forwarded in software. All data traffic passing between protected ports must be forwarded through a Layer 3 device.
- Forwarding behavior between a protected port and a nonprotected port proceeds as usual.
- Protected ports are supported on IEEE 802.1Q trunks.

The default is to have no protected ports defined.

Note The protected port feature is not compatible with fallback bridging. When fallback bridging is enabled, it is possible for packets to be forwarded from one protected port on a switch to another protected port on the same switch if the ports are in different VLANs.

Note There could be times when unknown unicast or multicast traffic from a nonprotected port is flooded to a protected port because a MAC address has timed out or has not been learned by the switch. Use the **switchport block unicast** and **switchport block multicast** interface configuration commands to guarantee that no unicast or multicast traffic is flooded to the port in such a case.

You can configure protected ports on a physical interface or an EtherChannel group. When you enable protected ports for a port channel, it is enabled for all ports in the port-channel group.

Beginning in privileged EXEC mode, follow these steps to define a port as a protected port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to configure, and enter interface configuration mode.
Step 3	**switchport protected**	Configure the interface to be a protected port.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **switchport**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable protected port, use the **no switchport protected** interface configuration command.

This example shows how to configure a port as a protected port:

```
Switch# configure terminal
Switch(config)# interface fastethernet0/1
Switch(config-if)# switchport protected
Switch(config-if)# end
```

Configuring Port Blocking

By default, the switch floods packets with unknown destination MAC addresses to all ports. If unknown unicast and multicast traffic is forwarded to a protected port, there could be security issues.

To prevent unknown unicast or multicast traffic from being forwarded from one port to another, you can configure a port (protected or nonprotected) to block unknown unicast or multicast packets.

Note Blocking unicast or multicast traffic is not automatically enabled on protected ports; you must explicitly configure it.

Blocking Flooded Traffic on an Interface

Note The interface can be a physical interface or an EtherChannel group. When you block multicast or unicast traffic for a port channel, it is blocked on all ports in the port channel group.

Beginning in privileged EXEC mode, follow these steps to disable the flooding of multicast and unicast packets to an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to configure, and enter interface configuration mode.
Step 3	**switchport block multicast**	Block unknown multicast forwarding to the port.

	Command	Purpose
Step 4	**switchport block unicast**	Block unknown unicast forwarding to the port.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the interface to the default condition where no traffic is blocked, use the **no switchport block** {**multicast** | **unicast**} interface configuration commands.

This example shows how to block unicast and multicast flooding on a port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# switchport block multicast
Switch(config-if)# switchport block unicast
Switch(config-if)# end
```

Resuming Normal Forwarding on a Port

Beginning in privileged EXEC mode, follow these steps to resume normal forwarding on a port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to configure, and enter interface configuration mode.
Step 3	**no switchport block multicast**	Enable unknown multicast flooding to the port.
Step 4	**no switchport block unicast**	Enable unknown unicast flooding to the port.
Step 5	**end**	Return to privileged EXEC mode
Step 6	**show interfaces** *interface-id* **switchport**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring Port Security

You can use the port security feature to restrict input to an interface by limiting and identifying MAC addresses of the stations allowed to access the port. When you assign secure MAC addresses to a secure port, the port does not forward packets with source addresses outside the group of defined addresses.

This section includes information about these topics:

- Understanding Port Security, page 21-8
- Default Port Security Configuration, page 21-9
- Port Security Configuration Guidelines, page 21-10
- Enabling and Configuring Port Security, page 21-11
- Enabling and Configuring Port Security Aging, page 21-15

Understanding Port Security

This section includes information about:

- Secure MAC Addresses, page 21-8
- Security Violations, page 21-8

Secure MAC Addresses

You can configure these types of secure MAC addresses:

- Static secure MAC addresses—These are manually configured by using the **switchport port-security mac-address** *mac-address* interface configuration command, stored in the address table, and added to the switch running configuration.
- Dynamic secure MAC addresses—These are dynamically learned, stored only in the address table, and removed when the switch restarts.
- *Sticky* secure MAC addresses—These can be dynamically learned or manually configured, stored in the address table, and added to the running configuration. If these addresses are saved in the configuration file, the interface does not need to dynamically relearn them when the switch restarts. Although sticky secure addresses can be manually configured, we do not recommend it.

You can configure an interface to convert the dynamic MAC addresses to sticky secure MAC addresses and to add them to the running configuration by enabling *sticky learning*. To enable sticky learning, enter the **switchport port-security mac-address sticky** interface configuration command. When you enter this command, the interface converts all the dynamic secure MAC addresses, including those that were dynamically learned before sticky learning was enabled, to sticky secure MAC addresses.

The sticky secure MAC addresses do not automatically become part of the configuration file, which is the startup configuration used each time the switch restarts. If you save the sticky secure MAC addresses in the configuration file, when the switch restarts, the interface does not need to relearn these addresses. If you do not save the configuration, they are lost.

If sticky learning is disabled, the sticky secure MAC addresses are converted to dynamic secure addresses and are removed from the running configuration.

The maximum number of available MAC addresses on a secure port or VLAN is determined by the active Switch Database Management (SDM) template. See the "Optimizing System Resources for User-Selected Features" section on page 6-26 for more information about configuring an SDM template.

Security Violations

It is a security violation when one of these situations occurs:

- The maximum number of secure MAC addresses have been added to the address table, and a station whose MAC address is not in the address table attempts to access the interface.
- An address learned or configured on one secure interface is seen on another secure interface in the same VLAN.

You can configure the interface for one of three violation modes, based on the action to be taken if a violation occurs:

- protect—When the number of secure MAC addresses reaches the limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred.

Note We do not recommend enabling the **protect** mode on a trunk port. The **protect** mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit.

- restrict—When the number of secure MAC addresses reaches the limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses or increase the number of maximum allowable addresses. In this mode, you are notified that a security violation has occurred. Specifically, an SNMP trap is sent, a syslog message is logged, and the violation counter increments.
- shutdown—In this mode, a port security violation causes the interface to immediately become error-disabled, and turns off the port LED. It also sends an SNMP trap, logs a syslog message, and increments the violation counter. When a secure port is in the error-disabled state, you can bring it out of this state by entering the **errdisable recovery cause** *psecure-violation* global configuration command, or you can manually re-enable it by entering the **shutdown** and **no shutdown** interface configuration commands. This is the default mode.

Table 21-1 shows the violation mode and the actions taken when you configure an interface for port security.

Table 21-1 Security Violation Mode Actions

Violation Mode	Traffic is forwarded[1]	Sends SNMP trap	Sends syslog message	Displays error message[2]	Violation counter increments	Shuts down port
protect	No	No	No	No	No	No
restrict	No	Yes	Yes	No	Yes	No
shutdown	No	Yes	Yes	No	Yes	Yes

1. Packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses.
2. The switch will return an error message if you manually configure an address that would cause a security violation.

Default Port Security Configuration

Table 21-2 shows the default port security configuration for an interface.

Table 21-2 Default Port Security Configuration

Feature	Default Setting
Port security	Disabled.
Maximum number of secure MAC addresses	One.
Violation mode	Shutdown.
Sticky address learning	Disabled.
Port security aging	Disabled. Aging time is 0. When enabled, the default **type** is **absolute**.

Port Security Configuration Guidelines

Follow these guidelines when configuring port security:

- Port security can only be configured on static access ports, trunk ports, or IEEE 802.1Q tunnel ports.
- A secure port cannot be a dynamic access port.
- A secure port cannot be a destination port for Switched Port Analyzer (SPAN).
- A secure port cannot belong to a Fast EtherChannel or a Gigabit EtherChannel port group.

Note Voice VLAN is only supported on access ports and not on trunk ports, even though the configuration is allowed.

- When you enable port security on an interface that is also configured with a voice VLAN, set the maximum allowed secure addresses on the port to two. When the port is connected to a Cisco IP phone, the IP phone requires one MAC address. The Cisco IP phone address is learned on the voice VLAN, but is not learned on the access VLAN. If you connect a single PC to the Cisco IP phone, no additional MAC addresses are required. If you connect more than one PC to the Cisco IP phone, you must configure enough secure addresses to allow one for each PC and one for the phone.
- When you enter a maximum secure address value for an interface, and the new value is greater than the previous value, the new value overwrites the previously configured value. If the new value is less than the previous value and the number of configured secure addresses on the interface exceeds the new value, the command is rejected.
- The switch does not support port security aging of sticky secure MAC addresses.

Table 21-3 summarizes port security compatibility with other features configured on a port.

Table 21-3 Port Security Compatibility with Other Catalyst 3550 Features

Type of Port	Compatible with Port Security
DTP[1] port[2]	No
Trunk port	Yes
Dynamic-access port[3]	No
Routed port	No
SPAN source port	Yes
SPAN destination port	No
EtherChannel	No
Tunneling port	Yes
Protected port	Yes
IEEE 802.1x port	Yes
Voice VLAN port[4]	Yes
IP source guard	Yes
Dynamic Address Resolution Protocol (ARP) inspection	Yes
Flex Links	Yes

1. DTP = Dynamic Trunking Protocol
2. A port configured with the **switchport mode dynamic** interface configuration command.
3. A VLAN Query Protocol (VQP) port configured with the **switchport access vlan dynamic** interface configuration command.
4. You must set the maximum allowed secure addresses on the port to two plus the maximum number of secure addresses allowed on the access VLAN.

Enabling and Configuring Port Security

Beginning in privileged EXEC mode, follow these steps to restrict input to an interface by limiting and identifying MAC addresses of the stations allowed to access the port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**switchport mode** {**access** | **trunk**}	Set the interface switchport mode as access or trunk. An interface in the default mode (dynamic auto) cannot be configured as a secure port.
Step 4	**switchport voice vlan** *vlan-id*	Enable voice VLAN on a port. *vlan-id*—Specify the VLAN to be used for voice traffic.
Step 5	**switchport port-security**	Enable port security on the interface.

	Command	Purpose
Step 6	**switchport port-security** [**maximum** *value* [**vlan** {*vlan-list* | {**access** | **voice**}}]]	(Optional) Set the maximum number of secure MAC addresses for the interface. The maximum number of available addresses is determined by the active Switch Database Management (SDM) template. The default is 1. If an interface is configured for voice VLAN, configure a maximum of two secure MAC addresses. (Optional) **vlan**—Set a per-VLAN maximum value. Enter one of these options after you enter the **vlan** keyword: • *vlan-list*—On a trunk port, you can set a per-VLAN maximum value on a range of VLANs separated by a hyphen or a series of VLANs separated by commas. For nonspecified VLANs, the per-VLAN maximum value is used. • **access**—On an access port, specify the VLAN as an access VLAN. • **voice**—On an access port, specify the VLAN as a voice VLAN. **Note** The **voice** keyword is available only if voice VLAN is configured on a port and if that port is not the access VLAN.
Step 7	**switchport port-security violation** {**protect** | **restrict** | **shutdown**}	(Optional) Set the violation mode, the action to be taken when a security violation is detected, as one of these: • **protect**—When the number of port secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred. **Note** We do not recommend configuring the protect mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit. • **restrict**—When the number of secure MAC addresses reaches the limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses or increase the number of maximum allowable addresses. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. • **shutdown**—The interface is error-disabled when a violation occurs, and the port LED turns off. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. **Note** When a secure port is in the error-disabled state, you can bring it out of this state by entering the **errdisable recovery cause psecure-violation** global configuration command, or you can manually re-enable it by entering the **shutdown** and **no shutdown** interface configuration commands.

	Command	Purpose
Step 8	**switchport port-security** [**mac-address** *mac-address* [**vlan** {*vlan-id* | {**access** | **voice**}}]	(Optional) Enter a secure MAC address for the interface. You can use this command to enter the maximum number of secure MAC addresses. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned. **Note** If you enable sticky learning after you enter this command, the secure addresses that were dynamically learned are converted to sticky secure MAC addresses and are added to the running configuration. (Optional) **vlan**—set a per-VLAN maximum value. Enter one of these options after you enter the **vlan** keyword: • *vlan-id*—On a trunk port, you can specify the VLAN ID and the MAC address. If you do not specify a VLAN ID, the native VLAN is used. • **access**—On an access port, specify the VLAN as an access VLAN. • **voice**—On an access port, specify the VLAN as a voice VLAN. **Note** The **voice** keyword is available only if voice VLAN is configured on a port and if that port is not the access VLAN.
Step 9	**switchport port-security mac-address sticky**	(Optional) Enable sticky learning on the interface.
Step 10	**switchport port-security mac-address sticky** [*mac-address* | **vlan** {*vlan-id* | {**access** | **voice**}}]	(Optional) Enter a sticky secure MAC address, repeating the command as many times as necessary. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned, are converted to sticky secure MAC addresses, and are added to the running configuration. **Note** If you do not enable sticky learning before this command is entered, an error message appears, and you cannot enter a sticky secure MAC address. (Optional) **vlan**—set a per-VLAN maximum value. Enter one of these options after you enter the **vlan** keyword: • *vlan-id*—On a trunk port, you can specify the VLAN ID and the MAC address. If you do not specify a VLAN ID, the native VLAN is used. • **access**—On an access port, specify the VLAN as an access VLAN. • **voice**—On an access port, specify the VLAN as a voice VLAN. **Note** The **voice** keyword is available only if voice VLAN is configured on a port and if that port is not the access VLAN.
Step 11	**end**	Return to privileged EXEC mode.
Step 12	**show port-security**	Verify your entries.
Step 13	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the interface to the default condition as not a secure port, use the **no switchport port-security** interface configuration command. If you enter this command when sticky learning is enabled, the sticky secure addresses remain part of the running configuration but are removed from the address table. All addresses are now dynamically learned.

To return the interface to the default number of secure MAC addresses, use the **no switchport port-security maximum** *value* interface configuration command.

To return the violation mode to the default condition (shutdown mode), use the **no switchport port-security violation** {**protect** | **restrict**} interface configuration command.

To disable sticky learning on an interface, use the **no switchport port-security mac-address sticky** interface configuration command. The interface converts the sticky secure MAC addresses to dynamic secure addresses.

To delete a static secure MAC address from the address table, use the **clear port-security configured address** *mac-address* privileged EXEC command. To delete all the static secure MAC addresses on an interface or a VLAN, use the **clear port-security configured interface** *interface-id* privileged EXEC command.

To delete a dynamic secure MAC address from the address table, use the **clear port-security dynamic address** *mac-address* privileged EXEC command. To delete all the dynamic addresses on an interface or a VLAN, use the **clear port-security dynamic interface** *interface-id* privileged EXEC command.

To delete a sticky secure MAC addresses from the address table, use the **clear port-security sticky address** *mac-address* privileged EXEC command. To delete all the sticky addresses on an interface or a VLAN, use the **clear port-security sticky interface** *interface-id* privileged EXEC command.

This example shows how to enable port security on a port and to set the maximum number of secure addresses to 50. The violation mode is the default, no static secure MAC addresses are configured, and sticky learning is enabled.

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface fastethernet0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security maximum 50
Switch(config-if)# switchport port-security mac-address sticky
Switch(config-if)# end
```

This example shows how to configure a static secure MAC address on a port and enable sticky learning:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface fastethernet0/2
Switch(config-if)# switchport mode access
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security mac-address 0000.02000.0004
Switch(config-if)# switchport port-security mac-address sticky
Switch(config-if)# end
```

This example shows how to configure a maximum of eight secure MAC addresses on VLAN 5 on a port:

```
Switch(config-if)# switchport port-security maximum 8 vlan 5
Switch(config-if)# end
```

This example shows how to configure a maximum of eight secure MAC addresses on VLAN 5 on a port:

```
Switch(config-if)# switchport port-security maximum 8 vlan 5
Switch(config-if)# end
```

This example shows how to enable sticky port security on a port, to manually configure MAC addresses for data VLAN and voice VLAN, and to set the total maximum number of secure addresses to 20 (10 for data VLAN and 10 for voice VLAN).

```
Switch(config)# interface FastEthernet1/0/1
Switch(config-if)# switchport access vlan 21
Switch(config-if)# switchport mode access
Switch(config-if)# switchport voice vlan 22
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security maximum 20
Switch(config-if)# switchport port-security violation restrict
Switch(config-if)# switchport port-security mac-address sticky
Switch(config-if)# switchport port-security mac-address sticky 0000.0000.0002
Switch(config-if)# switchport port-security mac-address 0000.0000.0003
Switch(config-if)# switchport port-security mac-address sticky 0000.0000.0001 vlan voice
Switch(config-if)# switchport port-security mac-address 0000.0000.0004 vlan voice
Switch(config-if)# switchport port-security maximum 10 vlan access
Switch(config-if)# switchport port-security maximum 10 vlan voice
```

This example shows how to configure a maximum of eight secure MAC addresses on VLAN 5 on a port:

```
Switch(config-if)# switchport port-security maximum 8 vlan 5
Switch(config-if)# end
```

Enabling and Configuring Port Security Aging

You can use port security aging to set the aging time for static and dynamic secure addresses on a port. Two types of aging are supported per port:

- Absolute—The secure addresses on the port are deleted after the specified aging time.
- Inactivity—The secure addresses on the port are deleted only if the secure addresses are inactive for the specified aging time.

Use this feature to remove and add PCs on a secure port without manually deleting the existing secure MAC addresses and to still limit the number of secure addresses on a port. You can enable or disable the aging of statically configured secure addresses on a per-port basis.

Beginning in privileged EXEC mode, follow these steps to configure port security aging:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the port on which you want to enable port security aging, and enter interface configuration mode. **Note** The switch does not support port security aging of sticky secure addresses.
Step 3	**switchport port-security aging** {**static** | **time** *time* | **type** {**absolute** | **inactivity**}}	Enable or disable static aging for the secure port, or set the aging time or type. Enter **static** to enable aging for statically configured secure addresses on this port. For *time*, specify the aging time for this port. The valid range is from 0 to 1440 minutes. If the time is equal to 0, aging is disabled for this port. For **type**, select one of these keywords: • **absolute**—Sets the aging type as absolute aging. All the secure addresses on this port age out after the specified time (minutes) lapses and are removed from the secure address list. **Note** The absolute aging time could vary by 1 minute, depending on the sequence of the system timer. • **inactivity**—Sets the aging type as inactivity aging. The secure addresses on this port age out only if there is no data traffic from the secure source addresses for the specified time period.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show port-security** [**interface** *interface-id*] [**address**]	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable port security aging for all secure addresses on a port, use the **no switchport port-security aging time** interface configuration command. To disable aging for only statically configured secure addresses, use the **no switchport port-security aging static** interface configuration command.

This example shows how to set the aging time as 2 hours for the secure addresses on a port:

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# switchport port-security aging time 120
```

This example shows how to set the aging time as 2 minutes for the inactivity aging type with aging enabled for the configured secure addresses on the interface:

```
Switch(config-if)# switchport port-security aging time 2
Switch(config-if)# switchport port-security aging type inactivity
Switch(config-if)# switchport port-security aging static
```

You can verify the previous commands by entering the **show port-security interface** *interface-id* privileged EXEC command.

Displaying Port-Based Traffic Control Settings

The **show interfaces** *interface-id* **switchport** privileged EXEC command displays (among other characteristics) the interface traffic suppression and control configuration. The **show storm-control** and **show port-security** privileged EXEC commands display those features.

To display traffic control information, use one or more of the privileged EXEC commands in Table 21-4.

Table 21-4 Commands for Displaying Traffic Control Status and Configuration

Command	Purpose
show interfaces [*interface-id*] **switchport**	Displays the administrative and operational status of all switching (nonrouting) ports or the specified port, including port blocking and port protection settings.
show storm-control [*interface-id*] [**broadcast** | **multicast** | **unicast**]	Displays storm control suppression levels set on all interfaces or the specified interface for the specified traffic type or for broadcast traffic if no traffic type is entered.
show port-security [**interface** *interface-id*]	Displays port security settings for the switch or for the specified interface, including the maximum allowed number of secure MAC addresses for each interface, the number of secure MAC addresses on the interface, the number of security violations that have occurred, and the violation mode.
show port-security [**interface** *interface-id*] **address**	Displays all secure MAC addresses configured on all switch interfaces or on a specified interface with aging information for each address.
show port-security [**interface** *interface-id*] **vlan**	Displays the maximum allowed number of secure MAC addresses for each VLAN and the number of secure MAC addresses on the VLAN.

CHAPTER 22

Configuring CDP

This chapter describes how to configure Cisco Discovery Protocol (CDP) on your Catalyst 3550 switch.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release and the "System Management Commands" section in the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2.*

This chapter consists of these sections:

- Understanding CDP, page 22-1
- Configuring CDP, page 22-2
- Monitoring and Maintaining CDP, page 22-4

Understanding CDP

CDP is a device discovery protocol that runs over Layer 2 (the data link layer) on all Cisco-manufactured devices (routers, bridges, access servers, and switches) and allows network management applications to discover Cisco devices that are neighbors of already known devices. With CDP, network management applications can learn the device type and the Simple Network Management Protocol (SNMP) agent address of neighboring devices running lower-layer, transparent protocols. This feature enables applications to send SNMP queries to neighboring devices.

CDP runs on all media that support Subnetwork Access Protocol (SNAP). Because CDP runs over the data-link layer only, two systems that support different network-layer protocols can learn about each other.

Each CDP-configured device sends periodic messages to a multicast address, advertising at least one address at which it can receive SNMP messages. The advertisements also contain time-to-live, or holdtime information, which is the length of time a receiving device holds CDP information before discarding it. Each device also listens to the messages sent by other devices to learn about neighboring devices.

On the switch, CDP enables the Network Assistant software to display a graphical view of the network. The switch uses CDP to find cluster candidates and maintain information about cluster members and other devices up to three cluster-enabled devices away from the command switch by default.

The switch supports CDP Version 2.

Configuring CDP

These sections include CDP configuration information and procedures:

- Default CDP Configuration, page 22-2
- Configuring the CDP Characteristics, page 22-2
- Disabling and Enabling CDP, page 22-3
- Disabling and Enabling CDP on an Interface, page 22-4

Default CDP Configuration

Table 22-1 shows the default CDP configuration.

Table 22-1 Default CDP Configuration

Feature	Default Setting
CDP global state	Enabled
CDP interface state	Enabled
CDP timer (packet update frequency)	60 seconds
CDP holdtime (before discarding)	180 seconds
CDP Version-2 advertisements	Enabled

Configuring the CDP Characteristics

You can configure the frequency of CDP updates, the amount of time to hold the information before discarding it, and whether or not to send Version-2 advertisements.

Beginning in privileged EXEC mode, follow these steps to configure the CDP timer, holdtime, and advertisement type.

Note Steps 2 through 4 are all optional and can be performed in any order.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**cdp timer** *seconds*	(Optional) Set the transmission frequency of CDP updates in seconds. The range is 5 to 254; the default is 60 seconds.
Step 3	**cdp holdtime** *seconds*	(Optional) Specify the amount of time a receiving device should hold the information sent by your device before discarding it. The range is 10 to 255 seconds; the default is 180 seconds.
Step 4	**cdp advertise-v2**	(Optional) Configure CDP to send Version-2 advertisements. This is the default state.
Step 5	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 6	**show cdp**	Verify your settings.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of the CDP commands to return to the default settings.

This example shows how to configure CDP characteristics.

```
Switch# configure terminal
Switch(config)# cdp timer 50
Switch(config)# cdp holdtime 120
Switch(config)# cdp advertise-v2
Switch(config)# end
```

For additional CDP **show** commands, see the "Monitoring and Maintaining CDP" section on page 22-4.

Disabling and Enabling CDP

CDP is enabled by default.

Note Creating and maintaining switch clusters is based on the regular exchange of CDP messages. Disabling CDP can interrupt cluster discovery. For more information, see Clustering Switches and see the *Getting Started with Cisco Network Assistant*, available on Cisco.com.For more information about switch clusters, see Chapter 5, "Clustering Switches."

Beginning in privileged EXEC mode, follow these steps to disable the CDP device discovery capability:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no cdp run**	Disable CDP.
Step 3	**end**	Return to privileged EXEC mode.

Beginning in privileged EXEC mode, follow these steps to enable CDP when it has been disabled:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**cdp run**	Enable CDP after disabling it.
Step 3	**end**	Return to privileged EXEC mode.

This example shows how to enable CDP if it has been disabled.

```
Switch# configure terminal
Switch(config)# cdp run
Switch(config)# end
```

Disabling and Enabling CDP on an Interface

CDP is enabled by default on all supported interfaces to send and receive CDP information.

Beginning in privileged EXEC mode, follow these steps to disable CDP on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface on which you are disabling CDP, and enter interface configuration mode.
Step 3	**no cdp enable**	Disable CDP on the interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Beginning in privileged EXEC mode, follow these steps to enable CDP on an interface when it has been disabled:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface on which you are enabling CDP, and enter interface configuration mode.
Step 3	**cdp enable**	Enable CDP on the interface after disabling it.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to enable CDP on an interface when it has been disabled.

```
Switch# configure terminal
Switch(config)# interface fastethernet0/1
Switch(config-if)# cdp enable
Switch(config-if)# end
```

Monitoring and Maintaining CDP

To monitor and maintain CDP on your device, perform one or more of these tasks, beginning in privileged EXEC mode.

Command	Description
clear cdp counters	Reset the traffic counters to zero.
clear cdp table	Delete the CDP table of information about neighbors.
show cdp	Display global information, such as frequency of transmissions and the holdtime for packets being sent.

Command	Description
show cdp entry *entry-name* [**protocol** | **version**]	Display information about a specific neighbor. You can enter an asterisk (*) to display all CDP neighbors, or you can enter the name of the neighbor about which you want information. You can also limit the display to information about the protocols enabled on the specified neighbor or information about the version of software running on the device.
show cdp interface [*interface-id*]	Display information about interfaces where CDP is enabled. You can limit the display to the interface about which you want information.
show cdp neighbors [*interface-id*] [**detail**]	Display information about neighbors, including device type, interface type and number, holdtime settings, capabilities, platform, and port ID. You can limit the display to neighbors on a specific type or number of interface or expand the display to provide more detailed information.
show cdp traffic	Display CDP counters, including the number of packets sent and received and checksum errors.

CHAPTER 23

Configuring UDLD

This chapter describes how to configure the UniDirectional Link Detection (UDLD) protocol on your Catalyst 3550 switch.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding UDLD, page 23-1
- Configuring UDLD, page 23-3
- Displaying UDLD Status, page 23-7

Understanding UDLD

UDLD is a Layer 2 protocol that enables devices connected through fiber-optic or twisted-pair Ethernet cables to monitor the physical configuration of the cables and detect when a unidirectional link exists. All connected devices must support UDLD for the protocol to successfully identify and disable unidirectional links. When UDLD detects a unidirectional link, it administratively shuts down the affected port and alerts you. Unidirectional links can cause a variety of problems, including spanning-tree topology loops.

Modes of Operation

UDLD supports two modes of operation: normal (the default) and aggressive. In normal mode, UDLD can detect unidirectional links due to misconnected interfaces on fiber-optic connections. In aggressive mode, UDLD can also detect unidirectional links due to one-way traffic on fiber-optic and twisted-pair links and to misconnected interfaces on fiber-optic links.

In normal and aggressive modes, UDLD works with the Layer 1 mechanisms to determine the physical status of a link. At Layer 1, autonegotiation takes care of physical signaling and fault detection. UDLD performs tasks that autonegotiation cannot perform, such as detecting the identities of neighbors and shutting down misconnected interfaces. When you enable both autonegotiation and UDLD, the Layer 1 and Layer 2 detections work together to prevent physical and logical unidirectional connections and the malfunctioning of other protocols.

A unidirectional link occurs whenever traffic sent by a local device is received by its neighbor but traffic from the neighbor is not received by the local device.

In normal mode, UDLD detects a unidirectional link when fiber strands in a fiber-optic interface are misconnected and the Layer 1 mechanisms do not detect this misconnection. If the interfaces are connected correctly but the traffic is one way, UDLD does not detect the unidirectional link because the Layer 1 mechanism, which is supposed to detect this condition, does not do so. In case, the logical link is considered undetermined, and UDLD does not disable the interface.

When UDLD is in normal mode, if one of the fiber strands in a pair is disconnected and autonegotiation is active, the link does not stay up because the Layer 1 mechanisms did not detect a physical problem with the link. In this case, UDLD does not take any action, and the logical link is considered undetermined.

In aggressive mode, UDLD detects a unidirectional link by using the previous detection methods. UDLD in aggressive mode can also detect a unidirectional link on a point-to-point link on which no failure between the two devices is allowed. It can also detect a unidirectional link when one of these problems exists:

- On fiber-optic or twisted-pair links, one of the interfaces cannot send or receive traffic.
- On fiber-optic or twisted-pair links, one of the interfaces is down while the other is up.
- One of the fiber strands in the cable is disconnected.

In these cases, UDLD shuts down the affected interface.

In a point-to-point link, UDLD hello packets can be considered as a heart beat whose presence guarantees the health of the link. Conversely, the loss of the heart beat means that the link must be shut down if it is not possible to re-establish a bidirectional link.

If both fiber strands in a cable are working normally from a Layer 1 perspective, UDLD in aggressive mode determines whether those fiber strands are connected correctly and whether traffic is flowing bidirectionally between the correct neighbors. This check cannot be performed by autonegotiation because autonegotiation operates at Layer 1.

Methods to Detect Unidirectional Links

UDLD operates by using two mechanisms:

- Neighbor database maintenance

  UDLD learns about other UDLD-capable neighbors by periodically sending a hello packet (also called an advertisement or probe) on every active interface to keep each device informed about its neighbors.

  When the switch receives a hello message, it caches the information until the age time (hold time or time-to-live) expires. If the switch receives a new hello message before an older cache entry ages, the switch replaces the older entry with the new one.

  Whenever an interface is disabled and UDLD is running, whenever UDLD is disabled on an interface, or whenever the switch is reset, UDLD clears all existing cache entries for the interfaces affected by the configuration change. UDLD sends at least one message to inform the neighbors to flush the part of their caches affected by the status change. The message is intended to keep the caches synchronized.

- Event-driven detection and echoing

  UDLD relies on echoing as its detection mechanism. Whenever a UDLD device learns about a new neighbor or receives a resynchronization request from an out-of-sync neighbor, it restarts the detection window on its side of the connection and sends echo messages in reply. Because this behavior is the same on all UDLD neighbors, the sender of the echoes expects to receive an echo in reply.

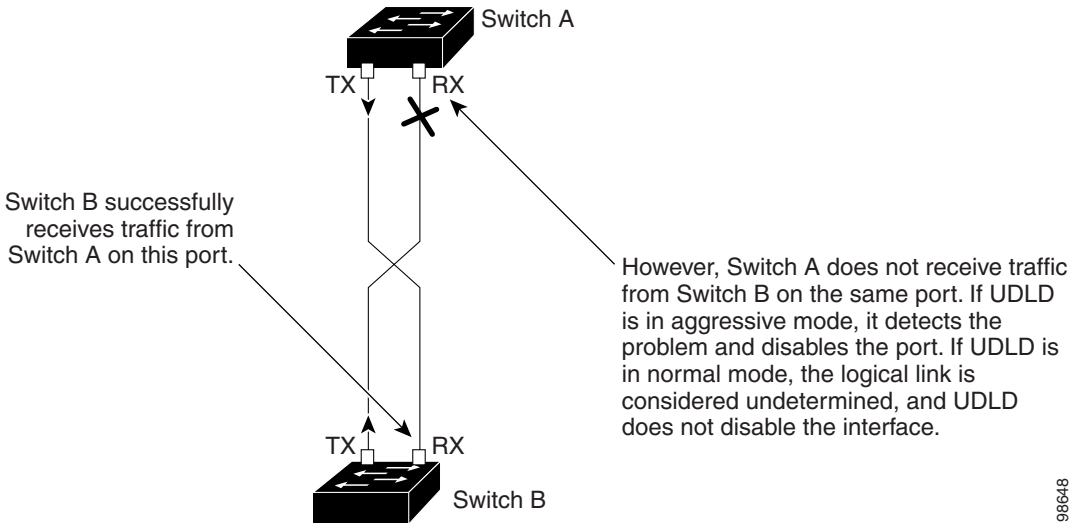
  If the detection window ends and no valid reply message is received, the link might shut down, depending on the UDLD mode. When UDLD is in normal mode, the link might be considered undetermined and might not be shut down. When UDLD is in aggressive mode, the link is considered unidirectional, and the interface is shut down.

If UDLD in normal mode is in the advertisement or in the detection phase and all the neighbor cache entries are aged out, UDLD restarts the link-up sequence to resynchronize with any potentially out-of-sync neighbors.

If you enable aggressive mode when all the neighbors of a port have aged out either in the advertisement or in the detection phase, UDLD restarts the link-up sequence to resynchronize with any potentially out-of-sync neighbor. UDLD shuts down the port if, after the fast train of messages, the link state is still undetermined.

Figure 23-1 shows an example of a unidirectional link condition.

Figure 23-1 UDLD Detection of a Unidirectional Link



Configuring UDLD

This section describes how to configure UDLD on your switch. It contains this configuration information:

- Default UDLD Configuration, page 23-4
- Configuration Guidelines, page 23-4
- Enabling UDLD Globally, page 23-4

- Enabling UDLD on an Interface, page 23-5
- Resetting an Interface Shut Down by UDLD, page 23-6

Default UDLD Configuration

Table 23-1 shows the default UDLD configuration.

Table 23-1 Default UDLD Configuration

Feature	Default Setting
UDLD global enable state	Globally disabled
UDLD per-interface enable state for fiber-optic media	Disabled on all Ethernet fiber-optic interfaces
UDLD per-interface enable state for twisted-pair (copper) media	Disabled on all Ethernet 10/100 and 1000BASE-TX interfaces
UDLD aggressive mode	Disabled

Configuration Guidelines

These are the UDLD configuration guidelines:

- A UDLD-capable interface also cannot detect a unidirectional link if it is connected to a UDLD-incapable port of another switch.
- When configuring the mode (normal or aggressive), make sure that the same mode is configured on both sides of the link.

Caution Loop guard works only on point-to-point links. We recommend that each end of the link has a directly connected device that is running STP.

Enabling UDLD Globally

Beginning in privileged EXEC mode, follow these steps to enable UDLD in the aggressive or normal mode and to set the configurable message timer on all fiber-optic interfaces on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**udld** {**aggressive** | **enable** | **message time** *message-timer-interval*}	Specify the UDLD mode of operation: • **aggressive**—Enables UDLD in aggressive mode on all fiber-optic interfaces. • **enable**—Enables UDLD in normal mode on all fiber-optic interfaces on the switch. UDLD is disabled by default. An individual interface configuration overrides the setting of the **udld enable** global configuration command. For more information about aggressive and normal modes, see the "Modes of Operation" section on page 23-1. • **message time** *message-timer-interval*—Configures the period of time between UDLD probe messages on ports that are in the advertisement phase and are determined to be bidirectional. The range is from 1 to 90 seconds. **Note** This command affects fiber-optic interfaces only. Use the **udld** interface configuration command to enable UDLD on other interface types. For more information, see the "Enabling UDLD on an Interface" section on page 23-5.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show udld**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable UDLD globally, use the **no udld enable** global configuration command to disable normal mode UDLD on all fiber-optic ports. Use the **no udld aggressive** global configuration command to disable aggressive mode UDLD on all fiber-optic ports.

Enabling UDLD on an Interface

Beginning in privileged EXEC mode, follow these steps to enable UDLD in the aggressive or normal mode on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be enabled for UDLD, and enter interface configuration mode.

	Command	Purpose
Step 3	**udld port** [**aggressive**]	Specify the UDLD mode of operation: • (Optional) **aggressive**— Enables UDLD in aggressive mode on the specified interface. UDLD is disabled by default. If you do not enter the **aggressive** keyword, the switch enables UDLD in normal mode. On a fiber-optic interface, this command overrides the **udld enable** global configuration command setting. For more information about aggressive and normal modes, see the "Modes of Operation" section on page 23-1.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show udld** *interface-id*	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable UDLD on a non-fiber-optic interface, use the **no udld port** interface configuration command.

Note On fiber-optic interfaces, the **no udld port** command reverts the interface configuration to the **udld enable** global configuration command setting.

Use the **no udld port** interface configuration command to disable UDLD on a fiber-optic interface

Resetting an Interface Shut Down by UDLD

Beginning in privileged EXEC mode, follow these steps to reset all interfaces shut down by UDLD:

	Command	Purpose
Step 1	**udld reset**	Reset all interfaces shut down by UDLD.
Step 2	**show udld**	Verify your entries.
Step 3	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

You can also bring up the interface by using these commands:

- **The shutdown** interface configuration command followed by the **no shutdown** interface configuration command restarts the disabled interface.
- The **no udld** {**aggressive** | **enable**} global configuration command followed by the **udld** {**aggressive** | **enable**} global configuration command re-enables UDLD globally.
- The **no udld port** interface configuration command followed by the **udld port** [**aggressive**] interface configuration command re-enables UDLD on the specified interface.
- The **errdisable recovery cause udld** global configuration command enables the timer to automatically recover from the UDLD error-disabled state, and the **errdisable recovery interval** *interval* global configuration command specifies the time to recover from the UDLD error-disabled state.

Displaying UDLD Status

To display the UDLD status for the specified interface or for all interfaces, use the **show udld** [*interface-id*] privileged EXEC command.

For detailed information about the fields in the display, see the command reference for this release.

CHAPTER 24

Configuring SPAN and RSPAN

This chapter describes how to configure Switched Port Analyzer (SPAN) and Remote SPAN (RSPAN) on your Catalyst 3550 switch.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

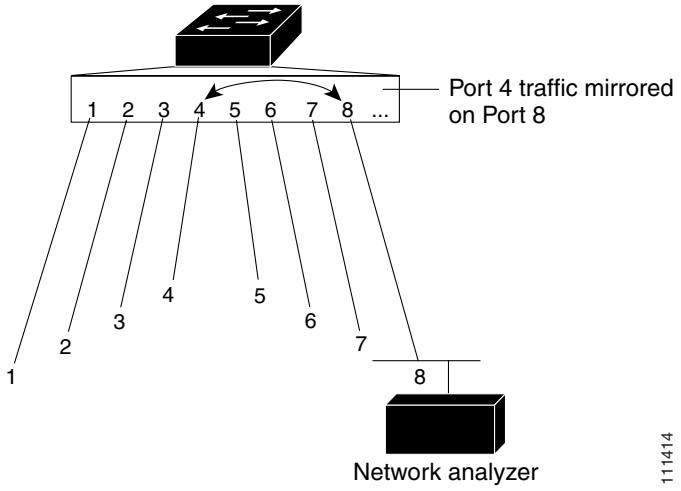
- Understanding SPAN and RSPAN, page 24-1
- Configuring SPAN, page 24-8
- Configuring RSPAN, page 24-16
- Displaying SPAN and RSPAN Status, page 24-24

Understanding SPAN and RSPAN

You can analyze network traffic passing through ports or VLANs by using SPAN to send a copy of the traffic to another port on the switch that has been connected to a SwitchProbe device or other Remote Monitoring (RMON) probe or security device. SPAN mirrors received or sent (or both) traffic on a source port and received traffic on one or more source ports or source VLANs, to a destination port for analysis.

For example, in Figure 24-1, all traffic on port 4 (the source port) is mirrored to port 8 (the destination port). A network analyzer on port 8 receives all network traffic from port 4 without being physically attached to port 4.

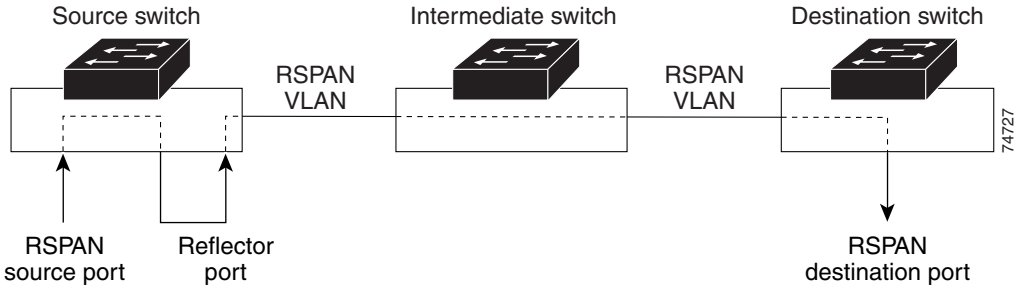
Figure 24-1 Example SPAN Configuration



Only traffic that enters or leaves source ports or traffic that enters source VLANs can be monitored by using SPAN; traffic that gets routed to ingress source ports or source VLANs cannot be monitored. For example, if incoming traffic is being monitored, traffic that gets routed from another VLAN to the source VLAN is not monitored; however, traffic that is received on the source VLAN and routed to another VLAN is monitored.

RSPAN extends SPAN by enabling remote monitoring of multiple switches across your network. The traffic for each RSPAN session is carried over a user-specified RSPAN VLAN that is dedicated for that RSPAN session in all participating switches. The SPAN traffic from the sources is copied onto the RSPAN VLAN through a reflector port and then forwarded over trunk ports that are carrying the RSPAN VLAN to any RSPAN destination sessions monitoring the RSPAN VLAN, as shown in Figure 24-2.

Figure 24-2 Example of RSPAN Configuration



SPAN and RSPAN do not affect the switching of network traffic on source ports or source VLANs; a copy of the packets received or sent by the source interfaces are sent to the destination interface.

You can use the SPAN or RSPAN destination port to inject traffic from a network security device. For example, if you connect a Cisco Intrusion Detection System (IDS) Sensor Appliance to a destination port, the IDS device can send TCP Reset packets to close down the TCP session of a suspected attacker.

SPAN and RSPAN Concepts and Terminology

This section describes concepts and terminology associated with SPAN and RSPAN configuration.

SPAN Session

A local SPAN session is an association of a destination port with source ports and source VLANs. An RSPAN session is an association of source ports and source VLANs across your network with an RSPAN VLAN. The destination source is the RSPAN VLAN.

You configure SPAN sessions by using parameters that specify the source of network traffic to monitor. Traffic monitoring in a SPAN session has these restrictions:

- You can monitor incoming traffic on a series or range of ports and VLANs.
- You can monitor outgoing traffic on a single port; you cannot monitor outgoing traffic on multiple ports.
- You cannot monitor outgoing traffic on VLANs.

You can configure two separate SPAN or RSPAN sessions with separate or overlapping sets of SPAN source ports and VLANs. Both switched and routed ports can be configured as SPAN sources and destinations.

SPAN sessions do not interfere with the normal operation of the switch. However, an oversubscribed SPAN destination, for example, a 10-Mbps port monitoring a 100-Mbps port, results in dropped or lost packets.

You can configure SPAN sessions on disabled ports; however, a SPAN session does not become active unless you enable the destination port and at least one source port or VLAN for that session. The **show monitor session** *session_number* privileged EXEC command displays the operational status of a SPAN session.

A SPAN session remains inactive after system power-on until the destination port is operational.

Traffic Types

SPAN sessions include these traffic types:

- Receive (Rx) SPAN—The goal of receive (or ingress) SPAN is to monitor as much as possible all the packets received by the source interface or VLAN before any modification or processing is performed by the switch. A copy of each packet received by the source is sent to the destination port for that SPAN session. You can monitor a series or range of ingress ports or VLANs in a SPAN session.

  On tagged packets (Inter-Switch Link [ISL] or IEEE 802.1Q), the tagging is removed at the ingress port. At the destination port, if tagging is enabled, the packets appear with the ISL or IEEE 802.1Q headers. If no tagging is specified, packets appear in the native format.

  Packets that are modified because of routing are copied without modification for Rx SPAN; that is, the original packet is copied. Packets that are modified because of quality of service (QoS)—for example, modified Differentiated Services Code Point (DSCP)—are copied with modification for Rx SPAN.

  Some features that can cause a packet to be dropped during receive processing have no effect on SPAN; the destination port receives a copy of the packet even if the actual incoming packet is dropped. These features include IP standard and extended input access control lists (ACLs), IP standard and extended output ACLs for unicast and ingress QoS policing.VLAN maps, ingress QoS policing, and policy-based routing. Switch congestion that causes packets to be dropped also has no effect on SPAN.

- Transmit (Tx) SPAN—The goal of transmit (or egress) SPAN is to monitor as much as possible all the packets sent by the source interface after all modification and processing is performed by the switch. A copy of each packet sent by the source is sent to the destination port for that SPAN session. The copy is provided after the packet is modified.

  Only one egress source port is allowed per SPAN session. VLAN monitoring is not supported in the egress direction.

  Packets that are modified because of routing—for example, with a time-to-live (TTL) or MAC-address modification—are duplicated at the destination port. On packets that are modified because of QoS, the modified packet might not have the same DSCP (IP packet) or CoS (non-IP packet) as the SPAN source.

  Some features that can cause a packet to be dropped during transmit processing might also affect the duplicated copy for SPAN.These features include VLAN maps, IP standard and extended output ACLs on multicast packets, and egress QoS policing. In the case of output ACLs, if the SPAN source drops the packet, the SPAN destination would also drop the packet. In the case of egress QoS policing, if the SPAN source drops the packet, the SPAN destination might not drop it. If the source port is oversubscribed, the destination ports will have different dropping behavior.
- Both—In a SPAN session, you can monitor a single port for both received and sent packets.

Source Port

A source port (also called a *monitored port*) is a switched or routed port that you monitor for network traffic analysis. In a single local SPAN session or RSPAN source session, you can monitor source port traffic such as received (Rx), transmitted (Tx), or bidirectional (both); however, on a VLAN, you can monitor only received traffic. The switch supports any number of source ports (up to the maximum number of available ports on the switch) and any number of source ingress VLANs (up to the maximum number of VLANs supported).

A source port has these characteristics:

- It can be any port type (for example, EtherChannel, Fast Ethernet, Gigabit Ethernet, and so forth).
- It can be monitored in multiple SPAN sessions.
- It cannot be a destination port.
- Each source port can be configured with a direction (ingress, egress, or both) to monitor. For EtherChannel sources, the monitored direction would apply to all the physical ports in the group.
- Source ports can be in the same or different VLANs.
- For VLAN SPAN sources, all active ports in the source VLAN are included as source ports.

You can configure a trunk port as a source port. By default, all VLANs active on the trunk are monitored. You can limit SPAN traffic monitoring on trunk source ports to specific VLANs by using VLAN filtering. Only switched traffic in the selected VLANs is sent to the destination port. This feature affects only traffic forwarded to the destination SPAN port and does not affect the switching of normal traffic. This feature is not allowed in sessions with VLAN sources.

Destination Port

Each local SPAN session or RSPAN destination session must have a destination port (also called a *monitoring port*) that receives a copy of traffic from the source ports and VLANs.

The destination port has these characteristics:

- It must reside on the same switch as the source port (for a local SPAN session).
- It can be any Ethernet physical port.
- It can participate in only one SPAN session at a time (a destination port in one SPAN session cannot be a destination port for a second SPAN session).
- It cannot be a source port or a reflector port.
- It cannot be an EtherChannel group or a VLAN.
- It can be a physical port that is assigned to an EtherChannel group, even if the EtherChannel group has been specified as a SPAN source. The port is removed from the group while it is configured as a SPAN destination port.
- The port does not transmit any traffic except that required for the SPAN session.
- If ingress traffic forwarding is enabled for a network security device, the destination port forwards traffic at Layer 2.
- It does not participate in spanning tree while the SPAN session is active.
- When it is a destination port, it does not participate in any of the Layer 2 protocols— Cisco Discovery Protocol (CDP), VLAN Trunk Protocol (VTP), Dynamic Trunking Protocol (DTP), Spanning Tree Protocol (STP), Port Aggregation Protocol (PagP), and Link Aggregation Control Protocol (LACP).
- A destination port that belongs to a source VLAN of any SPAN session is excluded from the source list and is not monitored.
- No address learning occurs on the destination port.

Reflector Port

The reflector port is the mechanism that copies packets onto an RSPAN VLAN. The reflector port forwards only the traffic from the RSPAN source session with which it is affiliated. Any device connected to a port set as a reflector port loses connectivity until the RSPAN source session is disabled.

The reflector port has these characteristics:

- It is a port set to loopback.
- It cannot be an EtherChannel group, it does not trunk, and it cannot do protocol filtering.
- It can be a physical port that is assigned to an EtherChannel group, even if the EtherChannel group is specified as a SPAN source. The port is removed from the group while it is configured as a reflector port.
- A port used as a reflector port cannot be a SPAN source or destination port, nor can a port be a reflector port for more than one session at a time.
- It is invisible to all VLANs.
- The native VLAN for looped-back traffic on a reflector port is the RSPAN VLAN.

- The reflector port loops back untagged traffic to the switch. The traffic is then placed on the RSPAN VLAN and flooded to any trunk ports that carry the RSPAN VLAN.
- Spanning tree is automatically disabled on a reflector port.

If the bandwidth of the reflector port is not sufficient for the traffic volume from the corresponding source ports and VLANs, the excess packets are dropped. A 10/100 port reflects at 100 Mbps. A Gigabit port reflects at 1 Gbps.

VLAN-Based SPAN

VLAN-based SPAN (VSPAN) is the monitoring of the network traffic in one or more VLANs. You can configure VSPAN to monitor only received (Rx) traffic, which applies to all the ports for that VLAN.

Use these guidelines for VSPAN sessions:

- Only traffic on the monitored VLAN is sent to the destination port.
- If a destination port belongs to a source VLAN, it is excluded from the source list and is not monitored.
- If ports are added to or removed from the source VLANs, the traffic on the source VLAN received by those ports is added to or removed from the sources being monitored.
- VLAN pruning and the VLAN allowed list have no effect on SPAN monitoring.
- VSPAN only monitors traffic that enters the switch, not traffic that is routed between VLANs. For example, if a VLAN is being Rx-monitored and the multilayer switch routes traffic from another VLAN to the monitored VLAN, that traffic is not monitored and is not received on the SPAN destination port.
- You cannot use filter VLANs in the same session with VLAN sources.
- You can monitor only Ethernet VLANs.

SPAN Traffic

You can use local SPAN to monitor all network traffic, including multicast and bridge protocol data unit (BPDU) packets, and CDP, VTP, DTP, STP, PagP, and LACP packets. You cannot use RSPAN to monitor Layer 2 protocols. See the "RSPAN Configuration Guidelines" section on page 24-16 for more information.

In some SPAN configurations, multiple copies of the same source packet are sent to the SPAN destination port. For example, a bidirectional (both Rx and Tx) SPAN session is configured for the sources a1 Rx monitor and the a2 Rx and Tx monitor to destination port d1. If a packet enters the switch through a1 and is switched to a2, both incoming and outgoing packets are sent to destination port d1. Both packets are the same (unless a Layer 3 rewrite occurs, in which case the packets are different because of the added Layer 3 information).

SPAN and RSPAN Interaction with Other Features

SPAN interacts with these features:

- Routing—Ingress SPAN does not monitor routed traffic. VSPAN only monitors traffic that enters the switch, not traffic that is routed between VLANs. For example, if a VLAN is being Rx-monitored and the multilayer switch routes traffic from another VLAN to the monitored VLAN, that traffic is not monitored and not received on the SPAN destination port.
- Spanning Tree Protocol (STP)—A destination port or a reflector port does not participate in STP while its SPAN or RSPAN session is active. The destination or reflector port can participate in STP after the SPAN or RSPAN session is disabled. On a source port, SPAN does not affect the STP status. STP can be active on trunk ports carrying an RSPAN VLAN.
- Cisco Discovery Protocol (CDP)—A SPAN destination port does not participate in CDP while the SPAN session is active. After the SPAN session is disabled, the port again participates in CDP.
- VLAN Trunking Protocol (VTP)—You can use VTP to prune an RSPAN VLAN between switches.
- VLAN and trunking—You can modify VLAN membership or trunk settings for source, destination, or reflector ports at any time. However, changes in VLAN membership or trunk settings for a destination or reflector port do not take effect until you disable the SPAN or RSPAN session. Changes in VLAN membership or trunk settings for a source port immediately take effect, and the respective SPAN sessions automatically adjust accordingly.
- EtherChannel—You can configure an EtherChannel group as a source port but not as a SPAN destination port. When a group is configured as a SPAN source, the entire group is monitored.

  If a port is added to a monitored EtherChannel group, the new port is added to the SPAN source port list. If a port is removed from a monitored EtherChannel group, it is automatically removed from the source port list. If the port is the only port in the EtherChannel group, the EtherChannel group is removed from SPAN.

  If a physical port that belongs to an EtherChannel group is configured as a SPAN source, destination, or reflector port, it is removed from the group. After the port is removed from the SPAN session, it rejoins the EtherChannel group. Ports removed from an EtherChannel group remain members of the group, but they are in the *down* or *standalone* state.

  If a physical port that belongs to an EtherChannel group is a destination or reflector port and the EtherChannel group is a source, the port is removed from the EtherChannel group and from the list of monitored ports.
- QoS—For ingress monitoring, the packets sent to the SPAN destination port might be different from the packets actually received at the SPAN source port because the packets are forwarded after ingress QoS classification and policing. The packet DSCP might not be the same as the received packet.

  For egress monitoring, the packets sent out the SPAN destination port might not be the same as the packets sent out of SPAN source ports because the egress QoS policing at the SPAN source port might change the packet classification. QoS policing is not applied at SPAN destination ports.
- Multicast traffic can be monitored. For egress and ingress port monitoring, only a single unedited packet is sent to the SPAN destination port. It does not reflect the number of times the multicast packet is sent.

- Port security—A secure port cannot be a SPAN destination port.

  For SPAN sessions, do not enable port security on ports that are egress monitored when ingress forwarding is enabled on the destination port. For RSPAN source sessions, do not enable port security on any ports that are egress monitored.

- IEEE 802.1x—You can enable IEEE 802.1x on a port that is a SPAN destination or reflector port; however, IEEE 802.1x is disabled until the port is removed as a SPAN destination or reflector port. You can enable IEEE 802.1x on a SPAN source port.

  For SPAN sessions, do not enable IEEE 802.1x on ports that are egress monitored when ingress forwarding is enabled on the destination port. For RSPAN source sessions, do not enable IEEE 802.1x on any ports that are egress monitored.

SPAN and RSPAN Session Limits

You can configure (and store in NVRAM) one local SPAN session or multiple RSPAN sessions on a switch. You can configure (and store in NVRAM) a maximum of two SPAN or RSPAN sessions on each switch. You can divide the two sessions between SPAN, RSPAN source, and RSPAN destination sessions. You can configure multiple source ports or source VLANs for each session.

Default SPAN and RSPAN Configuration

Table 24-1 shows the default SPAN and RSPAN configuration.

Table 24-1 Default SPAN and RSPAN Configuration

Feature	Default Setting
SPAN state	Disabled.
Source port traffic to monitor	Both received and sent traffic (**both**); for additional source ports or VLANs, only received (**rx**) traffic can be monitored.
Encapsulation type (destination port)	Native form (no encapsulation type header).
Ingress forwarding (destination port)	Disabled.

Configuring SPAN

This section describes how to configure SPAN on your switch. It contains this configuration information:

- SPAN Configuration Guidelines, page 24-9
- Creating a SPAN Session and Specifying Ports to Monitor, page 24-10
- Creating a SPAN Session and Enabling Ingress Traffic, page 24-11
- Removing Ports from a SPAN Session, page 24-13
- Specifying VLANs to Monitor, page 24-14
- Specifying VLANs to Filter, page 24-15

SPAN Configuration Guidelines

Follow these guidelines when configuring SPAN:

- SPAN sessions can coexist with RSPAN sessions within the limits described in the "SPAN and RSPAN Session Limits" section on page 24-8.
- The destination port cannot be a source port; a source port cannot be a destination port.
- You can have only one destination port per SPAN session. You cannot have two SPAN sessions using the same destination port.
- An EtherChannel port can be a SPAN source port; it cannot be a SPAN destination port.
- An IEEE 802.1x port can be a SPAN source port. You can enable IEEE 802.1x on a port that is a SPAN destination or reflector port; however, IEEE 802.1x is disabled until the port is removed as a SPAN destination or reflector port.
- For SPAN source ports, you can monitor transmitted traffic for a single port and received traffic for a series or range of ports or VLANs.
- When you configure a switch port as a SPAN destination port, it is no longer a normal switch port; only monitored traffic passes through the SPAN destination port.
- A trunk port can be a source port or a destination port. Outgoing packets through the SPAN destination port carry the configured encapsulation headers—either Inter-Switch Link (ISL) or IEEE 802.1Q. If no encapsulation type is defined, the packets are sent in native form.
- You can configure a disabled port to be a source or destination port, but the SPAN function does not start until the destination port and at least one source port or source VLAN are enabled.
- For received traffic, you can mix multiple source port and source VLANs within a single SPAN session. You cannot mix source VLANs and filter VLANs within a SPAN session; you can have source VLANs or filter VLANs, but not both at the same time.
- You can limit SPAN traffic to specific VLANs by using the **filter vlan** keyword. If a trunk port is being monitored, only traffic on the VLANs specified with this keyword is monitored. By default, all VLANs are monitored on a trunk port.
- A SPAN destination port never participates in any VLAN spanning tree. SPAN does include BPDUs in the monitored traffic, so any spanning-tree BPDUs received on the SPAN destination port for a SPAN session were copied from the SPAN source ports.
- When SPAN is enabled, configuration changes have these results:
  - If you change the VLAN configuration of a destination port, the change is not effective until SPAN is disabled.
  - If you disable all source ports or the destination port, the SPAN function stops until both a source and the destination port are enabled.
  - If the source is a VLAN, the number of ports being monitored changes when you move a port in or out of the monitored VLAN.

Creating a SPAN Session and Specifying Ports to Monitor

Beginning in privileged EXEC mode, follow these steps to create a SPAN session and specify the source (monitored) and destination (monitoring) ports:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**}	Clear any existing SPAN configuration for the session. For *session_number*, specify 1 or 2. Specify **all** to remove all SPAN sessions, **local** to remove all local sessions, or **remote** to remove all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source interface** *interface-id* [**,** | **-**] [**both** | **rx** | **tx**]	Specify the SPAN session and the source port (monitored port). For *session_number*, specify 1 or 2. For *interface-id*, specify the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (**port-channel** *port-channel-number*). (Optional) [**,** | **-**] Specify a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. (Optional) Specify the direction of traffic to monitor. If you do not specify a traffic direction, the source interface sends both sent and received traffic. Only received (rx) traffic can be monitored on additional source ports. • **both**—Monitor both received and sent traffic. • **rx**—Monitor received traffic. • **tx**—Monitor sent traffic.
Step 4	**monitor session** *session_number* **destination interface** *interface-id* [**encapsulation** {**dot1q** | **isl**}]	Specify the SPAN session and the destination port (monitoring port). For *session_number*, specify 1 or 2. For *interface-id*, specify the destination port. Valid interfaces include physical interfaces. (Optional) Specify the encapsulation header for outgoing packets. If not specified, packets are sent in native form. • **isl**—Use ISL encapsulation. • **dot1q**—Use IEEE 802.1Q encapsulation.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show monitor** [**session** *session_number*]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to set up a SPAN session, session 1, for monitoring source port traffic to a destination port. First, any existing SPAN configuration for session 1 is cleared, and then bidirectional traffic is mirrored from source port 1 to destination port 8.

```
Switch(config)# no monitor session 1
Switch(config)# monitor session 1 source interface fastethernet0/1
```

```
Switch(config)# monitor session 1 destination interface fastethernet0/8
encapsulation dot1q
Switch(config)# end
```

Creating a SPAN Session and Enabling Ingress Traffic

Beginning in privileged EXEC mode, follow these steps to create a SPAN session, to specify the source and destination ports, and to enable ingress traffic on the destination port for a network security device (such as a Cisco IDS Sensor Appliance):

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**}	Clear any existing SPAN configuration for the session. For *session_number*, specify 1 or 2. Specify **all** to remove all SPAN sessions, **local** to remove all local sessions, or **remote** to remove all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source interface** *interface-id* [**,** | **-**] [**both** | **rx** | **tx**]	Specify the SPAN session and the source port (monitored port). For *session_number*, specify 1 or 2. For *interface-id*, specify the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (**port-channel** *port-channel-number*). (Optional) [**,** | **-**] Specify a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. (Optional) Specify the direction of traffic to monitor. If you do not specify a traffic direction, the source interface sends both sent and received traffic. Only received (rx) traffic can be monitored on additional source ports. • **both**—Monitor both received and sent traffic. • **rx**—Monitor received traffic. • **tx**—Monitor sent traffic.

	Command	Purpose
Step 4	**monitor session** *session_number* **destination interface** *interface-id* [**encapsulation** {**dot1q** [**ingress vlan** *vlan id*] | **isl** [**ingress**]} | **ingress vlan** *vlan id*]	Specify the SPAN session, the destination port (monitoring port), the packet encapsulation, and the ingress VLAN. For *session_number*, specify 1 or 2. For *interface-id*, specify the destination port. Valid interfaces include physical interfaces. (Optional) Specify the encapsulation of the packets transmitted on the SPAN destination port. If no encapsulation is specified, all transmitted packets will be sent in native format (untagged). • Enter **encapsulation dot1q** to send native VLAN packets untagged and all other VLAN **tx** packets tagged **dot1q**. • Enter **encapsulation isl** to send all **tx** packets encapsulated using ISL. • ((Optional) Specify **ingress** to enable forwarding for ingress traffic on the SPAN destination port when using ISL encapsulation. (Optional) For native (untagged) and dot1q encapsulation, specify **ingress vlan** *vlan id* to enable ingress forwarding with *vlan id* as the native VLAN; *vlan id* will also be used as the native VLAN for transmitted packets
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show monitor** [**session** *session_number*]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to configure the destination port for ingress traffic on VLAN 5 by using a security device that does not support IEEE 802.1Q encapsulation.

```
Switch(config)# monitor session 1 destination interface fastethernet0/5 ingress vlan 5
```

This example shows how to configure the destination port for ingress traffic on VLAN 5 by using a security device that supports IEEE 802.1Q encapsulation.

```
Switch(config)# monitor session 1 destination interface fastethernet0/5 encapsulation
dot1q ingress vlan 5
```

This example shows how to disable ingress traffic forwarding on the destination port.

```
Switch(config)# monitor session 1 destination interface fastethernet0/5 encapsulation
dot1q
```

Removing Ports from a SPAN Session

Beginning in privileged EXEC mode, follow these steps to remove a port as a SPAN source for a session:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no monitor session** *session_number* **source interface** *interface-id* [**,** | **-**] [**both** | **rx** | **tx**]	Specify the characteristics of the source port (monitored port) and SPAN session to remove. For *session*, specify 1 or 2. For *interface-id*, specify the source port to no longer monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (**port-channel** *port-channel-number*). (Optional) Use [**,** | **-**] to specify a series or range of interfaces if they were configured. This option is valid when monitoring only received traffic. Enter a space before and after the comma; enter a space before and after the hyphen. (Optional) Specify the direction of traffic (**both**, **rx**, or **tx**) to no longer monitor. If you do not specify a traffic direction, both transmit and receive are disabled.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show monitor** [**session** *session_number*]	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a source or destination port from the SPAN session, use the **no monitor session** *session_number* **source interface** *interface-id* global configuration command or the **no monitor session** *session_number* **destination interface** *interface-id* global configuration command. To change the encapsulation type back to the default (native), use the **monitor session** *session_number* **destination interface** *interface-id* without the **encapsulation** keyword.

This example shows how to remove a port as a SPAN source for SPAN session 1:

```
Switch(config)# no monitor session 1 source interface fastethernet0/1
Switch(config)# end
```

This example shows how to disable received traffic monitoring on a port that was configured for bidirectional monitoring:

```
Switch(config)# no monitor session 1 source interface fastethernet0/1 rx
```

The monitoring of traffic received on port 1 is disabled, but traffic sent from this port continues to be monitored.

Specifying VLANs to Monitor

VLAN monitoring is similar to port monitoring. Beginning in privileged EXEC mode, follow these steps to specify VLANs to monitor:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**}	Clear any existing SPAN configuration for the session. For *session_number*, specify 1 or 2. Specify **all** to remove all SPAN sessions, **local** to remove all local sessions, or **remote** to remove all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source vlan** *vlan-id* [**,** | **-**] **rx**	Specify the SPAN session and the source VLANs (monitored VLANs). You can monitor only received (**rx**) traffic on VLANs. For *session_number*, specify 1 or 2. For *vlan-id*, the range is 1 to 4094; do not enter leading zeros. (Optional) Use a comma (**,**) to specify a series of VLANs, or use a hyphen (**-**) to specify a range of VLANs. Enter a space before and after the comma; enter a space before and after the hyphen.
Step 4	**monitor session** *session_number* **destination interface** *interface-id* [**encapsulation** {**dot1q** | **isl**}]	Specify the SPAN session and the destination port (monitoring port). For *session_number*, specify 1 or 2. For *interface-id*, specify the destination port. Valid interfaces include physical interfaces. (Optional) Specify the encapsulation header for outgoing packets. If not specified, packets are sent in native form. • **isl**—Use ISL encapsulation. • **dot1q**—Use IEEE 802.1Q encapsulation.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show monitor** [**session** *session_number*]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove one or more source VLANs or destination ports from the SPAN session, use the **no monitor session** *session_number* **source vlan** *vlan-id* **rx** global configuration command or the **no monitor session** *session_number* **destination interface** *interface-id* global configuration command.

This example shows how to clear any existing configuration on SPAN session 2, configure SPAN session 2 to monitor received traffic on all ports belonging to VLANs 1 through 3, and send it to destination port 7. The configuration is then modified to also monitor received traffic on all ports belonging to VLAN 10.

```
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source vlan 1 - 3 rx
Switch(config)# monitor session 2 destination interface gigabitethernet0/7
Switch(config)# monitor session 2 source vlan 10 rx
Switch(config)# end
```

Specifying VLANs to Filter

Beginning in privileged EXEC mode, follow these steps to limit SPAN source traffic to specific VLANs:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**}	Clear any existing SPAN configuration for the session. For *session_number*, specify 1 or 2. Specify **all** to remove all SPAN sessions, **local** to remove all local sessions, or **remote** to remove all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source interface** *interface-id* **rx**	Specify the characteristics of the source port (monitored port) and SPAN session. For *session_number*, specify 1 or 2. For *interface-id*, specify the source port to monitor. The interface specified must already be configured as a trunk port.
Step 4	**monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**]	Limit the SPAN source traffic to specific VLANs. For *session_number*, specify 1 or 2. For *vlan-id*, the range is 1 to 4094; do not enter leading zeros. (Optional) Use a comma (**,**) to specify a series of VLANs or use a hyphen (**-**) to specify a range of VLANs. Enter a space before and after the comma; enter a space before and after the hyphen.
Step 5	**monitor session** *session_number* **destination interface** *interface-id*	Specify the characteristics of the destination port (monitoring port) and SPAN session. For *session_number*, specify 1 or 2. For *interface-id*, specify the destination port. Valid interfaces include physical interfaces.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show monitor** [**session** *session_number*]	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To monitor all VLANs on the trunk port, use the **no monitor session** *session_number* **filter** global configuration command.

This example shows how to clear any existing configuration on SPAN session 2, configure SPAN session 2 to monitor traffic received on trunk port 4, and send traffic for only VLANs 1 through 5 and 9 to destination port 8.

```
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source interface gigabitethernet0/4 rx
Switch(config)# monitor session 2 filter vlan 1 - 5, 9
Switch(config)# monitor session 2 destination interface gigabitethernet0/8
Switch(config)# end
```

Configuring RSPAN

This section describes how to configure RSPAN on your switch. It contains this configuration information:

- RSPAN Configuration Guidelines, page 24-16
- Configuring a VLAN as an RSPAN VLAN, page 24-17
- Creating an RSPAN Source Session, page 24-18
- Creating an RSPAN Destination Session, page 24-19
- Creating an RSPAN Destination Session and Enabling Ingress Traffic, page 24-20
- Removing Ports from an RSPAN Session, page 24-21
- Specifying VLANs to Monitor, page 24-21
- Specifying VLANs to Filter, page 24-23

RSPAN Configuration Guidelines

Follow these guidelines when configuring RSPAN:

- All the items in the "SPAN Configuration Guidelines" section on page 24-9 apply to RSPAN.
- As RSPAN VLANs have special properties, you should reserve a few VLANs across your network for use as RSPAN VLANs; do not assign access ports to these VLANs.
- You can apply an output access control list (ACL) to RSPAN traffic to selectively filter or monitor specific packets. Specify these ACLs on the RSPAN VLAN in the RSPAN source switches.
- RSPAN sessions can coexist with SPAN sessions within the limits described in the "SPAN and RSPAN Session Limits" section on page 24-8.
- For RSPAN configuration, you can distribute the source ports and the destination ports across multiple switches in your network.
- A port cannot serve as an RSPAN source port or RSPAN destination port while designated as an RSPAN reflector port.
- When you configure a switch port as a reflector port, it is no longer a normal switch port; only looped-back traffic passes through the reflector port.
- RSPAN does not support BPDU packet monitoring or other Layer 2 switch protocols.
- The RSPAN VLAN is configured only on trunk ports and not on access ports. To avoid unwanted traffic in RSPAN VLANs, make sure that the VLAN remote-span feature is supported in all the participating switches. Access ports on the RSPAN VLAN are silently disabled.
- RSPAN VLANs are included as sources for port-based RSPAN sessions when source trunk ports have active RSPAN VLANs. RSPAN VLANs can also be sources in SPAN sessions.

- You can configure any VLAN as an RSPAN VLAN as long as these conditions are met:
  - No access port is configured in the RSPAN VLAN.
  - The same RSPAN VLAN is used for an RSPAN session in all the switches.
  - All participating switches support RSPAN.

Note The RSPAN VLAN cannot be VLAN 1 (the default VLAN) or VLAN IDs 1002 through 1005 (reserved to Token Ring and FDDI VLANs).

- You should create an RSPAN VLAN before configuring an RSPAN source or destination session.
- If you enable VTP and VTP pruning, RSPAN traffic is pruned in the trunks to prevent the unwanted flooding of RSPAN traffic across the network for VLAN-IDs that are lower than 1005.
- Because RSPAN traffic is carried across a network on an RSPAN VLAN, the original VLAN association of the mirrored packets is lost. Therefore, RSPAN can only support forwarding of traffic from an IDS device onto a single user-specified VLAN.

Configuring a VLAN as an RSPAN VLAN

First create a new VLAN to be the RSPAN VLAN for the RSPAN session. You must create the RSPAN VLAN in all switches that will participate in RSPAN. If the RSPAN VLAN-ID is in the normal range (lower than 1005) and VTP is enabled in the network, you can create the RSPAN VLAN in one switch, and VTP propagates it to the other switches in the VTP domain. For extended-range VLANs (greater than 1005), you must configure RSPAN VLAN on both source and destination switches and any intermediate switches.

Use VTP pruning to get an efficient flow of RSPAN traffic, or manually delete the RSPAN VLAN from all trunks that do not need to carry the RSPAN traffic.

Beginning in privileged EXEC mode, follow these steps to create an RSPAN VLAN:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vlan** *vlan-id*	Enter a VLAN ID to create a VLAN, or enter the VLAN ID of an existing VLAN, and enter VLAN configuration mode. The range is 2 to 1001 and 1006 to 4094. **Note** The RSPAN VLAN cannot be VLAN 1 (the default VLAN) or VLAN IDs 1002 through 1005 (reserved for Token Ring and FDDI VLANs).
Step 3	remote-span	Configure the VLAN as an RSPAN VLAN.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**copy running-config startup-config**	(Optional) Save the configuration in the configuration file.

To remove the remote SPAN characteristic from a VLAN and convert it back to a normal VLAN, use the **no remote-span** VLAN configuration command.

This example shows how to create RSPAN VLAN 901.

```
Switch(config)# vlan 901
Switch(config-vlan)# remote span
Switch(config-vlan)# end
```

Creating an RSPAN Source Session

Beginning in privileged EXEC mode, follow these steps to start an RSPAN source session and to specify the monitored source and the destination RSPAN VLAN:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**}	Clear any existing RSPAN configuration for the session. For *session_number*, specify 1 or 2. Specify **all** to remove all RSPAN sessions, **local** to remove all local sessions, or **remote** to remove all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source interface** *interface-id* [**,** | **-**] [**both** | **rx** | **tx**]	Specify the RSPAN session and the source port (monitored port). For *session_number*, specify 1 or 2. For *interface-id*, specify the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (**port-channel** *port-channel-number*). (Optional) [**,** | **-**] Specify a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. (Optional) Specify the direction of traffic to monitor. If you do not specify a traffic direction, the source interface sends both sent and received traffic. Only received (**rx**) traffic can be monitored on additional source ports. • **both**—Monitor both received and sent traffic. • **rx**—Monitor received traffic. • **tx**—Monitor sent traffic.
Step 4	**monitor session** *session_number* **destination remote vlan** *vlan-id* **reflector-port** *interface*	Specify the RSPAN session, the destination remote VLAN, and the reflector port. For *session_number*, enter 1 or 2. For *vlan-id*, specify the RSPAN VLAN to carry the monitored traffic to the destination port. (See the "Creating or Modifying an Ethernet VLAN" section on page 11-8 for more information about creating an RSPAN VLAN.) For *interface*, specify the interface that will flood the RSPAN traffic onto the RSPAN VLAN.
Step 5	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 6	**show monitor** [**session** *session_number*]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to clear any existing RSPAN configuration for session 1, configure RSPAN session 1 to monitor multiple source interfaces, and configure the destination RSPAN VLAN and the reflector-port.

```
Switch(config)# no monitor session 1
Switch(config)# monitor session 1 source interface fastethernet0/10 tx
Switch(config)# monitor session 1 source interface fastethernet0/2 rx
Switch(config)# monitor session 1 source interface fastethernet0/3 rx
Switch(config)# monitor session 1 source interface port-channel 102 rx
Switch(config)# monitor session 1 destination remote vlan 901 reflector-port
fastethernet0/1
Switch(config)# end
```

Creating an RSPAN Destination Session

Beginning in privileged EXEC mode, follow these steps to create an RSPAN destination session and to specify the source RSPAN VLAN and the destination port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**monitor session** *session_number* **source remote vlan** *vlan-id*	Specify the RSPAN session and the source RSPAN VLAN. For *session_number*, specify 1 or 2. For *vlan-id*, specify the source RSPAN VLAN to monitor.
Step 3	**monitor session** *session_number* **destination interface** *interface-id* [**encapsulation** {**dot1q** | **isl**}]	Specify the RSPAN session and the destination interface. For *session_number*, specify 1 or 2. For *interface-id*, specify the destination interface. (Optional) Specify the encapsulation header for outgoing packets. If not specified, packets are sent in native form. • **isl**—Use ISL encapsulation. • **dot1q**—Use IEEE 802.1Q encapsulation.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show monitor** [**session** *session_number*]	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to configure VLAN 901 as the source remote VLAN and port 5 as the destination interface:

```
Switch(config)# monitor session 1 source remote vlan 901
Switch(config)# monitor session 1 destination interface fastethernet0/5
Switch(config)# end
```

Creating an RSPAN Destination Session and Enabling Ingress Traffic

Beginning in privileged EXEC mode, follow these steps to create an RSPAN destination session, to specify the source RSPAN VLAN, and to enable ingress traffic on the destination port for a network security device (such as a Cisco IDS Sensor Appliance):

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**monitor session** *session_number* **source remote vlan** *vlan-id*	Specify the RSPAN session and the source RSPAN VLAN. For *session_number*, specify 1 or 2. For *vlan-id*, specify the source RSPAN VLAN to monitor.
Step 3	**monitor session** *session_number* **destination interface** *interface-id* [**encapsulation** {**dot1q** [**ingress vlan** *vlan id*] | **ISL** [**ingress**]} | **ingress vlan** *vlan id*]	Specify the RSPAN session, the destination port, the packet encapsulation, and the ingress VLAN. For *session_number*, specify 1 or 2. For *interface-id*, specify the destination port. Valid interfaces include physical interfaces. (Optional) Specify the encapsulation of the packets transmitted on the RSPAN destination port. If no encapsulation is specified, all transmitted packets will be sent in native format (untagged). • Enter **encapsulation dot1q** to send native VLAN packets untagged and all other VLAN **tx** packets tagged **dot1q**. • Enter **encapsulation isl** to send all **tx** packets encapsulated using ISL. (Optional) Specify whether forwarding is enabled for ingress traffic on the SPAN destination port. • For native (untagged) and dot1q encapsulation, specify **ingress vlan** *vlan id* to enable ingress forwarding with *vlan id* as the native VLAN; *vlan id* will also be used as the native VLAN for transmitted packets. • Specify **ingress** to enable ingress forwarding when using ISL encapsulation.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show monitor** [**session** *session_number*]	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to configure VLAN 901 as the source remote VLAN and how to configure the destination port for ingress traffic on VLAN 5 by using a security device that supports IEEE 802.1Q encapsulation:

```
Switch(config)# monitor session 1 source remote vlan 901
Switch(config)# monitor session 1 destination interface fastEthernet0/5 ingress vlan 5
Switch(config)# end
```

Removing Ports from an RSPAN Session

Beginning in privileged EXEC mode, follow these steps to remove a port as an RSPAN source for a session:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no monitor session** *session_number* **source interface** *interface-id* [**,** | **-**] [**both** | **rx** | **tx**]	Specify the characteristics of the RSPAN source port (monitored port) to remove. For *session_number*, specify 1 or 2. For *interface-id*, specify the source port to no longer monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (**port-channel** *port-channel-number*). (Optional) Use [**,** | **-**] to specify a series or range of interfaces if they were configured. Enter a space before and after the comma; enter a space before and after the hyphen. (Optional) Specify the direction of traffic (**both**, **rx**, or **tx**) to no longer monitor. If you do not specify a traffic direction, both transmit and receive are disabled.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show monitor** [**session** *session_number*]	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to remove port 1 as an RSPAN source for RSPAN session 1:

```
Switch(config)# no monitor session 1 source interface fastEthernet0/1
Switch(config)# end
```

This example shows how to disable received traffic monitoring on port 1, which was configured for bidirectional monitoring:

```
Switch(config)# no monitor session 1 source interface fastEthernet0/1 rx
```

The monitoring of traffic received on port 1 is disabled, but traffic sent from this port continues to be monitored.

Specifying VLANs to Monitor

VLAN monitoring is similar to port monitoring. Beginning in privileged EXEC mode, follow these steps to specify VLANs to monitor:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**}	Clear any existing SPAN configuration for the session. For *session_number*, specify 1 or 2. Specify **all** to remove all SPAN sessions, **local** to remove all local sessions, or **remote** to remove all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source vlan** *vlan-id* [**,** | **-**] **rx**	Specify the RSPAN session and the source VLANs (monitored VLANs). You can monitor only received (**rx**) traffic on VLANs. For *session_number*, specify 1 or 2. For *vlan-id*, the range is 1 to 4094; do not enter leading zeros. (Optional) Use a comma (**,**) to specify a series of VLANs, or use a hyphen (**-**) to specify a range of VLANs. Enter a space before and after the comma; enter a space before and after the hyphen.
Step 4	**monitor session** *session_number* **destination remote vlan** *vlan-id* **reflector port** *interface*	Specify the RSPAN session, the destination remote VLAN, and the reflector port. For *session_number*, enter 1 or 2. For *vlan-id*, specify the RSPAN VLAN to carry the monitored traffic to the destination port. For *interface*, specify the interface that will flood the RSPAN traffic to the RSPAN VLAN.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show monitor** [**session** *session_number*]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove one or more source VLANs from the RSPAN session, use the **no monitor session** *session_number* **source vlan** *vlan-id* **rx** global configuration command.

This example shows how to clear any existing configuration on RSPAN session 2, configure RSPAN session 2 to monitor received traffic on all ports belonging to VLANs 1 through 3, and send it to destination remote VLAN 902 using reflector port 7. The configuration is then modified to also monitor received traffic on all ports belonging to VLAN 10.

```
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source vlan 1 - 3 rx
Switch(config)# monitor session 2 destination remote vlan 902 reflector-port
gigabitethernet0/7
Switch(config)# monitor session 2 source vlan 10 rx
Switch(config)# end
```

Specifying VLANs to Filter

Beginning in privileged EXEC mode, follow these steps to limit RSPAN source traffic to specific VLANs:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**}	Clear any existing SPAN configuration for the session. For *session_number*, specify 1 or 2. Specify **all** to remove all SPAN sessions, **local** to remove all local sessions, or **remote** to remove all remote SPAN sessions.
Step 3	**monitor session** *session_number* **source interface** *interface-id* **rx**	Specify the characteristics of the source port (monitored port) and RSPAN session. For *session_number*, specify 1 or 2. For *interface-id*, specify the source port to monitor. The interface specified must already be configured as a trunk port.
Step 4	**monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**]	Limit the RSPAN source traffic to specific VLANs. For *session_number*, specify 1 or 2. For *vlan-id*, the range is 1 to 4094; do not enter leading zeros. (Optional) Use a comma (**,**) to specify a series of VLANs or use a hyphen (**-**) to specify a range of VLANs. Enter a space before and after the comma; enter a space before and after the hyphen.
Step 5	**monitor session** *session_number* **destination remote vlan** *vlan-id* **reflector port** *interface*	Specify the RSPAN session, the destination remote VLAN, and the reflector port. For *session_number*, enter 1 or 2. For *vlan-id*, specify the RSPAN VLAN to carry the monitored traffic to the destination port. For *interface*, specify the interface that will flood the RSPAN traffic to the RSPAN VLAN.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show monitor** [**session** *session_number*]	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To monitor all VLANs on the trunk port, use the **no monitor session** *session_number* **filter vlan** global configuration command.

This example shows how to clear any existing configuration on RSPAN session 2, configure RSPAN session 2 to monitor traffic received on trunk port 4, and send traffic for only VLANs 1 through 5 and 9 to destination remote VLAN 902 with port 8 as the reflector port.

```
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source interface gigabitethernet0/4 rx
Switch(config)# monitor session 2 filter vlan 1 - 5 , 9
Switch(config)# monitor session 2 destination remote vlan 902 reflector-port
gigabitethernet0/8
Switch(config)# end
```

Displaying SPAN and RSPAN Status

To display the status of the current SPAN or RSPAN configuration, use the **show monitor** privileged EXEC command.

This is an example of output for the **show monitor** privileged EXEC command for SPAN source session 1:

```
Switch# show monitor session 1
Session 1
---------
Type              : Local Session
Source Ports      :
    RX Only       : None
    TX Only       : None
    Both          : Fa0/4
Source VLANs      :
    RX Only       : None
    TX Only       : None
    Both          : None
Source RSPAN VLAN : None
Destination Ports : Fa0/5
    Encapsulation: DOT1Q
          Ingress: Enabled, default VLAN = 5
Reflector Port    : None
Filter VLANs      : None
Dest RSPAN VLAN   : None
```

CHAPTER 25

Configuring RMON

This chapter describes how to configure Remote Network Monitoring (RMON) on your Catalyst 3550 switch. RMON is a standard monitoring specification that defines a set of statistics and functions that can be exchanged between RMON-compliant console systems and network probes. RMON provides you with comprehensive network-fault diagnosis, planning, and performance-tuning information.

Note For complete syntax and usage information for the commands used in this chapter, see the "System Management Commands" section in the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

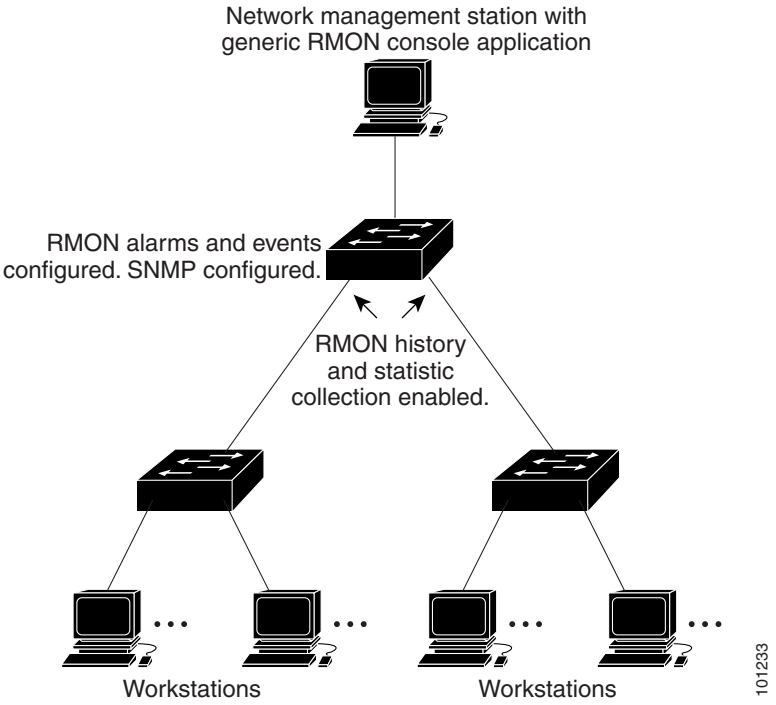
This chapter consists of these sections:

- Understanding RMON, page 25-1
- Configuring RMON, page 25-2
- Displaying RMON Status, page 25-6

Understanding RMON

RMON is an Internet Engineering Task Force (IETF) standard monitoring specification that allows various network agents and console systems to exchange network monitoring data. You can use the RMON feature with the Simple Network Management Protocol (SNMP) agent in the switch to monitor all the traffic flowing among switches on all connected LAN segments.

Figure 25-1 Remote Monitoring Example



The switch supports these RMON groups (defined in RFC 1757):

- Statistics (RMON group 1)—Collects Ethernet, Fast Ethernet, and Gigabit Ethernet statistics on an interface.
- History (RMON group 2)—Collects a history group of statistics on Ethernet, Fast Ethernet, and Gigabit Ethernet interfaces for a specified polling interval.
- Alarm (RMON group 3)—Monitors a specific MIB object for a specified interval, triggers an alarm at a specified value (rising threshold), and resets the alarm at another value (falling threshold). Alarms can be used with events; the alarm triggers an event, which can generate a log entry or an SNMP trap.
- Event (RMON group 9)—Determines the action to take when an event is triggered by an alarm. The action can be to generate a log entry or an SNMP trap.

Because switches supported by this software release use hardware counters for RMON data processing, the monitoring is more efficient, and little processing power is required.

Configuring RMON

This section describes how to configure RMON on your switch. It contains this configuration information:

- Default RMON Configuration, page 25-3
- Configuring RMON Alarms and Events, page 25-3
- Configuring RMON Collection on an Interface, page 25-4

Default RMON Configuration

RMON is disabled by default; no alarms or events are configured.

Only RMON 1 is supported on the switch.

Configuring RMON Alarms and Events

You can configure your switch for RMON by using the command-line interface (CLI) or an SNMP-compatible network management station. We recommend that you use a generic RMON console application on the network management station (NMS) to take advantage of RMON's network management capabilities. You must also configure SNMP on the switch to access RMON MIB objects. For more information, see Chapter 27, "Configuring SNMP."

Beginning in privileged EXEC mode, follow these steps to enable RMON alarms and events:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**rmon alarm** *number variable interval* {**absolute** | **delta**} **rising-threshold** *value* [*event-number*] **falling-threshold** *value* [*event-number*] [**owner** *string*]	Set an alarm on a MIB object. • For *number*, specify the alarm number. The range is 1 to 65535. • For *variable*, specify the MIB object to monitor. • For *interval*, specify the time in seconds the alarm monitors the MIB variable. The range is 1 to 4294967295 seconds. • Specify the **absolute** keyword to test each MIB variable directly; specify the **delta** keyword to test the change between samples of a MIB variable. • For *value*, specify a number at which the alarm is triggered and one for when the alarm is reset. The range for the rising threshold and falling threshold *values* is -2147483648 to 2147483647. • (Optional) For *event-number*, specify the event number to trigger when the rising or falling threshold exceeds its limit. • (Optional) For **owner** *string*, specify the owner of the alarm.

	Command	Purpose
Step 3	**rmon event** *number* [**description** *string*] [**log**] [**owner** *string*] [**trap** *community*]	Add an event in the RMON event table that is associated with an RMON event number. • For *number*, assign an event number. The range is 1 to 65535. • (Optional) For **description** *string*, specify a description of the event. • (Optional) Use the **log** keyword to generate an RMON log entry when the event is triggered. • (Optional) For **owner** *string*, specify the owner of this event. • (Optional) For *community*, enter the SNMP community string used for this trap.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable an alarm, use the **no rmon alarm** *number* global configuration command on each alarm you configured. You cannot disable at once all the alarms that you configured. To disable an event, use the **no rmon event** *number* global configuration command. To learn more about alarms and events and how they interact with each other, see RFC 1757.

You can set an alarm on any MIB object. The following example configures RMON alarm number 10 by using the **rmon alarm** command. The alarm monitors the MIB variable *ifEntry.20.1* once every 20 seconds until the alarm is disabled and checks the change in the variable's rise or fall. If the *ifEntry.20.1* value shows a MIB counter increase of 15 or more, such as from 100000 to 100015, the alarm is triggered. The alarm in turn triggers event number 1, which is configured with the **rmon event** command. Possible events can include a log entry or an SNMP trap. If the *ifEntry.20.1* value changes by 0, the alarm is reset and can be triggered again.

```
Switch(config)# rmon alarm 10 ifEntry.20.1 20 delta rising-threshold 15 1
falling-threshold 0 owner jjohnson
```

The following example creates RMON event number 1 by using the **rmon event** command. The event is defined as *High ifOutErrors* and generates a log entry when the event is triggered by the alarm. The user *jjones* owns the row that is created in the event table by this command. This example also generates an SNMP trap when the event is triggered.

```
Switch(config)# rmon event 1 log trap eventtrap description "High ifOutErrors" owner
jjones
```

Configuring RMON Collection on an Interface

You must first configure RMON alarms and events to display collection information.

Beginning in privileged EXEC mode, follow these steps to collect group history statistics on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface on which to collect history, and enter interface configuration mode.
Step 3	**rmon collection history** *index* [**buckets** *bucket-number*] [**interval** *seconds*] [**owner** *ownername*]	Enable history collection for the specified number of buckets and time period. • For *index*, identify the RMON group of statistics The range is 1 to 65535. • (Optional) For **buckets** *bucket-number,* specify the maximum number of buckets desired for the RMON collection history group of statistics. The range is 1 to 65535. The default is 50 buckets. • (Optional) For **interval** *seconds*, specify the number of seconds in each polling cycle. • (Optional) For **owner** *ownername*, enter the name of the owner of the RMON group of statistics.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**show rmon history**	Display the contents of the switch history table.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable history collection, use the **no rmon collection history** *index* interface configuration command.

Beginning in privileged EXEC mode, follow these steps to collect group Ethernet statistics on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface on which to collect statistics, and enter interface configuration mode.
Step 3	**rmon collection stats** *index* [**owner** *ownername*]	Enable RMON statistic collection on the interface. • For *index*, specify the RMON group of statistics. The range is from 1 to 65535. • (Optional) For **owner** *ownername*, enter the name of the owner of the RMON group of statistics.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**show rmon statistics**	Display the contents of the switch statistics table.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the collection of group Ethernet statistics, use the **no rmon collection stats** *index* interface configuration command.

Displaying RMON Status

To display the RMON status, use one or more of the privileged EXEC commands in Table 25-1:

Table 25-1 Commands for Displaying RMON Status

Command	Purpose
show rmon	Displays general RMON statistics.
show rmon alarms	Displays the RMON alarm table.
show rmon events	Displays the RMON event table.
show rmon history	Displays the RMON history table.
show rmon statistics	Displays the RMON statistics table.

For information about the fields in these displays, see the "System Management Commands" section in the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

CHAPTER 26

Configuring System Message Logging

This chapter describes how to configure system message logging on your Catalyst 3550 switch.

Note For complete syntax and usage information for the commands used in this chapter, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

This chapter consists of these sections:

- Understanding System Message Logging, page 26-1
- Configuring System Message Logging, page 26-2
- Displaying the Logging Configuration, page 26-12

Caution Logging messages to the console at a high rate can cause high CPU utilization and adversely affect how the switch operates.

Understanding System Message Logging

By default, a switch sends the output from system messages and **debug** privileged EXEC commands to a logging process. The logging process controls the distribution of logging messages to various destinations, such as the logging buffer, terminal lines, or a UNIX syslog server, depending on your configuration. The process also sends messages to the console.

Note The syslog format is compatible with 4.3 BSD UNIX.

When the logging process is disabled, messages are sent only to the console. The messages are sent as they are generated, so message and debug output are interspersed with prompts or output from other commands. Messages appear on the console after the process that generated them has finished.

You can set the severity level of the messages to control the type of messages displayed on the console and each of the destinations. You can timestamp log messages or set the syslog source address to enhance real-time debugging and management. For information on possible messages, see the system message guide for this release.

You can access logged system messages by using the switch command-line interface (CLI) or by saving them to a properly configured syslog server. The switch software saves syslog messages in an internal buffer. You can remotely monitor system messages by accessing the switch through Telnet, through the console port, or by viewing the logs on a syslog server.

Configuring System Message Logging

These sections describe how to configure system message logging:

- System Log Message Format, page 26-2
- Default System Message Logging Configuration, page 26-3
- Disabling and Enabling Message Logging, page 26-4
- Setting the Message Display Destination Device, page 26-5
- Synchronizing Log Messages, page 26-6
- Enabling and Disabling Timestamps on Log Messages, page 26-7
- Enabling and Disabling Sequence Numbers in Log Messages, page 26-8
- Defining the Message Severity Level, page 26-8
- Limiting Syslog Messages Sent to the History Table and to SNMP, page 26-10
- Configuring UNIX Syslog Servers, page 26-10

System Log Message Format

System log messages can contain up to 80 characters and a percent sign (%), which follows the optional sequence number or timestamp information, if configured. Messages appear in this format:

seq no:timestamp: %facility-severity-MNEMONIC:description

The part of the message preceding the percent sign depends on the setting of the **service sequence-numbers**, **service timestamps log datetime**, **service timestamps log datetime** [**localtime**] [**msec**] [**show-timezone**], or **service timestamps log uptime** global configuration command.

Table 26-1 describes the elements of syslog messages.

Table 26-1 System Log Message Elements

Element	Description
seq no:	Stamps log messages with a sequence number only if the **service sequence-numbers** global configuration command is configured. For more information, see the "Enabling and Disabling Sequence Numbers in Log Messages" section on page 26-8.
timestamp formats: *mm/dd hh:mm:ss* or *hh:mm:ss* (short uptime) or *d h* (long uptime)	Date and time of the message or event. This information appears only if the **service timestamps log** [**datetime** | **log**] global configuration command is configured. For more information, see the "Enabling and Disabling Timestamps on Log Messages" section on page 26-7.
facility	The facility to which the message refers (for example, SNMP, SYS, and so forth). For a list of supported facilities, see Table 26-4 on page 26-12.
severity	Single-digit code from 0 to 7 that is the severity of the message. For a description of the severity levels, see Table 26-3 on page 26-9.
MNEMONIC	Text string that uniquely describes the message.
description	Text string containing detailed information about the event being reported.

This example shows a partial switch system message:

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/2, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed
state to down 2
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
*Mar  1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```

Default System Message Logging Configuration

Table 26-2 shows the default system message logging configuration.

Table 26-2 Default System Message Logging Configuration

Feature	Default Setting
System message logging to the console	Enabled.
Console severity	Debugging (and numerically lower levels; see Table 26-3 on page 26-9).
Logging buffer size	4096 bytes.
Logging history size	1 message.
Timestamps	Disabled.
Synchronous logging	Disabled.

Table 26-2 Default System Message Logging Configuration (continued)

Feature	Default Setting
Logging server	Disabled.
Syslog server IP address	None configured.
Server facility	Local7 (see Table 26-4 on page 26-12).
Server severity	Informational (and numerically lower levels; see Table 26-3 on page 26-9).

Disabling and Enabling Message Logging

Message logging is enabled by default. It must be enabled to send messages to any destination other than the console. When enabled, log messages are sent to a logging process, which logs messages to designated locations asynchronously to the processes that generated the messages.

Beginning in privileged EXEC mode, follow these steps to disable message logging:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no logging console**	Disable message logging.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config** or **show logging**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Disabling the logging process can slow down the switch because a process must wait until the messages are written to the console before continuing. When the logging process is disabled, messages appear on the console as soon as they are produced, often appearing in the middle of command output.

The **logging synchronous** global configuration command also affects the display of messages to the console. When this command is enabled, messages appear only after you press Return. For more information, see the "Synchronizing Log Messages" section on page 26-6.

To re-enable message logging after it has been disabled, use the **logging on** global configuration command.

Setting the Message Display Destination Device

If message logging is enabled, you can send messages to specific locations in addition to the console. Beginning in privileged EXEC mode, use one or more of the following commands to specify the locations that receive messages:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**logging buffered** [*size*]	Log messages to an internal buffer. The default buffer size is 4096. The range is 4096 to 4294967295 bytes. **Note** Do not make the buffer size too large because the switch could run out of memory for other tasks. Use the **show memory** privileged EXEC command to view the free processor memory on the switch; however, this value is the maximum available, and the buffer size should *not* be set to this amount.
Step 3	**logging** *host*	Log messages to a UNIX syslog server host. For *host*, specify the name or IP address of the host to be used as the syslog server. To build a list of syslog servers that receive logging messages, enter this command more than once. For complete syslog server configuration steps, see the "Configuring UNIX Syslog Servers" section on page 26-10.
Step 4	**logging file flash:***filename* [*max-file-size*] [*min-file-size*] [*severity-level-number* | *type*]	Store log messages in a file in flash memory. • For *filename*, enter the log message filename. • (Optional) For *max-file-size, s*pecify the maximum logging file size. The range is 4096 to 2147483647. The default is 4069 bytes. • (Optional) For *min-file-size, s*pecify the minimum logging file size. The range is 1024 to 2147483647. The default is 2048 bytes. • (Optional) For *severity-level-number* | *type,* specify either the logging severity level or the logging type. The severity range is 0 to 7. For a list of logging type keywords, see Table 26-3 on page 26-9. By default, the log file receives debugging messages and numerically lower levels.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**terminal monitor**	Log messages to a nonconsole terminal during the current session. Terminal parameter-setting commands are set locally and do not remain in effect after the session has ended. You must perform this step for each session to see the debugging messages.
Step 7	**show running-config**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The **logging buffered** global configuration command copies logging messages to an internal buffer. The buffer is circular, so newer messages overwrite older messages after the buffer is full. To display the messages that are logged in the buffer, use the **show logging** privileged EXEC command. The first message displayed is the oldest message in the buffer. To clear the contents of the buffer, use the **clear logging** privileged EXEC command.

To disable logging to the console, use the **no logging console** global configuration command. To disable logging to a file, use the **no logging file** [*severity-level-number* | *type*] global configuration command.

Synchronizing Log Messages

You can configure the system to synchronize unsolicited messages and **debug** privileged EXEC command output with solicited device output and prompts for a specific console port line or virtual terminal line. You can identify the types of messages to be output asynchronously based on the level of severity. You can also determine the maximum number of buffers for storing asynchronous messages for the terminal after which messages are dropped.

When synchronous logging of unsolicited messages and **debug** command output is enabled, unsolicited device output appears on the console or is printed after solicited device output appears or is printed. Unsolicited messages and **debug** command output appears on the console after the prompt for user input is returned. Therefore, unsolicited messages and **debug** command output are not interspersed with solicited device output and prompts. After the unsolicited messages appear, the console again displays the user prompt.

Beginning in privileged EXEC mode, follow these steps to configure synchronous logging:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**line** [**console** | **vty**] *line-number* [*ending-line-number*]	Specify the line to be configured for synchronous logging of messages. • Use the **console** keyword for configurations that occur through the switch console port. • Use the **line vty** *line-number* command to specify which vty lines are to have synchronous logging enabled. You use a vty connection for configurations that occur through a Telnet session. The range of line numbers is from 0 to 15. You can change the setting of all 16 vty lines at once by entering: **line vty 0 15** Or you can change the setting of the single vty line being used for your current connection. For example, to change the setting for vty line 2, enter: **line vty 2** When you enter this command, the mode changes to line configuration.

	Command	Purpose
Step 3	**logging synchronous** [**level** *severity-level* | **all**] [**limit** *number-of-buffers*]	Enable synchronous logging of messages. • (Optional) For **level** *severity-level*, specify the message severity level. Messages with a severity level equal to or higher than this value are printed asynchronously. Low numbers mean greater severity and high numbers mean lesser severity. The default is 2. • (Optional) Specifying **level all** means that all messages are printed asynchronously regardless of the severity level. • (Optional) For **limit** *number-of-buffers*, specify the number of buffers to be queued for the terminal after which new messages are dropped. The default is 20.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable synchronization of unsolicited messages and debug output, use the **no logging synchronous** [**level** *severity-level* | **all**] [**limit** *number-of-buffers*] line configuration command.

Enabling and Disabling Timestamps on Log Messages

By default, log messages are not timestamped.

Beginning in privileged EXEC mode, follow these steps to enable timestamping of log messages:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**service timestamps log uptime** or **service timestamps log datetime** [**msec**] [**localtime**] [**show-timezone**]	Enable log timestamps. The first command enables timestamps on log messages, showing the time since the system was rebooted. The second command enables timestamps on log messages. Depending on the options selected, the timestamp can include the date, time in milliseconds relative to the local time zone, and the time zone name.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable timestamps for both debug and log messages, use the **no service timestamps** global configuration command.

This example shows part of a logging display with the **service timestamps log datetime** global configuration command enabled:

```
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```

This example shows part of a logging display with the **service timestamps log uptime** global configuration command enabled:

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up
```

Enabling and Disabling Sequence Numbers in Log Messages

Because there is a chance that more than one log message can have the same timestamp, you can display messages with sequence numbers so that you can unambiguously see a single message. By default, sequence numbers in log messages are not displayed.

Beginning in privileged EXEC mode, follow these steps to enable sequence numbers in log messages:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**service sequence-numbers**	Enable sequence numbers.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable sequence numbers, use the **no service sequence-numbers** global configuration command.

This example shows part of a logging display with sequence numbers enabled:

```
000019: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```

Defining the Message Severity Level

You can limit messages displayed to the selected device by specifying the severity level of the message, which are described in Table 26-3.

Beginning in privileged EXEC mode, follow these steps to define the message severity level:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**logging console** *level*	Limit messages logged to the console. By default, the console receives debugging messages and numerically lower levels (see Table 26-3 on page 26-9).
Step 3	**logging monitor** *level*	Limit messages logged to the terminal lines. By default, the terminal receives debugging messages and numerically lower levels (see Table 26-3 on page 26-9).
Step 4	**logging trap** *level*	Limit messages logged to the syslog servers. By default, syslog servers receive informational messages and numerically lower levels (see Table 26-3 on page 26-9). For complete syslog server configuration steps, see the "Configuring UNIX Syslog Servers" section on page 26-10.

	Command	Purpose
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config** or **show logging**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note Specifying a *level* causes messages at that level and numerically lower levels to appear at the destination.

To disable logging to the console, use the **no logging console** global configuration command. To disable logging to a terminal other than the console, use the **no logging monitor** global configuration command. To disable logging to syslog servers, use the **no logging trap** global configuration command.

Table 26-3 describes the *level* keywords. It also lists the corresponding UNIX syslog definitions from the most severe level to the least severe level.

Table 26-3 Message Logging Level Keywords

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

The software generates four other categories of messages:

- Error messages about software or hardware malfunctions that appear at levels **warnings** through **emergencies**. These types of messages mean that the functionality of the switch is affected. For information on how to recover from these malfunctions, see the system message guide for this release.
- Output from the **debug** commands, displayed at the **debugging** level. Debug commands are typically used only by the Technical Assistance Center.
- Interface up or down transitions and system restart messages, displayed at the **notifications** level. This message is only for information; switch functionality is not affected.
- Reload requests and low-process stack messages, displayed at the **informational** level. This message is only for information; switch functionality is not affected.

Limiting Syslog Messages Sent to the History Table and to SNMP

If you enabled syslog message traps to be sent to an SNMP network management station by using the **snmp-server enable trap** global configuration command, you can change the level of messages sent and stored in the switch history table. You also can change the number of messages that are stored in the history table.

Messages are stored in the history table because SNMP traps are not guaranteed to reach their destination. By default, one message of the level **warning** and numerically lower levels (see Table 26-3 on page 26-9) are stored in the history table even if syslog traps are not enabled.

Beginning in privileged EXEC mode, follow these steps to change the level and history table size defaults:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**logging history** *level*[1]	Change the default level of syslog messages stored in the history file and sent to the SNMP server. See Table 26-3 on page 26-9 for a list of *level* keywords. By default, **warnings**, **errors**, **critical**, **alerts**, and **emergencies** messages are sent.
Step 3	**logging history size** *number*	Specify the number of syslog messages that can be stored in the history table. The default is to store one message. The range is 1 to 500 messages.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

1. Table 26-3 lists the level keywords and severity level. For SNMP usage, the severity level values increase by 1. For example, emergencies equal 1, not 0, and critical equals 3, not 2.

When the history table is full (it contains the maximum number of message entries specified with the **logging history size** global configuration command), the oldest message entry is deleted from the table to allow the new message entry to be stored.

To return the logging of syslog messages to the default level, use the **no logging history** global configuration command. To return the number of messages in the history table to the default value, use the **no logging history size** global configuration command.

Configuring UNIX Syslog Servers

The next sections describe how to configure the UNIX server syslog daemon and how to define the UNIX system logging facility.

Logging Messages to a UNIX Syslog Daemon

Before you can send system log messages to a UNIX syslog server, you must configure the syslog daemon on a UNIX server. Log in as root, and perform these steps:

Note Some recent versions of UNIX syslog daemons no longer accept by default syslog packets from the network. If this is the case with your system, use the UNIX **man syslogd** command to determine what options must be added to or removed from the syslog command line to enable logging of remote syslog messages.

Step 1 Add a line such as the following to the file /etc/syslog.conf:

```
local7.debug /usr/adm/logs/cisco.log
```

The **local7** keyword specifies the logging facility to be used; see Table 26-4 on page 26-12 for information on the facilities. The **debug** keyword specifies the syslog level; see Table 26-3 on page 26-9 for information on the severity levels. The syslog daemon sends messages at this level or at a more severe level to the file specified in the next field. The file must already exist, and the syslog daemon must have permission to write to it.

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

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

Step 3 Make sure the syslog daemon reads the new changes:

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

For more information, see the **man syslog.conf** and **man syslogd** commands on your UNIX system.

Configuring the UNIX System Logging Facility

When sending system log messages to an external device, you can cause the switch to identify its messages as originating from any of the UNIX syslog facilities.

Beginning in privileged EXEC mode, follow these steps to configure UNIX system facility message logging:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**logging** *host*	Log messages to a UNIX syslog server host by entering its IP address. To build a list of syslog servers that receive logging messages, enter this command more than once.
Step 3	**logging trap** *level*	Limit messages logged to the syslog servers. Be default, syslog servers receive informational messages and lower. See Table 26-3 on page 26-9 for *level* keywords.

	Command	Purpose
Step 4	**logging facility** *facility-type*	Configure the syslog facility. See Table 26-4 on page 26-12 for *facility-type* keywords. The default is **local7**.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a syslog server, use the **no logging** *host* global configuration command, and specify the syslog server IP address. To disable logging to syslog servers, enter the **no logging trap** global configuration command.

Table 26-4 lists the UNIX system facilities supported by the software. For more information about these facilities, consult the operator's manual for your UNIX operating system.

Table 26-4 Logging Facility-Type Keywords

Facility Type Keyword	Description
auth	Authorization system
cron	Cron facility
daemon	System daemon
kern	Kernel
local0-7	Locally defined messages
lpr	Line printer system
mail	Mail system
news	USENET news
sys9	System use
sys10	System use
sys11	System use
sys12	System use
sys13	System use
sys14	System use
syslog	System log
user	User process
uucp	UNIX-to-UNIX copy system

Displaying the Logging Configuration

To display the logging configuration and the contents of the log buffer, use the **show logging** privileged EXEC command. For information about the fields in this display, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

CHAPTER 27

Configuring SNMP

This chapter describes how to configure the Simple Network Management Protocol (SNMP) on your Catalyst 3550 switch.

Note For complete syntax and usage information for the commands used in this chapter, see the switch command reference for this release and to the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

This chapter consists of these sections:

- Understanding SNMP, page 27-1
- Configuring SNMP, page 27-6
- Displaying SNMP Status, page 27-17

Understanding SNMP

SNMP is an application-layer protocol that provides a message format for communication between managers and agents. The SNMP system consists of an SNMP manager, an SNMP agent, and a MIB. The SNMP manager can be part of a network management system (NMS) such as CiscoWorks. The agent and MIB reside on the switch. To configure SNMP on the switch, you define the relationship between the manager and the agent.

The SNMP agent contains MIB variables whose values the SNMP manager can request or change. A manager can get a value from an agent or store a value into the agent. The agent gathers data from the MIB, the repository for information about device parameters and network data. The agent can also respond to a manager's requests to get or set data.

An agent can send unsolicited traps to the manager. Traps are messages alerting the SNMP manager to a condition on the network. Traps can mean improper user authentication, restarts, link status (up or down), MAC address tracking, closing of a TCP connection, loss of connection to a neighbor, or other significant events.

This section includes information about these topics:

- SNMP Versions, page 27-2
- SNMP Manager Functions, page 27-3
- SNMP Agent Functions, page 27-4
- SNMP Community Strings, page 27-4

- Using SNMP to Access MIB Variables, page 27-4
- SNMP Notifications, page 27-5
- SNMP ifIndex MIB Object Values, page 27-5

SNMP Versions

This software release supports these SNMP versions:

- SNMPv1—The Simple Network Management Protocol, a Full Internet Standard, defined in RFC 1157.
- SNMPv2C replaces the Party-based Administrative and Security Framework of SNMPv2Classic with the community-string-based Administrative Framework of SNMPv2C while retaining the bulk retrieval and improved error handling of SNMPv2Classic. It has these features:
  - SNMPv2—Version 2 of the Simple Network Management Protocol, a Draft Internet Standard, defined in RFCs 1902 through 1907.
  - SNMPv2C—The community-string-based Administrative Framework for SNMPv2, an Experimental Internet Protocol defined in RFC 1901.
- SNMPv3—Version 3 of the SNMP is an interoperable standards-based protocol defined in RFCs 2273 to 2275. SNMPv3 provides secure access to devices by authenticating and encrypting packets over the network and includes these security features:
  - Message integrity—ensuring that a packet was not tampered with in transit
  - Authentication—determining that the message is from a valid source
  - Encryption—mixing the contents of a package to prevent it from being read by an unauthorized source.

Note To select encryption, enter the **priv** keyword. This keyword is available only when the cryptographic (encrypted) software image is installed.

Both SNMPv1 and SNMPv2C use a community-based form of security. The community of managers able to access the agent's MIB is defined by an IP address access control list and password.

SNMPv2C includes a bulk retrieval mechanism and more detailed error message reporting to management stations. The bulk retrieval mechanism retrieves tables and large quantities of information, minimizing the number of round-trips required. The SNMPv2C improved error-handling includes expanded error codes that distinguish different kinds of error conditions; these conditions are reported through a single error code in SNMPv1. Error return codes in SNMPv2C report the error type.

SNMPv3 provides for both security models and security levels. A security model is an authentication strategy set up for a user and the group within which the user resides. A security level is the permitted level of security within a security model. A combination of the security level and the security model determine which security mechanism is used when handling an SNMP packet. Available security models are SNMPv1, SNMPv2C, and SNMPv3.

Table 27-1 identifies the characteristics of the different combinations of security models and levels.

Table 27-1 SNMP Security Models and Levels

Model	Level	Authentication	Encryption	Result
SNMPv1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
SNMPv2C	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
SNMPv3	noAuthNoPriv	Username	No	Uses a username match for authentication.
SNMPv3	authNoPriv	MD5 or SHA	No	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms.
SNMPv3	authPriv (requires the cryptographic software image)	MD5 or SHA	DES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides DES 56-bit encryption in addition to authentication based on the CBC-DES (DES-56) standard.

You must configure the SNMP agent to use the SNMP version supported by the management station. Because an agent can communicate with multiple managers, you can configure the software to support communications using SNMPv1, SNMPv2C, or SNMPv3.

SNMP Manager Functions

The SNMP manager uses information in the MIB to perform the operations described in Table 27-2.

Table 27-2 SNMP Operations

Operation	Description
get-request	Retrieves a value from a specific variable.
get-next-request	Retrieves a value from a variable within a table.[1]
get-bulk-request[2]	Retrieves large blocks of data, such as multiple rows in a table, that would otherwise require the transmission of many small blocks of data.
get-response	Replies to a get-request, get-next-request, and set-request sent by an NMS.
set-request	Stores a value in a specific variable.
trap	An unsolicited message sent by an SNMP agent to an SNMP manager when some event has occurred.

1. With this operation, an SNMP manager does not need to know the exact variable name. A sequential search is performed to find the needed variable from within a table.
2. The **get-bulk** command only works with SNMPv2 or later.

SNMP Agent Functions

The SNMP agent responds to SNMP manager requests as follows:

- Get a MIB variable—The SNMP agent begins this function in response to a request from the NMS. The agent retrieves the value of the requested MIB variable and responds to the NMS with that value.
- Set a MIB variable—The SNMP agent begins this function in response to a message from the NMS. The SNMP agent changes the value of the MIB variable to the value requested by the NMS.

The SNMP agent also sends unsolicited trap messages to notify an NMS that a significant event has occurred on the agent. Examples of trap conditions include, but are not limited to, when a port or module goes up or down, when spanning-tree topology changes occur, and when authentication failures occur.

SNMP Community Strings

SNMP community strings authenticate access to MIB objects and function as embedded passwords. In order for the NMS to access the switch, the community string definitions on the NMS must match at least one of the three community string definitions on the switch.

A community string can have one of these attributes:

- Read-only (RO)—Gives read access to authorized management stations to all objects in the MIB except the community strings, but does not allow write access
- Read-write (RW)—Gives read and write access to authorized management stations to all objects in the MIB, but does not allow access to the community strings

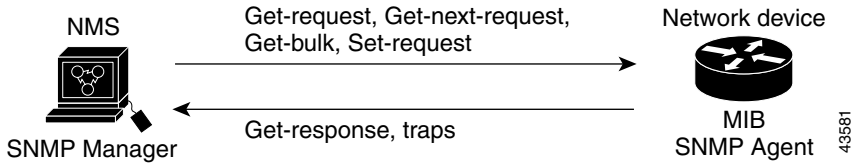
Note When a cluster is created, the command switch manages the exchange of messages among member switches and the SNMP application. The Network Assistant software appends the member switch number (*@esN*, where *N* is the switch number) to the first configured RW and RO community strings on the command switch and propagates them to the member switches. For more information, see Chapter 5, "Clustering Switches" and see the *Getting Started with Cisco Network Assistant*, available on Cisco.com.

Using SNMP to Access MIB Variables

An example of an NMS is the CiscoWorks network management software. CiscoWorks 2000 software uses the switch MIB variables to set device variables and to poll devices on the network for specific information. The results of a poll can be displayed as a graph and analyzed to troubleshoot internetworking problems, increase network performance, verify the configuration of devices, monitor traffic loads, and more.

As shown in Figure 27-1, the SNMP agent gathers data from the MIB. The agent can send traps, or notification of certain events, to the SNMP manager, which receives and processes the traps. Traps alert the SNMP manager to a condition on the network such as improper user authentication, restarts, link status (up or down), MAC address tracking, and so forth. The SNMP agent also responds to MIB-related queries sent by the SNMP manager in *get-request*, *get-next-request*, and *set-request* format.

Figure 27-1 SNMP Network



For information on supported MIBs and how to access them, see Appendix A, "Supported MIBs."

SNMP Notifications

SNMP allows the switch to send notifications to SNMP managers when particular events occur. SNMP notifications can be sent as traps or inform requests. In command syntax, unless there is an option in the command to select either traps or informs, the keyword *traps* refers to either traps or informs, or both. Use the **snmp-server host** command to specify whether to send SNMP notifications as traps or informs.

Note SNMPv1 does not support informs.

Traps are unreliable because the receiver does not send an acknowledgment when it receives a trap, and the sender cannot determine if the trap was received. When an SNMP manager receives an inform request, it acknowledges the message with an SNMP response protocol data unit (PDU). If the sender does not receive a response, the inform request can be sent again. Because they can be re-sent, informs are more likely than traps to reach their intended destination.

The characteristics that make informs more reliable than traps also consume more resources in the switch and in the network. Unlike a trap, which is discarded as soon as it is sent, an inform request is held in memory until a response is received or the request times out. Traps are sent only once, but an inform might be re-sent or retried several times. The retries increase traffic and contribute to a higher overhead on the network. Therefore, traps and informs require a trade-off between reliability and resources. If it is important that the SNMP manager receive every notification, use inform requests. If traffic on the network or memory in the switch is a concern and notification is not required, use traps.

SNMP ifIndex MIB Object Values

In an NMS, the IF-MIB generates and assigns an interface index (ifIndex) object value that is a unique number greater than zero to identify a physical or a logical interface. When the switch reboots or the switch software is upgraded, the switch uses this same value for the interface. For example, if the switch assigns a port 2 an ifIndex value of 10003, this value is the same after the switch reboots.

Use the **snmp-server ifindex persist** global configuration command to enable ifindex persistence on the switch.

Configuring SNMP

This section describes how to configure SNMP on your switch. It contains this configuration information:

- Default SNMP Configuration, page 27-6
- SNMP Configuration Guidelines, page 27-6
- Disabling the SNMP Agent, page 27-7
- Configuring Community Strings, page 27-8
- Configuring SNMP Groups and Users, page 27-9
- Configuring SNMP Notifications, page 27-11
- Configuring SNMP Trap Notification Priority, page 27-14
- Setting the Agent Contact and Location Information, page 27-15
- Limiting TFTP Servers Used Through SNMP, page 27-16
- SNMP Examples, page 27-16

Default SNMP Configuration

Table 27-3 shows the default SNMP configuration.

Table 27-3 Default SNMP Configuration

Feature	Default Setting
SNMP agent	Disabled[1].
SNMP community strings	Read-Only: Public Read-Write: Private
SNMP trap receiver	None configured.
SNMP traps	None enabled except the trap for TCP connections (**tty**).
SNMP version	If no **version** keyword is present, the default is Version 1.
SNMPv3 authentication	If no keyword is entered, the default is the **noauth** (noAuthNoPriv) security level.
SNMP notification type	If no type is specified, all notifications are sent.

1. This is the default when the switch starts and the startup configuration does not have any **snmp-server** global configuration commands.

SNMP Configuration Guidelines

If the switch starts and the switch startup configuration has at least one **snmp-server** global configuration command, the SNMP agent is enabled.

An SNMP *group* is a table that maps SNMP users to SNMP views. An SNMP *user* is a member of an SNMP group. An SNMP *host* is the recipient of an SNMP trap operation. An SNMP *engine ID* is a name for the local or remote SNMP engine.

When configuring SNMP, follow these guidelines:

- When configuring an SNMP group, do not specify a notify view. The **snmp-server host** global configuration command autogenerates a notify view for the user and then adds it to the group associated with that user. Modifying the group's notify view affects all users associated with that group. See the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2* for information about when you should configure notify views.
- To configure a remote user, specify the IP address or port number for the remote SNMP agent of the device where the user resides.
- Before you configure remote users for a particular agent, configure the SNMP engine ID, using the **snmp-server engineID** global configuration with the **remote** option. The remote agent's SNMP engine ID and user password are used to compute the authentication and privacy digests. If you do not configure the remote engine ID first, the configuration command fails.
- When configuring SNMP informs, you need to configure the SNMP engine ID for the remote agent in the SNMP database before you can send proxy requests or informs to it.
- If a local user is not associated with a remote host, the switch does not send informs for the **auth** (authNoPriv) and the **priv** (authPriv) authentication levels.
- Changing the value of the SNMP engine ID has important side effects. A user's password (entered on the command line) is converted to an MD5 or SHA security digest based on the password and the local engine ID. The command-line password is then destroyed, as required by RFC 2274. Because of this deletion, if the value of engineID changes, the security digests of SNMPv3 users become invalid, and you need to reconfigure SNMP users by using the **snmp-server user** *username* global configuration command. Similar restrictions require the reconfiguration of community strings when the engine ID changes.

Disabling the SNMP Agent

Beginning in privileged EXEC mode, follow these steps to disable the SNMP agent:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no snmp-server**	Disable the SNMP agent operation.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The **no snmp-server** global configuration command disables all running versions (Version 1, Version 2C, and Version 3) on the device. No specific Cisco IOS command exists to enable SNMP. The first **snmp-server** global configuration command that you enter enables all versions of SNMP.

Configuring Community Strings

You use the SNMP community string to define the relationship between the SNMP manager and the agent. The community string acts like a password to permit access to the agent on the switch. Optionally, you can specify one or more of these characteristics associated with the string:

- An access list of IP addresses of the SNMP managers that are permitted to use the community string to gain access to the agent
- A MIB view, which defines the subset of all MIB objects accessible to the given community
- Read and write or read-only permission for the MIB objects accessible to the community

Beginning in privileged EXEC mode, follow these steps to configure a community string on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [*access-list-number*]	Configure the community string. **Note** The @ symbol is used for delimiting the context information. Avoid using the @ symbol as part of the SNMP community string when configuring this command. • For *string*, specify a string that acts like a password and permits access to the SNMP protocol. You can configure one or more community strings of any length. • (Optional) For **view**, specify the view record accessible to the community. • (Optional) Specify either read-only (**ro**) if you want authorized management stations to retrieve MIB objects, or specify read-write (**rw**) if you want authorized management stations to retrieve and modify MIB objects. By default, the community string permits read-only access to all objects. • (Optional) For *access-list-number*, enter an IP standard access list numbered from 1 to 99 and 1300 to 1999.

	Command	Purpose
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	(Optional) If you specified an IP standard access list number in Step 2, then create the list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the IP address of the SNMP managers that are permitted to use the community string to gain access to the agent. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note To disable access for an SNMP community, set the community string for that community to the null string (do not enter a value for the community string).

To remove a specific community string, use the **no snmp-server community** *string* global configuration command.

This example shows how to assign the string *comaccess* to SNMP, to allow read-only access, and to specify that IP access list 4 can use the community string to gain access to the switch SNMP agent:

```
Switch(config)# snmp-server community comaccess ro 4
```

Configuring SNMP Groups and Users

You can specify an identification name (engineID) for the local or remote SNMP server engine on the switch. You can configure an SNMP server group that maps SNMP users to SNMP views, and you can add new users to the SNMP group.

Beginning in privileged EXEC mode, follow these steps to configure SNMP on the switch:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**snmp-server engineID** {**local** *engineid-string* | **remote** *ip-address* [**udp-port** *port-number*] *engineid-string*}	Configure a name for either the local or remote copy of SNMP. • The *engineid-string* is a 24-character ID string with the name of the copy of SNMP. You need not specify the entire 24-character engine ID if it contains trailing zeros. Specify only the portion of the engine ID up to the point where only zeros remain in the value. For example, to configure an engine ID of 123400000000000000000000, you can enter this: **snmp-server engineID local 1234** • If you select **remote**, specify the *ip-address* of the device that contains the remote copy of SNMP and the optional User Datagram Protocol (UDP) port on the remote device. The default is 162.
Step 3	**snmp-server group** *groupname* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]	Configure a new SNMP group on the remote device. • For *groupname,* specify the name of the group. • Specify a security model: – **v1** is the least secure of the possible security models. – **v2c** is the second least secure model. It allows transmission of informs and integers twice the normal width. – **v3,** the most secure, requires you to select an authentication level: **auth**—Enables the Message Digest 5 (MD5) and the Secure Hash Algorithm (SHA) packet authentication. **noauth**—Enables the noAuthNoPriv security level. This is the default if no keyword is specified. **priv**—Enables Data Encryption Standard (DES) packet encryption (also called *privacy*). **Note** The **priv** keyword is available only when the cryptographic software image is installed. • (Optional) Enter **read** *readview* with a string (not to exceed 64 characters) that is the name of the view in which you can only view the contents of the agent. • (Optional) Enter **write** *writeview* with a string (not to exceed 64 characters) that is the name of the view in which you enter data and configure the contents of the agent. • (Optional) Enter **notify** *notifyview* with a string (not to exceed 64 characters) that is the name of the view in which you specify a notify, inform, or trap. • (Optional) Enter **access** *access-list* with a string (not to exceed 64 characters) that is the name of the access list.

	Command	Purpose
Step 4	**snmp-server user** *username groupname* {**remote** *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*]}	Add a new user for an SNMP group. • The *username* is the name of the user on the host that connects to the agent. • The *groupname* is the name of the group to which the user is associated. • Enter **remote** to specify a remote SNMP entity to which the user belongs and the hostname or IP address of that entity with the optional UDP port number. The default is 162. • Enter the SNMP version number (**v1**, **v2c**, or **v3**). If you enter **v3**, you have these additional options: – **encrypted** specifies that the password appears in encrypted format. This keyword is available only when the **v3** keyword is specified. – **auth** is an authentication level setting session that can be either the HMAC-MD5-96 (**md5**) or the HMAC-SHA-96 (**sha**) authentication level, and requires a password string (not to exceed 64 characters). • (Optional) Enter **access** *access-list* with a string (not to exceed 64 characters) that is the name of the access list.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring SNMP Notifications

A trap manager is a management station that receives and processes traps. Traps are system alerts that the switch generates when certain events occur. By default, no trap manager is defined, and no traps are sent. Switches running this Cisco IOS release can have an unlimited number of trap managers.

Note Many commands use the word *traps* in the command syntax. Unless there is an option in the command to select either traps or informs, the keyword *traps* refers to either traps, informs, or both. Use the **snmp-server host** global configuration command to specify whether to send SNMP notifications as traps or informs.

Table 27-4 describes the supported switch traps (notification types). You can enable any or all of these traps and configure a trap manager to receive them.

Table 27-4 Switch Notification Types

Notification Type Keyword	Description
bgp	Generates Border Gateway Protocol (BGP) state change traps. This option is only available when the enhanced multilayer image is installed.
bridge	Generates STP bridge MIB traps.

Table 27-4 Switch Notification Types (continued)

Notification Type Keyword	Description
cluster	Generates a trap when the cluster configuration changes.
config	Generates a trap for SNMP configuration changes.
copy-config	Generates a trap for SNMP copy configuration changes.
entity	Generates a trap for SNMP entity changes.
envmon	Generates environmental monitor traps. You can enable any or all of these environmental traps: fan, shutdown, status, supply, temperature.
flash	Generates SNMP FLASH notifications.
hsrp	Generates a trap for Hot Standby Router Protocol (HSRP) changes.
ipmulticast	Generates a trap for IP multicast routing changes.
mac-notification	Generates a trap for MAC address notifications.
msdp	Generates a trap for Multicast Source Discovery Protocol (MSDP) changes.
ospf	Generates a trap Open Shortest Path First (OSPF) changes. You can enable any or all of these traps: Cisco specific, errors, link-state advertisement, rate limit, retransmit, and state changes.
pim	Generates a trap for Protocol-Independent Multicast (PIM) changes. You can enable any or all of these traps: invalid PIM messages, neighbor changes, and rendezvous point (RP)-mapping changes.
port-security	Generates SNMP port security traps. You can also set a maximum trap rate per second. The range is from 0 to 1000; the default is 0, which means that there is no rate limit. **Note** When you configure a trap by using the notification type **port-security**, configure the port security trap first, and then configure the port security trap rate: • **snmp-server enable traps port-security** • **snmp-server enable traps port-security trap-rate** *rate*
rtr	Generates a trap for the SNMP Response Time Reporter (RTR).
snmp	Generates a trap for SNMP-type notifications.
storm-control	Generates a trap for SNMP storm-control. You can also set a maximum trap rate per minute. The range is from 0 to 1000; the default is 0 (no limit is imposed; a trap is sent at every occurrence).
stpx	Generates SNMP STP Extended MIB traps.
syslog	Generates SNMP syslog traps.
tty	Generates a trap for TCP connections. This trap is enabled by default.
vlancreate	Generates SNMP VLAN-created traps.
vlandelete	Generates SNMP VLAN-deleted traps.
vlan-membership	Generates a trap for SNMP VLAN membership changes.
vtp	Generates a trap for VLAN Trunking Protocol (VTP) changes.

You can use the **snmp-server host** global configuration command to a specific host to receive the notification types listed in Table 27-4.

Beginning in privileged EXEC mode, follow these steps to configure the switch to send traps or informs to a host:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**snmp-server engineID remote** *ip-address engineid-string*	Specify the engine ID for the remote host.
Step 3	**snmp-server user** *username groupname* {**remote** *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*]}	Configure an SNMP user to be associated with the remote host created in Step 2. **Note** You cannot configure a remote user for an address without first configuring the engine ID for the remote host. Otherwise, you receive an error message, and the command is not executed.
Step 4	**snmp-server group** [*groupname* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}}] [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]	Configure an SNMP group.
Step 5	**snmp-server host** *host-addr* [**informs** | **traps**] [**version** {**1** | **2c** | **3** {**auth** | **noauth** | **priv**}}] *community-string* [*notification-type*]	Specify the recipient of an SNMP trap operation. • For *host-addr,* specify the name or Internet address of the host (the targeted recipient). • (Optional) Enter **informs** to send SNMP informs to the host. • (Optional) Enter **traps** (the default) to send SNMP traps to the host. • (Optional) Specify the SNMP **version** (**1**, **2c**, or **3**). SNMPv1 is not available with informs. • (Optional) For Version 3, select authentication level **auth, noauth**, or **priv**. **Note** The **priv** keyword is available only when the cryptographic software image is installed. • For *community-string*, when **version 1** or **version 2c** is specified, enter the password-like community string sent with the notification operation. When **version 3** is specified, enter the SNMPv3 username. **Note** The @ symbol is used for delimiting the context information. Avoid using the @ symbol as part of the SNMP community string when configuring this command. • (Optional) For *notification-type*, use the keywords listed in Table 27-4 on page 27-11. If no type is specified, all notifications are sent.

	Command	Purpose
Step 6	**snmp-server enable traps** *notification-types*	Enable the switch to send traps or informs and specify the type of notifications to be sent. For a list of notification types, see Table 27-4 on page 27-11, or enter this: **snmp-server enable traps ?** To enable multiple types of traps, you must enter a separate **snmp-server enable traps** command for each trap type. **Note** When you configure a trap by using the notification type **port-security**, configure the port security trap first, and then configure the port security trap rate: • **snmp-server enable traps port-security** • **snmp-server enable traps port-security trap-rate** *rate*
Step 7	**snmp-server trap-source** *interface-id*	(Optional) Specify the source interface, which provides the IP address for the trap message. This command also sets the source IP address for informs.
Step 8	**snmp-server queue-length** *length*	(Optional) Establish the message queue length for each trap host. The range is 1 to 1000; the default is 10.
Step 9	**snmp-server trap-timeout** *seconds*	(Optional) Define how often to resend trap messages. The range is 1 to 1000; the default is 30 seconds.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**show running-config**	Verify your entries.
Step 12	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

The **snmp-server host** command specifies which hosts receive the notifications. The **snmp-server enable trap** command globally enables the mechanism for the specified notification (for traps and informs). To enable a host to receive an inform, you must configure an **snmp-server host informs** command for the host and globally enable informs by using the **snmp-server enable traps** command.

To remove the specified host from receiving traps, use the **no snmp-server host** *host* global configuration command. The **no snmp-server host** command with no keywords disables traps, but not informs, to the host. To disable informs, use the **no snmp-server host informs** global configuration command. To disable a specific trap type, use the **no snmp-server enable traps** *notification-types* global configuration command.

Configuring SNMP Trap Notification Priority

You can prioritize outgoing SNMP trap notifications to move them more efficiently through the network, even during periods of congestion. The switch has these prioritization options for SNMP packets:

- IP precedence marker
- Differentiated Services Code Point (DSCP) marker

These markers specify the preference that SNMP packets should receive as they move through the network. You can set up to 8 different IP precedence markings or 64 different IP DSCP markings. The default IP precedence and DSCP marker, 0, forwards SNMP packets as normal traffic. The highest marker values, 7 for IP precedence and 63 for DSCP, are generally reserved for network control traffic. Choose a marker value that corresponds to the importance of SNMP notifications in your network. For example, set the IP precedence to 6 to assign a very high priority to outgoing SNMP notifications.

DSCP is partially backward-compatible with IP precedence. To choose DSCP values that work like IP precedence values, use these values: 0, 8, 16, 24, 32, 40, 48, and 56. Although DSCP has 64 possible values, the network could disregard the least significant bits or treat blocks of values the same.

Beginning in privileged EXEC mode, follow these steps to configure the switch to prioritize outgoing SNMP trap notifications:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**snmp-server ip** {**precedence** *precedence-value* | **dscp** *dscp-value*}	Specify the IP precedence or the DSCP marker value for SNMP notifications.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To specify the host that should receive SNMP traps, use the **snmp-server host** global configuration command. To enable specific trap types, use the **snmp-server enable traps** global configuration command.

Setting the Agent Contact and Location Information

Beginning in privileged EXEC mode, follow these steps to set the system contact and location of the SNMP agent so that these descriptions can be accessed through the configuration file:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**snmp-server contact** *text*	Set the system contact string. For example: `snmp-server contact Dial System Operator at beeper 21555.`
Step 3	**snmp-server location** *text*	Set the system location string. For example: `snmp-server location Building 3/Room 222`
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Limiting TFTP Servers Used Through SNMP

Beginning in privileged EXEC mode, follow these steps to limit the TFTP servers used for saving and loading configuration files through SNMP to the servers specified in an access list:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**snmp-server tftp-server-list** *access-list-number*	Limit TFTP servers used for configuration file copies through SNMP to the servers in the access list. For *access-list-number*, enter an IP standard access list numbered from 1 to 99 and 1300 to 1999.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the IP address of the TFTP servers that can access the switch. • (Optional) For *source-wildcard*, enter the wildcard bits, in dotted decimal notation, to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

SNMP Examples

This example shows how to enable all versions of SNMP. The configuration permits any SNMP manager to access all objects with read-only permissions using the community string *public*. This configuration does not cause the switch to send any traps.

```
Switch(config)# snmp-server community public
```

This example shows how to permit any SNMP manager to access all objects with read-only permission using the community string *public*. The switch also sends VTP traps to the hosts 192.180.1.111 and 192.180.1.33 using SNMPv1 and to the host 192.180.1.27 using SNMPv2C. The community string *public* is sent with the traps.

```
Switch(config)# snmp-server community public
Switch(config)# snmp-server enable traps vtp
Switch(config)# snmp-server host 192.180.1.27 version 2c public
Switch(config)# snmp-server host 192.180.1.111 version 1 public
Switch(config)# snmp-server host 192.180.1.33 public
```

This example shows how to allow read-only access for all objects to members of access list 4 that use the *comaccess* community string. No other SNMP managers have access to any objects. SNMP Authentication Failure traps are sent by SNMPv2C to the host *cisco.com* using the community string *public*.

```
Switch(config)# snmp-server community comaccess ro 4
Switch(config)# snmp-server enable traps snmp authentication
Switch(config)# snmp-server host cisco.com version 2c public
```

This example shows how to send Entity MIB traps to the host *cisco.com*. The community string is restricted. The first line enables the switch to send Entity MIB traps in addition to any traps previously enabled. The second line specifies the destination of these traps and overwrites any previous **snmp-server host** commands for the host *cisco.com*.

```
Switch(config)# snmp-server enable traps entity
Switch(config)# snmp-server host cisco.com restricted entity
```

This example shows how to enable the switch to send all traps to the host *myhost.cisco.com* using the community string *public*:

```
Switch(config)# snmp-server enable traps
Switch(config)# snmp-server host myhost.cisco.com public
```

This example shows how to associate a user with a remote host and to send **auth** (authNoPriv) authentication-level informs when the user enters global configuration mode:

```
Switch(config)# snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b
Switch(config)# snmp-server group authgroup v3 auth
Switch(config)# snmp-server user authuser authgroup remote 192.180.1.27 v3 auth md5
mypassword
Switch(config)# snmp-server user authuser authgroup v3 auth md5 mypassword
Switch(config)# snmp-server host 192.180.1.27 informs version 3 auth authuser config
Switch(config)# snmp-server enable traps
Switch(config)# snmp-server inform retries 0
```

Displaying SNMP Status

To display SNMP input and output statistics, including the number of illegal community string entries, errors, and requested variables, use the **show snmp** privileged EXEC command. You also can use the other privileged EXEC commands in Table 27-5 to display SNMP information. For information about the fields in the displays, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

Table 27-5 Commands for Displaying SNMP Information

Feature	Default Setting
show snmp	Displays SNMP statistics.
show snmp engineID [**local** | **remote**]	Displays information on the local SNMP engine and all remote engines that have been configured on the device.
show snmp group	Displays information on each SNMP group on the network.
show snmp pending	Displays information on pending SNMP requests.
show snmp sessions	Displays information on the current SNMP sessions.
show snmp user	Displays information on each SNMP user name in the SNMP users table.

CHAPTER 28

Configuring QoS

This chapter describes how to configure quality of service (QoS) by using automatic QoS (auto-QoS) commands or by using standard QoS commands. With QoS, you can give preferential treatment to certain types of traffic at the expense of others. Without QoS, the Catalyst 3550 switch offers best-effort service to each packet, regardless of the packet contents or size. It sends the packets without any assurance of reliability, delay bounds, or throughput.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding QoS, page 28-2
- Configuring Auto-QoS, page 28-17
- Displaying Auto-QoS Information, page 28-23
- Auto-QoS Configuration Example, page 28-23
- Configuring Standard QoS, page 28-25
- Displaying Standard QoS Information, page 28-69
- Standard QoS Configuration Examples, page 28-70

Note When you are configuring QoS parameters for the switch, in order to allocate system resources to maximize the number of possible QoS access control entries (ACEs) allowed, you can use the **sdm prefer access** global configuration command to set the Switch Database Management feature to the access template. For more information on the SDM templates, see the "Optimizing System Resources for User-Selected Features" section on page 6-26.

The switch supports some of the modular QoS CLI (MQC) commands. For more information about the MQC commands, see the "Modular Quality of Service Command Line Interface Overview" at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fqos_c/fqcprt8/qcfmdcli.htm#89799

Understanding QoS

Typically, networks operate on a best-effort delivery basis, which means that all traffic has equal priority and an equal chance of being delivered in a timely manner. When congestion occurs, all traffic has an equal chance of being dropped.

When you configure QoS, you can select specific network traffic, prioritize it according to its relative importance, and use congestion-management and congestion-avoidance techniques to give preferential treatment. Implementing QoS in your network makes network performance more predictable and bandwidth utilization more effective.

The QoS implementation is based on the DiffServ architecture, an emerging standard from the Internet Engineering Task Force (IETF). This architecture specifies that each packet is classified upon entry into the network. The classification is carried in the IP packet header, using 6 bits from the deprecated IP type of service (ToS) field to carry the classification (*class*) information. Classification can also be carried in the Layer 2 frame. These special bits in the Layer 2 frame or in the Layer 3 packet are described here and shown in Figure 28-1:

- Prioritization bits in Layer 2 frames:

  Layer 2 Inter-Switch Link (ISL) frame headers have a 1-byte User field that carries an IEEE 802.1p class of service (CoS) value in the three least-significant bits. On interfaces configured as Layer 2 ISL trunks, all traffic is in ISL frames.

  Layer 2 IEEE 802.1Q frame headers have a 2-byte Tag Control Information field that carries the CoS value in the three most-significant bits, which are called the User Priority bits. On interfaces configured as Layer 2 IEEE 802.1Q trunks, all traffic is in IEEE 802.1Q frames except for traffic in the native VLAN.

  Other frame types cannot carry Layer 2 CoS values.

  Layer 2 CoS values range from 0 for low priority to 7 for high priority.

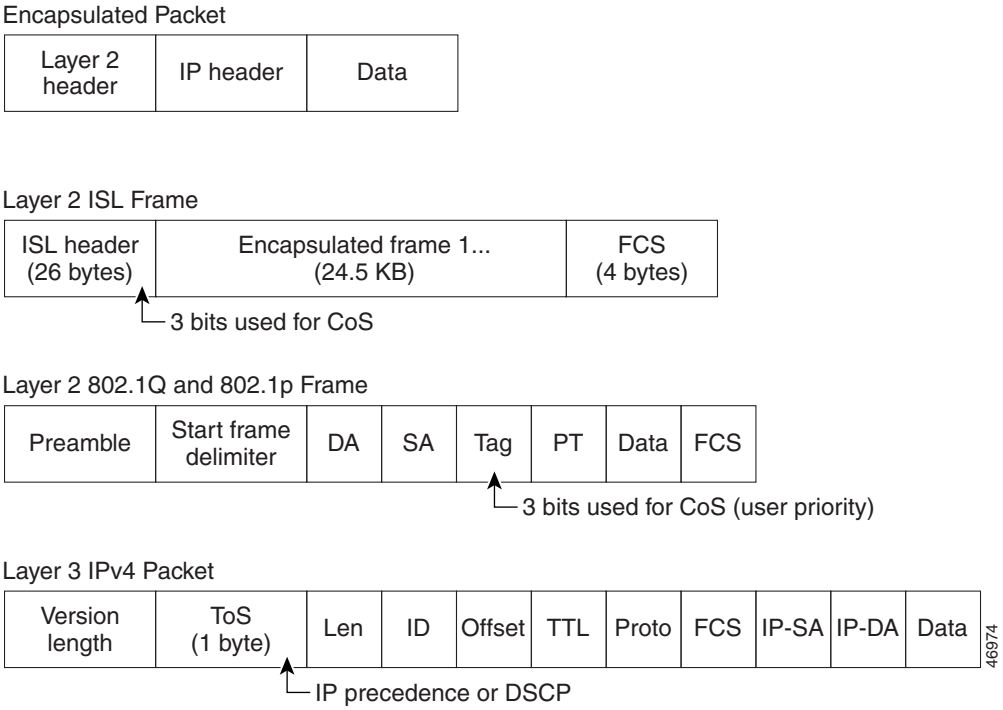
- Prioritization bits in Layer 3 packets:

  Layer 3 IP packets can carry either an IP precedence value or a Differentiated Services Code Point (DSCP) value. QoS supports the use of either value because DSCP values are backward-compatible with IP precedence values.

  IP precedence values range from 0 to 7.

  DSCP values range from 0 to 63.

Figure 28-1 QoS Classification Bits in Frames and Packets



Note Layer 3 IPv6 packets are treated as non-IP packets and are bridged by the switch.

To give the same forwarding treatment to packets with the same class information and different treatment to packets with different class information, all switches and routers that access the Internet rely on class information. Class information in the packet can be assigned by end hosts or by switches or routers along the way, based on a configured policy, detailed examination of the packet, or both. Detailed examination of the packet is expected to happen closer to the network edge so that core switches and routers are not overloaded.

Switches and routers along the path can use class information to limit the amount of resources allocated per traffic class. The behavior of an individual device when handling traffic in the DiffServ architecture is called per-hop behavior. If all devices along a path have a consistent per-hop behavior, you can construct an end-to-end QoS solution.

Implementing QoS in your network can be a simple or complex task and depends on the QoS features offered by your internetworking devices, the traffic types and patterns in your network, and the granularity of control that you need over incoming and outgoing traffic.

These sections describe the QoS stages and how they work:

- Basic QoS Model, page 28-4
- Classification, page 28-5
- Policing and Marking, page 28-8
- Mapping Tables, page 28-10
- Queueing and Scheduling, page 28-11
- Packet Modification, page 28-16

Basic QoS Model

Figure 28-2 shows the basic QoS model. Actions at the ingress interface include classifying traffic, policing, and marking:

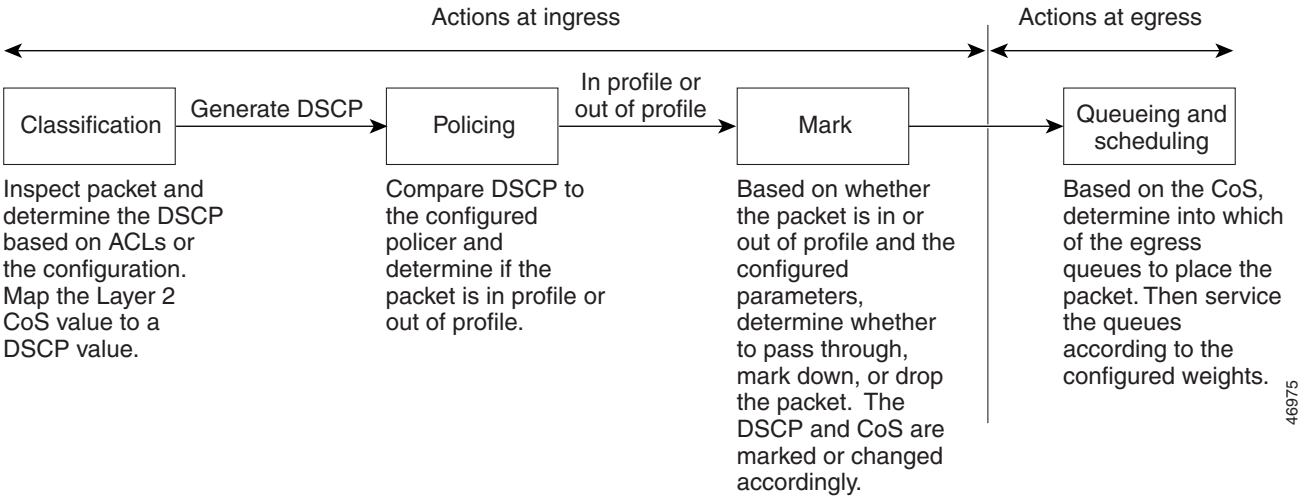
- Classifying distinguishes one kind of traffic from another. The process generates an internal DSCP for a packet, which identifies all the future QoS actions to be performed on this packet. For more information, see the "Classification" section on page 28-5.
- Policing determines whether a packet is in or out of profile by comparing the internal DSCP to the configured policer. The policer limits the bandwidth consumed by a flow of traffic. The result of this determination is passed to the marker. For more information, see the "Policing and Marking" section on page 28-8.
- Marking evaluates the policer and the configuration information for the action to be taken when a packet is out of profile and decides what to do with the packet (pass through a packet without modification, mark down the DSCP value in the packet, or drop the packet). For more information, see the "Policing and Marking" section on page 28-8.

Actions at the egress interface include queueing and scheduling:

- Queueing evaluates the internal DSCP and determines which of the four egress queues in which to place the packet. The DSCP value is mapped to a CoS value, which selects one of the queues. For more information, see the "Mapping Tables" section on page 28-10.
- Scheduling services the four egress queues based on their configured weighted round robin (WRR) weights and thresholds. One of the queues can be the expedite queue, which is serviced until empty before the other queues are serviced. Congestion avoidance techniques include tail drop and Weighted Random Early Detection (WRED) on Gigabit-capable Ethernet ports and tail drop (with only one threshold) on 10/100 Ethernet ports. For more information, see the "Queueing and Scheduling" section on page 28-11.

Note Policing and marking also can occur on egress interfaces.

Figure 28-2 Basic QoS Model



Classification

Classification is the process of distinguishing one kind of traffic from another by examining the fields in the packet. Classification is enabled only if QoS is globally enabled on the switch. By default, QoS is globally disabled, so no classification occurs.

Note Classification occurs on a physical interface or on a per-port per-VLAN basis. No support exists for classifying packets at the switch virtual interface level.

You specify which fields in the frame or packet that you want to use to classify incoming traffic.

For non-IP traffic, these are the classification options as shown in Figure 28-3:

- Use the port default. If the frame does not contain a CoS value, the switch assigns the default port CoS value to the incoming frame. Then, the switch uses the configurable CoS-to-DSCP map to generate the internal DSCP value. The switch uses the internal DSCP value to generate a CoS value representing the priority of the traffic.
- Trust the CoS value in the incoming frame (configure the port to trust CoS). Then, the switch uses the configurable CoS-to-DSCP map to generate the internal DSCP value. Layer 2 ISL frame headers carry the CoS value in the three least-significant bits of the 1-byte User field. Layer 2 IEEE 802.1Q frame headers carry the CoS value in the three most-significant bits of the Tag Control Information field. CoS values range from 0 for low priority to 7 for high priority.
- The trust DSCP and trust IP precedence configurations are meaningless for non-IP traffic. If you configure a port with either of these options and non-IP traffic is received, the switch assigns the default port CoS value and generates the internal DSCP from the CoS-to-DSCP map.
- Perform the classification based on the configured Layer 2 MAC access control list (ACL), which can examine the MAC source address, the MAC destination address, and the Ethertype field. If no ACL is configured, the packet is assigned the default DSCP of 0, which means best-effort traffic; otherwise, the policy map specifies the DSCP to assign to the incoming frame.

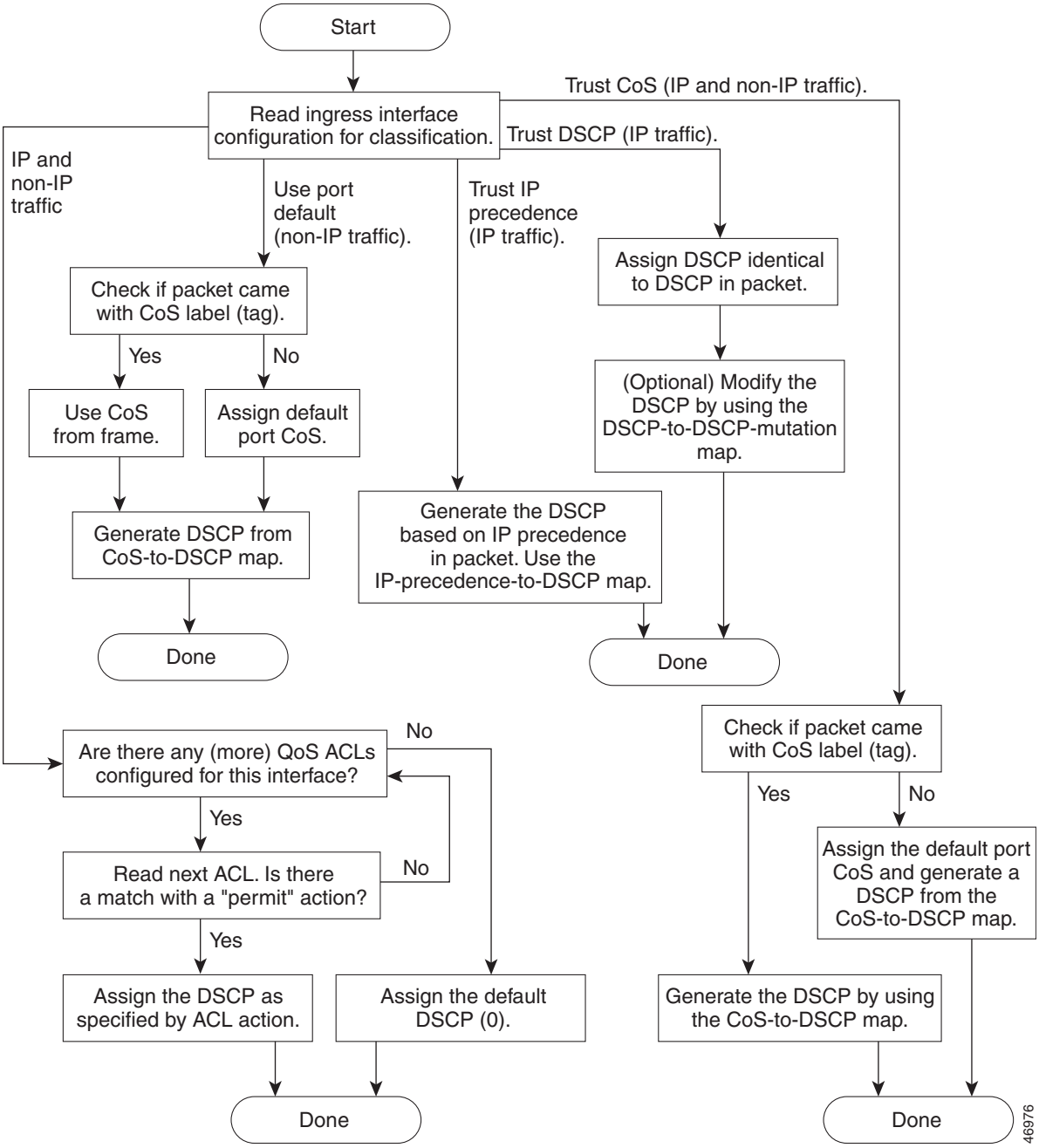
For IP traffic, these are the classification options as shown in Figure 28-3:

- Trust the IP DSCP in the incoming packet (configure the port to trust DSCP), and assign the same DSCP to the packet for internal use. The IETF defines the 6 most-significant bits of the 1-byte ToS field as the DSCP. The priority represented by a particular DSCP value is configurable. DSCP values range from 0 to 63.

  For ports that are on the boundary between two QoS administrative domains, you can modify the DSCP to another value by using the configurable DSCP-to-DSCP-mutation map.
- Trust the IP precedence in the incoming packet (configure the port to trust IP precedence), and generate a DSCP by using the configurable IP-precedence-to-DSCP map. The IP Version 4 specification defines the three most-significant bits of the 1-byte ToS field as the IP precedence. IP precedence values range from 0 for low priority to 7 for high priority.
- Trust the CoS value (if present) in the incoming packet, and generate the DSCP by using the CoS-to-DSCP map.
- Perform the classification based on a configured IP standard or an extended ACL, which examines various fields in the IP header. If no ACL is configured, the packet is assigned the default DSCP of 0, which means best-effort traffic; otherwise, the policy map specifies the DSCP to assign to the incoming frame.

For information on the maps described in this section, see the "Mapping Tables" section on page 28-10. For configuration information on port trust states, see the "Configuring Classification By Using Port Trust States" section on page 28-30.

Figure 28-3 Classification Flowchart



Classification Based on QoS ACLs

You can use IP standard, IP extended, and Layer 2 MAC ACLs to define a group of packets with the same characteristics (*class*). In the QoS context, the permit and deny actions in the access control entries (ACEs) have different meanings than with security ACLs:

- If a match with a permit action is encountered (first-match principle), the specified QoS-related action is taken.
- If a match with a deny action is encountered, the ACL being processed is skipped, and the next ACL is processed.
- If no match with a permit action is encountered and all the ACEs have been examined, no QoS processing occurs on the packet, and the switch offers best-effort service to the packet.
- If multiple ACLs are configured on an interface, the lookup stops after the packet matches the first ACL with a permit action, and QoS processing begins.

Note When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.

After a traffic class has been defined with the ACL, you can attach a policy to it. A policy might contain multiple classes with actions specified for each one of them. A policy might include commands to classify the class as a particular aggregate (for example, assign a DSCP) or rate-limit the class. This policy is then attached to a particular port on which it becomes effective.

You implement IP ACLs to classify IP traffic by using the **access-list** global configuration command; you implement Layer 2 MAC ACLs to classify non-IP traffic by using the **mac access-list extended** global configuration command. For configuration information, see the "Configuring a QoS Policy" section on page 28-36.

Classification Based on Class Maps and Policy Maps

A class map is a mechanism that you use to name and to isolate a specific traffic flow (or class) from all other traffic. The class map defines the criteria used to match against a specific traffic flow to further classify it; the criteria can include matching the access group defined by the ACL, matching a specific list of DSCP or IP precedence values, or matching a specific list of VLAN IDs associated with another class map that defines the actual criteria (for example, to match a standard or extended ACL). If you have more than one type of traffic that you want to classify, you can create another class map and use a different name. After a packet is matched against the class-map criteria, you further classify it through the use of a policy map.

A policy map specifies which traffic class to act on. Actions can include trusting the CoS, DSCP, or IP precedence values in the traffic class; setting a specific DSCP or IP precedence value in the traffic class; or specifying the traffic bandwidth limitations and the action to take when the traffic is out of profile. Before a policy map can be effective, you must attach it to an interface.

You create a class map by using the **class-map** global configuration command or the **class** policy-map configuration command; you should use the **class-map** command when the map is shared among many ports. When you enter the **class-map** command, the switch enters the class-map configuration mode. In this mode, you define the match criterion for the traffic by using the **match** class-map configuration command.

You create and name a policy map by using the **policy-map** global configuration command. When you enter this command, the switch enters the policy-map configuration mode. In this mode, you specify the actions to take on a specific traffic class by using the **class**, **trust**, or **set** policy-map configuration and policy-map class configuration commands. To make the policy map effective, you attach it to an interface by using the **service-policy** interface configuration command.

The policy map also can contain commands that define the policer, the bandwidth limitations of the traffic, and the action to take if the limits are exceeded. For more information, see the "Policing and Marking" section on page 28-8.

A policy map has these characteristics:

- A policy map can contain multiple class statements.
- A separate policy-map class can exist for each type of traffic received through an interface.
- The policy-map trust state and an interface trust state are mutually exclusive, and whichever is configured last takes affect.

For configuration information, see the "Configuring a QoS Policy" section on page 28-36.

Policing and Marking

After a packet is classified and has an internal DSCP value assigned to it, the policing and marking process can begin as shown in Figure 28-4.

Policing involves creating a policer that specifies the bandwidth limits for the traffic. Packets that exceed the limits are *out of profile* or *nonconforming*. Each policer specifies the action to take for packets that are in or out of profile. These actions, carried out by the marker, include passing through the packet without modification, dropping the packet, or marking down the packet with a new DSCP value that is obtained from the configurable policed-DSCP map. For information on the policed-DSCP map, see the "Mapping Tables" section on page 28-10.

You can create these types of policers:

- Individual

  QoS applies the bandwidth limits specified in the policer separately to each matched traffic class. You configure this type of policer within a policy map by using the **police** policy-map configuration command.

- Aggregate

  QoS applies the bandwidth limits specified in an aggregate policer cumulatively to all matched traffic flows. You configure this type of policer by specifying the aggregate policer name within a policy map by using the **police aggregate** policy-map configuration command. You specify the bandwidth limits of the policer by using the **mls qos aggregate-policer** global configuration command. In this way, the aggregate policer is shared by multiple classes of traffic within a policy map.

Policing uses a token bucket algorithm. As each frame is received by the switch, a token is added to the bucket. The bucket has a hole in it and leaks at a rate that you specify as the average traffic rate in bits per second. Each time a token is added to the bucket, the switch performs a check to determine if there is enough room in the bucket. If there is not enough room, the packet is marked as nonconforming, and the specified policer action is taken (dropped or marked down).

How quickly the bucket fills is a function of the bucket depth (burst-byte), the rate at which the tokens are removed (rate-bps), and the duration of the burst above the average rate. The size of the bucket imposes an upper limit on the burst length and determines the number of frames that can be sent

back-to-back. If the burst is short, the bucket does not overflow, and no action is taken against the traffic flow. However, if a burst is long and at a higher rate, the bucket overflows, and the policing actions are taken against the frames in that burst.

You configure the bucket depth (the maximum burst that is tolerated before the bucket overflows) by using the *burst-byte* option of the **police** policy-map class configuration command or the **mls qos aggregate-policer** global configuration command. You configure how fast (the average rate) that the tokens are removed from the bucket by using the *rate-bps* option of the **police** policy-map class configuration command or the **mls qos aggregate-policer** global configuration command.

When configuring policing and policers, keep these items in mind:

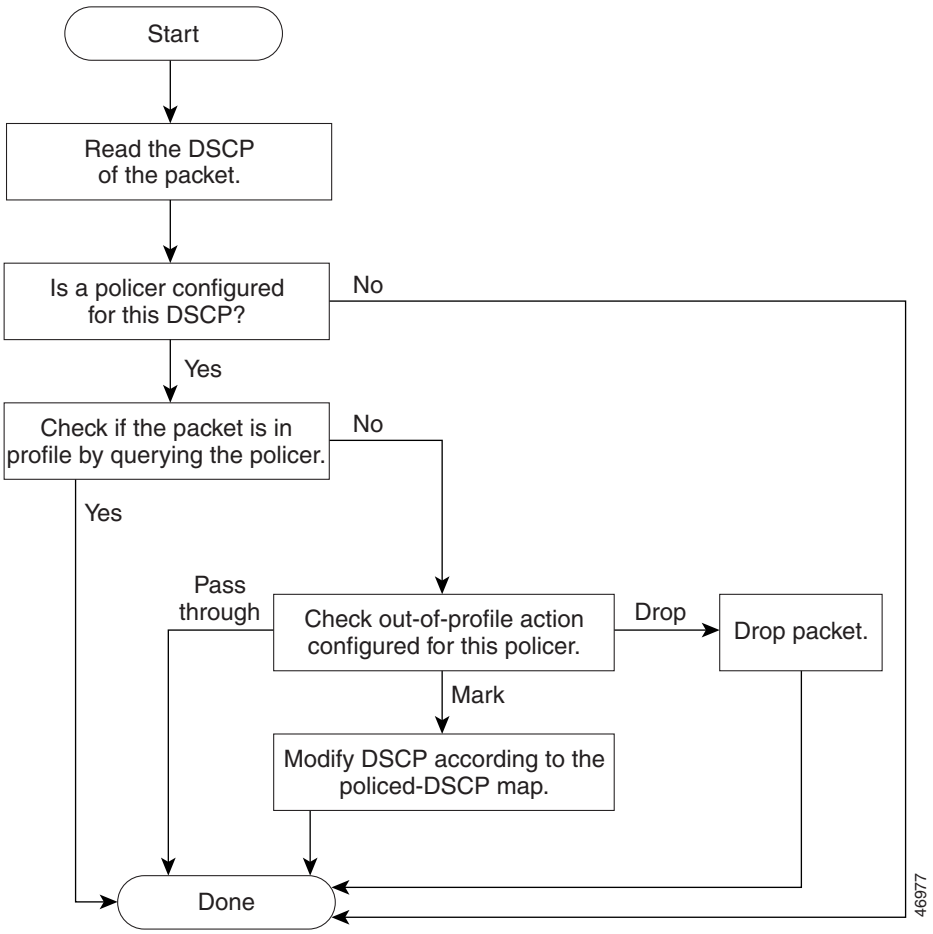
- By default, no policers are configured.
- Policers can be configured only on a physical port or on a per-port per-VLAN basis (specifies the bandwidth limits for the traffic on a per-VLAN basis, for a given port). Per-port per-VLAN policing is not supported on routed ports or on virtual (logical) interfaces. It is supported only on an ingress port configured as a trunk or as a static-access port.
- Only one policer can be applied to a packet per direction.
- Only the average rate and committed burst parameters are configurable.
- Policing can occur on ingress and egress interfaces:

  Note Per-port per-VLAN policing is supported only on ingress interfaces.

  - 128 policers are supported on ingress Gigabit-capable Ethernet ports.
  - 8 policers are supported on ingress 10/100 Ethernet ports.
  - 8 policers are supported on all egress ports.
  - Ingress policers can be individual or aggregate.
- On an interface configured for QoS, all traffic received through the interface is classified, policed, and marked according to the policy map attached to the interface. On a trunk interface configured for QoS, traffic in *all* VLANs received through the interface is classified, policed, and marked according to the policy map attached to the interface.

After you configure the policy map and policing actions, attach the policy to an ingress or egress interface by using the **service-policy** interface configuration command. For configuration information, see the "Classifying, Policing, and Marking Traffic by Using Policy Maps" section on page 28-44 and the "Classifying, Policing, and Marking Traffic by Using Aggregate Policers" section on page 28-50.

Figure 28-4 Policing and Marking Flowchart



Mapping Tables

During QoS processing, the switch represents the priority of all traffic (including non-IP traffic) with an internal DSCP value:

- During classification, QoS uses configurable mapping tables to derive the internal DSCP (a 6-bit value) from received CoS or IP precedence (3-bit) values. These maps include the CoS-to-DSCP map and the IP-precedence-to-DSCP map.

  On an ingress interface configured in the DSCP-trusted state, if the DSCP values are different between the QoS domains, you can apply the configurable DSCP-to-DSCP-mutation map to the interface that is on the boundary between the two QoS domains.

- During policing, QoS can assign another DSCP value to an IP or non-IP packet (if the packet is out of profile and the policer specifies a marked down DSCP value). This configurable map is called the policed-DSCP map.
- Before the traffic reaches the scheduling stage, QoS uses the configurable DSCP-to-CoS map to derive a CoS value from the internal DSCP value. Through the CoS-to-egress-queue map, the CoS values select one of the four egress queues for output processing.

The CoS-to-DSCP, DSCP-to-CoS, and the IP-precedence-to-DSCP map have default values that might or might not be appropriate for your network.

The default DSCP-to-DSCP-mutation map and the default policed-DSCP map are null maps; they map an incoming DSCP value to the same DSCP value. The DSCP-to-DSCP-mutation map is the only map you apply to a specific Gigabit-capable Ethernet port or to a group of 10/100 Ethernet ports. All other maps apply to the entire switch.

For configuration information, see the "Configuring DSCP Maps" section on page 28-52.

Queueing and Scheduling

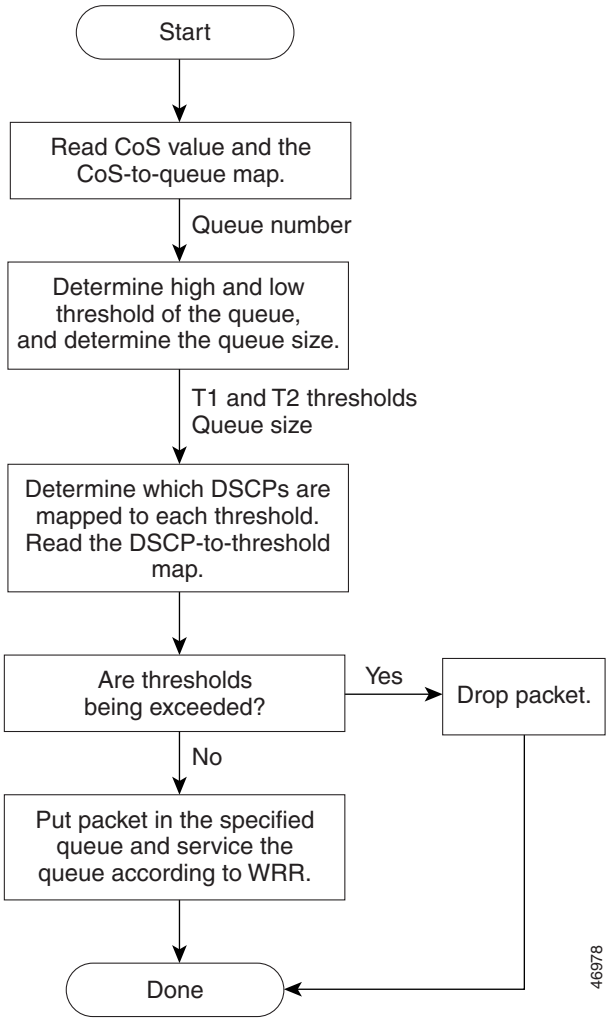
After a packet is policed and marked, the queueing and scheduling process begins as described in these sections:

- Queueing and Scheduling on Gigabit-Capable Ports, page 28-11
- Queueing and Scheduling on 10/100 Ethernet Ports, page 28-14

Queueing and Scheduling on Gigabit-Capable Ports

Figure 28-5 shows the queueing and scheduling flowchart for Gigabit-capable Ethernet ports.

Figure 28-5 Queueing and Scheduling Flowchart for Gigabit-Capable Ethernet Ports



Note If the expedite queue is enabled, WRR services it until it is empty before servicing the other three queues.

During the queueing and scheduling process, the switch uses egress queues and WRR for congestion management, and tail drop or WRED algorithms for congestion avoidance on Gigabit-capable Ethernet ports.

Each Gigabit-capable Ethernet port has four egress queues, one of which can be the egress expedite queue. You can configure the buffer space allocated to each queue as a ratio of weights by using the **wrr-queue queue-limit** interface configuration command, where the relative size differences in the numbers show the relative differences in the queue sizes. To display the absolute value of the queue size, use the **show mls qos interface** *interface-id* **statistics** privileged EXEC command, and examine the FreeQ information.

You assign two drop thresholds to each queue, map DSCPs to the thresholds through the DSCP-to-threshold map, and enable either tail drop or WRED on the interface. The queue size, drop thresholds, tail-drop or WRED algorithm, and the DSCP-to-threshold map work together to determine when and which packets are dropped when the thresholds are exceeded. You configure the drop

percentage thresholds by using either the **wrr-queue threshold** interface configuration command for tail drop or the **wrr-queue random-detect max-threshold** interface configuration command for WRED; in either case, you map DSCP values to the thresholds (DSCP-to-threshold map) by using the **wrr-queue dscp-map** interface configuration command. For more information, see the "Tail Drop" section on page 28-13 and "WRED" section on page 28-13.

The available bandwidth of the egress link is divided among the queues. You configure the queues to be serviced according to the ratio of WRR weights by using the **wrr-queue bandwidth** interface configuration command. The ratio represents the importance (weight) of a queue relative to the other queues. WRR scheduling prevents low-priority queues from being completely neglected during periods of high-priority traffic by sending some packets from each queue in turn. The number of packets sent corresponds to the relative importance of the queue. For example, if one queue has a weight of 3 and another has a weight of 4, three packets are sent from the first queue for every four that are sent from the second queue. By using this scheduling, low-priority queues can send packets even though the high-priority queues are not empty. Queues are selected by the CoS value that is mapped to an egress queue (CoS-to-egress-queue map) through the **wrr-queue cos-map** interface configuration command.

All four queues participate in the WRR unless the expedite queue is enabled, in which case, the fourth bandwidth weight is ignored and not used in the ratio calculation. The expedite queue is a priority queue, and it is serviced until empty before the other queues are serviced. You enable the expedite queue by using the **priority-queue out** interface configuration command.

You can combine the commands described in this section to prioritize traffic by placing packets with particular DSCPs into certain queues, allocate a larger queue size or service the particular queue more frequently, and adjust queue thresholds so that packets with lower priorities are dropped. For configuration information, see the "Configuring Egress Queues on Gigabit-Capable Ethernet Ports" section on page 28-58.

Tail Drop

Tail drop is the default congestion-avoidance technique on Gigabit-capable Ethernet ports. With tail drop, packets are queued until the thresholds are exceeded. Specifically, all packets with DSCPs assigned to the first threshold are dropped until the threshold is no longer exceeded. However, packets assigned to the second threshold continue to be queued and sent as long as the second threshold is not exceeded.

You can modify the two tail-drop threshold percentages assigned to the four egress queues by using the **wrr-queue threshold** interface configuration command. Each threshold value is a percentage of the total number of allocated queue descriptors for the queue. The default threshold is 100 percent for thresholds 1 and 2.

You modify the DSCP-to-threshold map to determine which DSCPs are mapped to which threshold ID by using the **wrr-queue dscp-map** interface configuration command. By default, all DSCPs are mapped to threshold 1, and when this threshold is exceeded, all the packets are dropped.

If you use tail-drop thresholds, you cannot use WRED, and vice versa. If tail drop is disabled, WRED is automatically enabled with the previous configuration (or the default if it was not previously configured).

WRED

Cisco's implementation of Random Early Detection (RED), called Weighted Random Early Detection (WRED), differs from other congestion-avoidance techniques because it attempts to anticipate and avoid congestion, rather than controlling congestion when it occurs.

WRED takes advantage of the Transmission Control Protocol (TCP) congestion control to try to control the average queue size by indicating to end hosts when they should temporarily stop sending packets. By randomly dropping packets before periods of high congestion, it tells the packet source to decrease its transmission rate. Assuming the packet source is using TCP, WRED tells it to decrease its transmission rate until all the packets reach their destination, meaning that the congestion is cleared.

WRED reduces the chances of tail drop by selectively dropping packets when the output interface begins to show signs of congestion. By dropping some packets early rather than waiting until the queue is full, WRED avoids dropping large numbers of packets at once. Thus, WRED allows the transmission line to be fully used at all times. WRED also drops more packets from large users than small. Therefore, sources that generate the most traffic are more likely to be slowed down versus sources that generate little traffic.

You can enable WRED and configure the two threshold percentages assigned to the four egress queues on a Gigabit-capable Ethernet port by using the **wrr-queue random-detect max-threshold** interface configuration command. Each threshold percentage represents where WRED starts to randomly drop packets. After a threshold is exceeded, WRED randomly begins to drop packets assigned to this threshold. As the queue limit is approached, WRED continues to drop more and more packets. When the queue limit is reached, WRED drops all packets assigned to the threshold. By default, WRED is disabled.

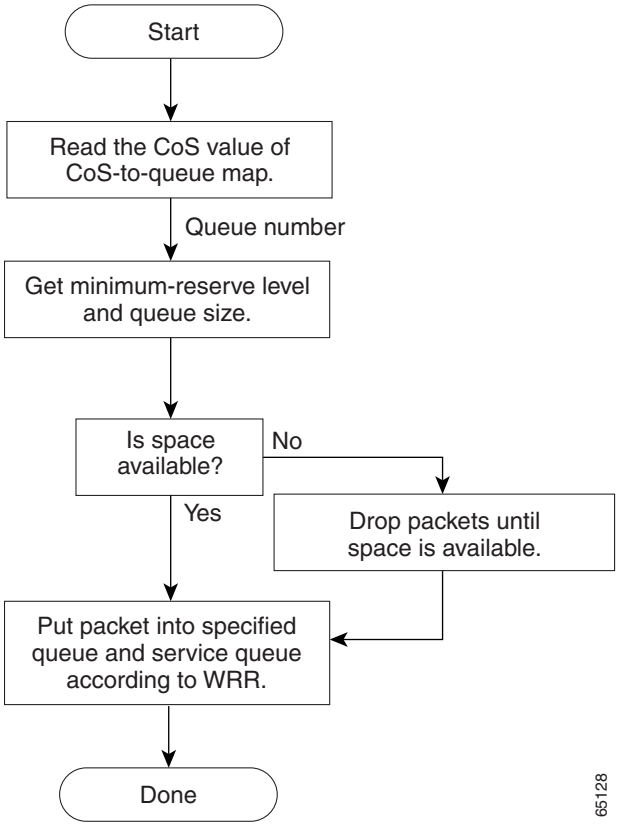
You modify the DSCP-to-threshold map to determine which DSCPs are mapped to which threshold ID by using the **wrr-queue dscp-map** interface configuration command. By default, all DSCPs are mapped to threshold 1, and when this threshold is exceeded, all the packets are randomly dropped.

If you use WRED thresholds, you cannot use tail drop, and vice versa. If WRED is disabled, tail drop is automatically enabled with the previous configuration (or the default if it was not previously configured).

Queueing and Scheduling on 10/100 Ethernet Ports

Figure 28-6 shows the queueing and scheduling flowchart for 10/100 Ethernet ports.

Figure 28-6 Queueing and Scheduling Flowchart for 10/100 Ethernet Ports



Note If the expedite queue is enabled, WRR services it until it is empty before servicing the other three queues.

During the queueing and scheduling process, the switch uses egress queues (to select the minimum-reserve level and buffer size) and WRR for congestion management.

Each 10/100 Ethernet port has four egress queues, one of which can be the egress expedite queue. Each queue can access one of eight minimum-reserve levels; each level has 100 packets of buffer space by default for queueing packets. When the buffer specified for the minimum-reserve level is full, packets are dropped until space is available.

Figure 28-7 is an example of the 10/100 Ethernet port queue assignments, minimum-reserve levels, and buffer sizes. The figure shows four egress queues per port, with each queue assigned to a minimum-reserve level. For example, for Fast Ethernet port 0/1, queue 1 is assigned to minimum-reserve level 1, queue 2 is assigned to minimum-reserve level 3, queue 3 is assigned to minimum-reserve level 5, and queue 4 is assigned to minimum-reserve level 7. You assign the minimum-reserve level to a queue by using the **wrr-queue min-reserve** interface configuration command.

Each minimum-reserve level is configured with a buffer size. As shown in the figure, queue 4 of Fast Ethernet port 1 has a buffer size of 70 packets, queue 4 of Fast Ethernet port 2 has a buffer size of 80 packets, queue 4 of Fast Ethernet port 3 has a buffer size of 40 packets, and Fast Ethernet port 4 has a buffer size of 80 packets. You configure the buffer size by using the **mls qos min-reserve** global configuration command.

Figure 28-7 10/100 Ethernet Port Queue Assignment, Minimum-Reserve Levels, and Buffer Size

Fast Ethernet Port Number	Q1	Q2	Q3	Q4
	MRL*	MRL	MRL	MRL
0/1	1	3	5	7
0/2	2	4	6	8
0/3	1	2	3	4
0/4	5	6	7	8
• • •				

MRL	Buffer size
1	10
2	20
3	30
4	40
5	50
6	60
7	70
8	80

65127

* MRL = Minimum-reserve level

The available bandwidth of the egress link is divided among the queues. You configure the queues to be serviced according to the ratio of WRR weights by using the **wrr-queue bandwidth** interface configuration command. The ratio represents the importance (weight) of a queue relative to the other queues. WRR scheduling prevents low-priority queues from being completely neglected during periods of high-priority traffic by sending some packets from each queue in turn. The number of packets sent corresponds to the relative importance of the queue. For example, if one queue has a weight of 3 and another has a weight of 4, three packets are sent from the first queue for every four that are sent from the second queue. By using this scheduling, low-priority queues can send packets even though the high-priority queues are not empty. Queues are selected by the CoS value that is mapped to an egress queue (CoS-to-egress-queue map) through the **wrr-queue cos-map** interface configuration command.

All four queues participate in the WRR unless the egress expedite queue is enabled, in which case, the fourth bandwidth weight is ignored and not used in the ratio calculation. The expedite queue is a priority queue, and it is serviced until empty before the other queues are serviced. You enable the expedite queue by using the **priority-queue out** interface configuration command.

You can combine the commands described in this section to prioritize traffic by placing packets with particular DSCPs into certain queues, allocate a larger minimum-reserve buffer size, and service a particular queue more frequently. For configuration information, see the "Configuring Egress Queues on 10/100 Ethernet Ports" section on page 28-65.

Packet Modification

A packet is classified, policed, and queued for QoS. Packet modifications can occur during this process:

- For IP packets, classification involves assigning a DSCP to the packet. However, the packet is not modified at this stage; only an indication of the assigned DSCP is carried along. The reason for this is that QoS classification and ACL lookup occur in parallel, and it is possible that the ACL specifies that the packet should be denied and logged. In this situation, the packet is forwarded with its original DSCP to the CPU, where it is again processed through ACL software. However, route lookup is performed based on classified DSCPs.
- For non-IP packets, classification involves assigning an internal DSCP to the packet, but because there is no DSCP in the non-IP packet, no overwrite occurs. Instead, the internal DSCP is translated to the CoS and is used both for queueing and scheduling decisions and for writing the CoS priority value in the tag if the packet is being sent on either an ISL or IEEE 802.1Q trunk port. Because the CoS priority is written in the tag, Catalyst 3500 series XL switches that use the IEEE 802.1p priority can interoperate with the QoS implementation on the Catalyst 3550 switches.

- During policing, IP and non-IP packets can have another DSCP assigned to them (if they are out of profile and the policer specifies a markdown DSCP). For IP packets, the packet modification occurs at a later stage; for non-IP packets the DSCP is converted to CoS and used for queueing and scheduling decisions.

Configuring Auto-QoS

You can use the auto-QoS feature to simplify the deployment of existing QoS features. Auto-QoS makes assumptions about the network design, and as a result, the switch can prioritize different traffic flows and appropriately use the egress queues instead of using the default QoS behavior. (The default is that QoS is disabled. The switch then offers best-effort service to each packet, regardless of the packet contents or size, and sends it from a single queue.)

When you enable auto-QoS, it automatically classifies traffic based on the traffic type and ingress packet label. The switch uses the resulting classification to choose the appropriate egress queue.

You use auto-QoS commands to identify ports connected to Cisco IP Phones and to devices running the Cisco SoftPhone application. You also use the commands to identify ports that receive trusted traffic through an uplink. Auto-QoS then performs these functions:

- Detects the presence or absence of Cisco IP Phones
- Configures QoS classification
- Configures egress queues

These sections describe how to configure auto-QoS on your switch:

- Generated Auto-QoS Configuration, page 28-17
- Effects of Auto-QoS on the Configuration, page 28-20
- Configuration Guidelines, page 28-21
- Upgrading from a Previous Software Release, page 28-21
- Enabling Auto-QoS for VoIP, page 28-22

Generated Auto-QoS Configuration

By default, auto-QoS is disabled on all interfaces.

When auto-QoS is enabled, it uses the ingress packet label to categorize traffic and to configure the egress queues as shown in Table 28-1.

Table 28-1 Traffic Types, Packet Labels, and Egress Queues

	VoIP[1] Data Traffic	VoIP Control Traffic	Routing Protocol Traffic	STP BPDU[2] Traffic	Real-Time Video Traffic	All Other Traffic
DSCP	46	24, 26	48	56	34	–
CoS	5	3	6	7	4	

Table 28-1 Traffic Types, Packet Labels, and Egress Queues (continued) (continued)

	VoIP[1] Data Traffic	VoIP Control Traffic	Routing Protocol Traffic	STP BPDU[2] Traffic	Real-Time Video Traffic	All Other Traffic	
CoS-to-Queue Map	5	3, 6, 7			4	2	0, 1
Egress Queue	Expedite (queue 4)	70% WRR (queue 3)			20% WRR (queue 2)	20% WRR (queue 2)	10% WRR (queue 1)

1. VoIP = voice over IP
2. BPDU = bridge protocol data unit

Table 28-2 shows the generated auto-QoS configuration for the egress queues.

Table 28-2 Auto-QoS Configuration for the Egress Queues

Egress Queue	Queue Number	CoS-to-Queue Map	Queue Weight	Queue Size for Gigabit-Capable Ports	Queue Size (in packets) for 10/100 Ethernet Ports
Expedite	4	5	–	10 percent	34 (10 percent)
70% WRR	3	3, 6, 7	70 percent	15 percent	51 (15 percent)
20% WRR	2	2, 4	20 percent	25 percent	82 (25 percent)
10% WRR	1	0, 1	10 percent	50 percent	170 (50 percent)

When you enable the auto-QoS feature on the first interface, these automatic actions occur:

- QoS is globally enabled (**mls qos** global configuration command).
- When you enter the **auto qos voip cisco-phone** interface configuration command on a port at the edge of the network that is connected to a Cisco IP Phone, the switch enables the trusted boundary feature. The switch uses the Cisco Discovery Protocol (CDP) to detect the presence or absence of a Cisco IP Phone. When a Cisco IP Phone is detected, the ingress classification on the interface is set to trust the QoS label received in the packet. When a Cisco IP Phone is absent, the ingress classification is set to not trust the QoS label in the packet. The switch configures egress queues on the port according to the settings in Table 28-2.
- When you enter the **auto qos voip cisco-softphone** interface configuration command on a port at the edge of the network that is connected to a device running the Cisco SoftPhone, the switch uses policing to determine whether a packet is in or out of profile and to specify the action on the packet. If the packet does not have a DSCP value of 24, 26, or 46 or is out of profile, the switch changes the DSCP value to 0. The switch configures egress queues on the port according to the settings in Table 28-2.
- When you enter the **auto qos voip trust** interface configuration command on a port connected to the interior of the network, the switch trusts the CoS value for nonrouted ports or the DSCP value for routed ports in ingress packets (the assumption is that traffic has already been classified by other edge devices). The switch configures egress queues on the port according to the settings in Table 28-2.

  For information about the trusted boundary feature, see the "Configuring a Trusted Boundary to Ensure Port Security" section on page 28-32.

When you enable auto-QoS by using the **auto qos voip cisco-phone**, the **auto qos voip cisco-softphone**, or the **auto qos voip trust** interface configuration command, the switch automatically generates a QoS configuration based on the traffic type and ingress packet label and applies the commands listed in Table 28-3 to the interface.

Table 28-3 Generated Auto-QoS Configuration

Description	Automatically Generated Command
The switch automatically enables standard QoS and configures the CoS-to-DSCP map (maps CoS values in incoming packets to a DSCP value) as shown in Table 28-1 on page 28-17.	`Switch(config)# mls qos` `Switch(config)# mls qos map cos-dscp 0 8 16 26 32 46 48 56`

If 10/100 Ethernet ports are present, the switch automatically configures the buffer size of the minimum-reserve levels 5, 6, 7, and 8:
• Level 5 can hold 170 packets.
• Level 6 can hold 85 packets.
• Level 7 can hold 51 packets.
• Level 8 can hold 34 packets.
	`Switch(config)# mls qos min-reserve 5 170`
	`Switch(config)# mls qos min-reserve 6 85`
	`Switch(config)# mls qos min-reserve 7 51`
	`Switch(config)# mls qos min-reserve 8 34`

If you entered the **auto qos voip trust** command, the switch automatically sets the ingress classification to trust the CoS value received in the packet on a nonrouted port or to trust the DSCP value received in the packet on a routed port.	`Switch(config-if)# mls qos trust cos` `Switch(config-if)# mls qos trust dscp`
If you entered the **auto qos voip cisco-phone** command, the switch automatically enables the trusted boundary feature, which uses the CDP to detect the presence or absence of a Cisco IP Phone.	`Switch(config-if)# mls qos trust device cisco-phone`

If you entered the **auto qos voip cisco-softphone** command, the switch automatically creates class maps and policy maps.	`Switch(config)# mls qos map policed-dscp 24 26 46 to 0`
	`Switch(config)# class-map match-all AutoQoS-VoIP-RTP-Trust`
	`Switch(config-cmap)# match ip dscp 46`
	`Switch(config)# class-map match-all AutoQoS-VoIP-Control-Trust`
	`Switch(config-cmap)# match ip dscp 24 26`
	`Switch(config)# policy-map AutoQoS-Police-SoftPhone`
	`Switch(config-pmap)# class AutoQoS-VoIP-RTP-Trust`
	`Switch(config-pmap-c)# set dscp 46`
	`Switch(config-pmap-c)# police 320000 8000 exceed-action policed-dscp-transmit`
	`Switch(config-pmap)# class AutoQoS-VoIP-Control-Trust`
	`Switch(config-pmap-c)# set dscp 24`
	`Switch(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit`

After creating the class maps and policy maps, the switch automatically applies the policy map called *AutoQoS-Police-SoftPhone* to an ingress interface on which auto-QoS with the Cisco SoftPhone feature is enabled.	`Switch(config-if)# service-policy input AutoQoS-Police-SoftPhone`

Table 28-3 Generated Auto-QoS Configuration (continued)

Description	Automatically Generated Command

The switch automatically assigns egress queue usage (as shown in Table 28-2 on page 28-18) on this interface.
The switch enables the egress expedite queue and assigns WRR weights to queues 1, 2, and 3. (The lowest value for a WRR queue is 1. When the WRR weight of a queue is set to 0, this queue becomes an expedite queue.)
The switch configures the CoS-to-egress-queue map:
• CoS values 0 and 1 select queue 1.
• CoS values 2 and 4 select queue 2.
• CoS values 3, 6, and 7 select queue 3.
• CoS value 5 selects queue 4 (expedite queue).
Because the expedite queue (queue 4) contains the VoIP data traffic, the queue is serviced until empty.
	`Switch(config-if)# wrr-queue bandwidth 10 20 70 1`
	`Switch(config-if)# no wrr-queue cos-map`
	`Switch(config-if)# wrr-queue cos-map 1 0 1`
	`Switch(config-if)# wrr-queue cos-map 2 2 4`
	`Switch(config-if)# wrr-queue cos-map 3 3 6 7`
	`Switch(config-if)# wrr-queue cos-map 4 5`
	`Switch(config-if)# priority-queue out`

On Gigabit-capable Ethernet ports only, the switch automatically configures the ratio of the sizes of the WRR egress queues: • Queue 1 is 50 percent. • Queue 2 is 25 percent. • Queue 3 is 15 percent. • Queue 4 is 10 percent.	`Switch(config-if)# wrr-queue queue-limit 50 25 15 10`

On 10/100 Ethernet ports only, the switch automatically configures minimum-reserve levels for the egress queues:
• Queue 1 selects the minimum-reserve level 5.
• Queue 2 selects the minimum-reserve level 6.
• Queue 3 selects the minimum-reserve level 7.
• Queue 4 selects the minimum-reserve level 8.
	`Switch(config-if)# wrr-queue min-reserve 1 5`
	`Switch(config-if)# wrr-queue min-reserve 2 6`
	`Switch(config-if)# wrr-queue min-reserve 3 7`
	`Switch(config-if)# wrr-queue min-reserve 4 8`

Effects of Auto-QoS on the Configuration

When auto-QoS is enabled, the **auto qos voip** interface configuration command and the generated configuration are added to the running configuration.

The switch applies the auto-QoS-generated commands as if the commands were entered from the CLI. An existing user configuration can cause the application of the generated commands to fail or to be overridden by the generated commands. These actions occur without warning. If all the generated commands are successfully applied, any user-entered configuration that was not overridden remains in the running configuration. Any user-entered configuration that was overridden can be retrieved by reloading the switch without saving the current configuration to memory. If the generated commands fail to be applied, the previous running configuration is restored.

Configuration Guidelines

Before configuring auto-QoS, you should be aware of this information:

- In releases earlier than Cisco IOS Release 12.1(20)EA2, auto-QoS configures the switch for VoIP only with Cisco IP Phones on nonrouted ports.
- In Cisco IOS Release 12.1(20)EA2 or later, auto-QoS configures the switch for VoIP with Cisco IP Phones on nonrouted and routed ports. Auto-QoS also configures the switch for VoIP with devices running the Cisco SoftPhone application.

Note When a device running Cisco SoftPhone is connected to a nonrouted or routed port, the switch supports only one Cisco SoftPhone application per port.

- To take advantage of the auto-QoS defaults, you should enable auto-QoS before you configure other QoS commands. If necessary, you can fine-tune the QoS configuration, but we recommend that you do so only after the auto-QoS configuration is completed. For more information, see the "Effects of Auto-QoS on the Configuration" section on page 28-20.
- After auto-QoS is enabled, do not modify a policy map or aggregate policer that includes *AutoQoS* in its name. If you need to modify the policy map or aggregate policer, make a copy of it, and change the copied policy map or policer. To use this new policy map instead of the generated one, remove the generated policy map from the interface, and apply the new policy map to the interface.
- You can enable auto-QoS on static, dynamic-access, voice VLAN access, and trunk ports.
- By default, the CDP is enabled on all interfaces. For auto-QoS to function properly, do not disable the CDP.
- When enabling auto-QoS with a Cisco IP Phone on a routed port, you must assign a static IP address to the IP phone.
- This release supports only Cisco IP SoftPhone Version 1.3(3) or later.
- Connected devices must use Cisco Call Manager Version 4 or later.

Upgrading from a Previous Software Release

In Cisco IOS Release 12.2(20)EA2, the implementation for auto-QoS changed from the previous release. The generated auto-QoS configuration was changed, support for the Cisco SoftPhone feature was added, and support for Cisco IP Phones on routed ports was added.

If auto-QoS is configured on the switch, if your switch is running a release earlier than Cisco IOS Release 12.2(20)EA2, and if you upgrade to Cisco IOS Release 12.2(20)EA2 or later, the configuration file will not contain the new configuration, and auto-QoS will not operate. Follow these steps to update the auto-QoS settings in your configuration file:

1. Upgrade your switch to Cisco IOS Release 12.2(20)EA2 or later.
2. Disable auto-QoS on all ports on which auto-QoS was enabled.
3. Return all the global auto-QoS settings to their default values by using the **no** commands.
4. Re-enable auto-QoS on the ports on which auto-QoS was disabled in Step 2. Configure the ports with the same auto-QoS settings as the previous ones.

Enabling Auto-QoS for VoIP

Beginning in privileged EXEC mode, follow these steps to enable auto-QoS for VoIP within a QoS domain:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface that is connected to a Cisco IP Phone or the uplink interface that is connected to another trusted switch or router in the interior of the network, and enter interface configuration mode.
Step 3	**auto qos voip** {cisco-phone | **cisco-softphone** | trust}	Enable auto-QoS. The keywords have these meanings: • **cisco-phone**—If the interface is connected to a Cisco IP Phone, the QoS labels of incoming packets are trusted only when the telephone is detected. • **cisco-softphone**—The port is connected to device running the Cisco SoftPhone feature. **Note** The **cisco-softphone** keyword is supported only in Cisco IOS Release 12.2(20)EA2 or later. • **trust**—The uplink interface is connected to a trusted switch or router, and the VoIP traffic classification in the ingress packet is trusted.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show auto qos interface** *interface-id*	Verify your entries. This command displays the QoS commands on the interface on which auto-QoS was enabled. You can use the **show running-config** privileged EXEC command to display the auto-QoS configuration and the user modifications.

To display the QoS commands that are automatically generated when auto-QoS is enabled or disabled, enter the **debug auto qos** privileged EXEC command before enabling auto-QoS. For more information, see the "Using the debug auto qos Command" section on page 6-18.

To disable auto-QoS on an interface, use the **no auto qos voip** interface configuration command. When you enter this command, the switch changes the auto-QoS settings to the standard-QoS default settings for that interface.

To disable auto-QoS on the switch, use the **no mls qos** global configuration command. When you enter this command, the switch disables QoS on all interfaces and enables pass-through mode.

This example shows how to enable auto-QoS and to trust the QoS labels in incoming packets when the device connected to the interface is detected as a Cisco IP Phone:

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# auto qos voip cisco-phone
```

This example shows how to enable auto-QoS and to trust the QoS labels in incoming packets when the switch or router connected to the interface is a trusted device:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# auto qos voip trust
```

Displaying Auto-QoS Information

To display the initial auto-QoS configuration, use the **show auto qos** [**interface** [*interface-id*]] privileged EXEC command. To display any user changes to that configuration, use the **show running-config** privileged EXEC command. You can compare the **show auto qos** and the **show running-config** command output to identify the user-defined QoS settings.

To display information about the QoS configuration that might be affected by auto-QoS, use one of these commands:

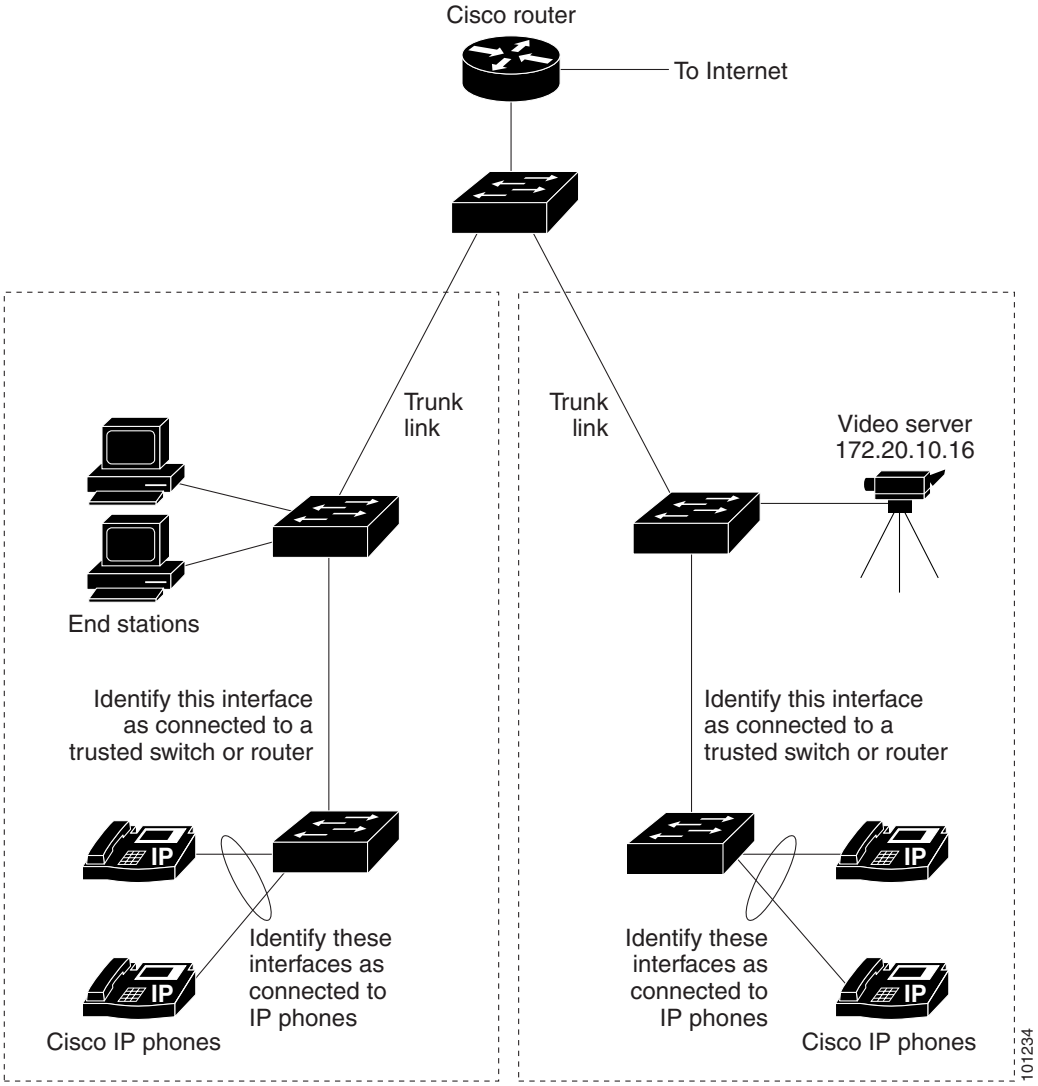
- **show mls qos**
- **show mls qos map cos-dscp**
- **show mls qos interface** [*interface-id*] [**buffers** | **queueing**]

For more information about these commands, see the command reference for this release.

Auto-QoS Configuration Example

This section describes how you could implement auto-QoS in a network, as shown in Figure 28-8. For optimum QoS performance, auto-QoS should be enabled on all the devices in the network.

Figure 28-8 Auto-QoS Configuration Example Network



The intelligent wiring closets in Figure 28-8 are composed of Catalyst 2950 switches running the EI and Catalyst 3550 switches. The object of this example is to prioritize the VoIP traffic over all other traffic. To do so, enable auto-QoS on the switches at the edge of the QoS domains in the wiring closets.

Note You should not configure any standard QoS commands before entering the auto-QoS commands. You can fine-tune the QoS configuration, but we recommend that you do so only after the auto-QoS configuration is completed.

Beginning in privileged EXEC mode, follow these steps to configure the switch at the edge of the QoS domain to prioritize the VoIP traffic over all other traffic:

	Command	Purpose
Step 1	debug auto qos	Enable debugging for auto-QoS. When debugging is enabled, the switch displays the QoS configuration that is automatically generated when auto-QoS is enabled.
Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**cdp enable**	Enable CDP globally. By default, it is enabled.
Step 4	**interface** *interface-id*	Specify the switch port connected to the Cisco IP Phone, and enter interface configuration mode.
Step 5	**auto qos voip** cisco-phone	Enable auto-QoS on the interface, and specify that the interface is connected to a Cisco IP Phone. The QoS labels of incoming packets are trusted only when the IP phone is detected.
Step 6	exit	Return to global configuration mode.
Step 7		Repeat Steps 4 to 6 for as many ports as are connected to the Cisco IP Phone.
Step 8	**interface** *interface-id*	Specify the switch port identified as connected to a trusted switch or router, and enter interface configuration mode. See Figure 28-8.
Step 9	**auto qos voip trust**	Enable auto-QoS on the interface, and specify that the interface is connected to a trusted router or switch.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**show auto qos**	Verify your entries. This command displays the auto-QoS command on the interface on which auto-QoS was enabled. You can use the **show running-config** privileged EXEC command to display the auto-QoS configuration and the user modifications. For information about the QoS configuration that might be affected by auto-QoS, see the "Displaying Auto-QoS Information" section on page 26-12.
Step 12	**copy running-config startup-config**	Save the **auto qos voip** interface configuration commands and the generated auto-QoS configuration in the configuration file.

Configuring Standard QoS

Before configuring standard QoS, you must have a thorough understanding of these items:

- The types of applications used and the traffic patterns on your network.
- Traffic characteristics and needs of your network. Is the traffic bursty? Do you need to reserve bandwidth for voice and video streams?
- Bandwidth requirements and speed of the network.
- Location of congestion points in the network.

These sections describe how to configure standard QoS on your switch:

- Default Standard QoS Configuration, page 28-26
- Standard QoS Configuration Guidelines, page 28-27
- Enabling QoS Globally, page 28-29
- Configuring Classification By Using Port Trust States, page 28-30
- Configuring a QoS Policy, page 28-36
- Configuring DSCP Maps, page 28-52
- Configuring Egress Queues on Gigabit-Capable Ethernet Ports, page 28-58
- Configuring Egress Queues on 10/100 Ethernet Ports, page 28-65

Default Standard QoS Configuration

Table 28-4 shows the default standard QoS configuration when QoS is disabled.

Table 28-4 Default Standard QoS Configuration when QoS is Disabled

Port Type	QoS State	Egress traffic (DSCP and CoS Value)	Queue	Queue Weights	Tail-drop Thresholds	CoS Mapping to Queue
Gigabit-capable Ethernet ports	Disabled	Pass through.	All of the queue RAM is allocated to queue 1 (no expedite queue).	–	100%, 100% WRED is disabled.	All CoS values map to queue 1.
10/100 Ethernet ports	Disabled	Pass through.	Each of the eight minimum-reserve levels have a buffer size of 100 packets. The queue selects the level.	–	–	All CoS values map to queue 1.

When QoS is disabled, there is no concept of trusted or untrusted ports because the packets are not modified (the CoS, DSCP, and IP precedence values in the packet are not changed).

Table 28-5 shows the default standard QoS configuration without any further configuration when QoS is enabled.

Table 28-5 Default Standard QoS Configuration when QoS is Enabled

Port Type	QoS State	Egress traffic (DSCP and CoS Value)	Queue	Queue Weights	Tail-drop Thresholds	CoS Mapping to Queue
Gigabit-capable Ethernet ports	Enabled (no policing)	DSCP=0 CoS=0 (0 means best-effort delivery.)	Four queues are available (no expedite queue).	Each queue has the same weight.	100%, 100% WRED is disabled.	0, 1: queue 1 2, 3: queue 2 4, 5: queue 3 6, 7: queue 4
10/100 Ethernet ports	Enabled (no policing)	DSCP=0 CoS=0 (0 means best-effort delivery.)	Each of the eight minimum-reserve levels have a buffer size of 100 packets. The queue selects the level.	Each queue has the same weight.	–	0, 1: queue 1 2, 3: queue 2 4, 5: queue 3 6, 7: queue 4

The default port CoS value is 0.

The default port trust state on all ports is untrusted.

No policy maps are configured.

No policers are configured.

The default CoS-to-DSCP map is shown in Table 28-6 on page 28-53.

The default IP-precedence-to-DSCP map is shown in Table 28-7 on page 28-54.

The default DSCP-to-CoS map is shown in Table 28-8 on page 28-55.

The default DSCP-to-DSCP-mutation map is a null map, which maps an incoming DSCP value to the same DSCP value.

The default policed-DSCP map is a null map, which maps an incoming DSCP value to the same DSCP value (no markdown).

The default DSCP-to-switch-priority map maps DSCPs 0 to 15 to priority 0, DSCPs 16 to 31 to priority 1, DSCPs 32 to 47 to priority 2, and DSCPs 48 to 63 to priority 3.

Standard QoS Configuration Guidelines

Before beginning the QoS configuration, you should be aware of this information:

- You must disable the IEEE 802.3x flow control on all ports before enabling QoS on the switch. To disable it, use the **flowcontrol receive off** and **flowcontrol send off** interface configuration commands.

Note If QoS is disabled and you enter the **mls qos** global configuration command, this message appears:

```
QoS:ensure flow-control on all interfaces are OFF for proper operation.
```

- If you have EtherChannel ports configured on your switch, you must configure QoS classification, policing, mapping, and queueing on the individual physical ports that comprise the EtherChannel. You must decide whether the QoS configuration should match on all ports in the EtherChannel.
- You can classify traffic on an ingress physical port or on a per-ingress-port per-VLAN basis. You cannot classify traffic at the switch virtual interface level.
- Only one ACL per class map and only one **match** class-map configuration command per class map are supported. The ACL can have multiple access control entries, which are commands that match fields against the contents of the packet.
- When classifying traffic on a per-port per-VLAN basis, you must use the **match-all** keyword with the **class-map** global configuration command. For more information, see the "Classifying Traffic on a Per-Port Per-VLAN Basis by Using Class Maps" section on page 28-41.
- The switch has only 256 VLAN labels (a few are always used internally for defaults), which are shared between VLAN maps and per-port per-VLAN policing. If a large number of VLANs are used in class maps and either different ACL actions are performed on them or they have different VLAN maps applied, the available VLAN labels might be insufficient. As a consequence, the TCAM entries are not programmed, and the feature does not work. Use the **show tcam qos** *tcam-id* **port-labels vlan-labels** privileged EXEC command to display how many VLAN labels are in use by this QoS feature.
- It is not possible to match IP fragments against configured IP extended ACLs to enforce QoS. IP fragments are sent as best-effort. IP fragments are denoted by fields in the IP header.
- You can match IP options against configured IP extended ACLs to enforce QoS. These packets are sent to the CPU and processed by software. IP options are denoted by fields in the IP header.
- You can configure a policer on an ingress or egress physical port; you can configure a per-port per-VLAN policer only on an ingress port (specifies the bandwidth limits for the traffic on a per-VLAN basis, for a given port). You cannot police at the switch virtual interface level.

  You cannot configure per-port per-VLAN policing on routed ports or on virtual (logical) interfaces. It is supported only on an ingress port configured as a trunk or as a static-access port.

  The switch does not support per-VLAN QoS or VLAN QoS policing across the entire switch.
- Use only the **match ip dscp** *dscp-list* class-map configuration command in a policy map that is attached to an egress interface.
- You cannot classify traffic by using a port trust state (for example, **mls qos trust** [**cos** | **dscp** | **ip-precedence**] and by using a policy map (for example, **service-policy input** *policy-map-name*) at the same time on an interface. These commands are mutually exclusive. The last one configured overwrites the previous configuration.
- You cannot use the **service-policy** interface configuration command to attach policy maps that contain these elements to an egress interface:
  - **set** or **trust** policy-map class configuration commands. Instead, you can use the **police** policy-map class configuration command to mark down (reduce) the DSCP value at the egress interface.
  - Access control list (ACL) classification.
  - Per-port per-VLAN classification.

  The only match criterion in a policy map that can be attached to an egress interface is the **match ip dscp** *dscp-list* class-map configuration command.
- You can create an aggregate policer that is shared by multiple traffic classes within the same policy map. However, you cannot use the aggregate policer across different policy maps or interfaces.

- All ingress QoS processing actions apply to control traffic (such as spanning-tree bridge protocol data units [BPDUs] and routing update packets) that the switch receives.
- Layer 3 QoS ACLs and other QoS features related to Layer 3 information are not supported on tunnel ports. MAC-based QoS is supported on tunnel ports. When applied to trunk ports, Layer 3 QoS ACLs do not work for VLANs that include tunnel ports.
- Do not use the **show policy-map interface** privileged EXEC command to display classification information for incoming traffic. The **interface** keyword is not supported, and you should ignore the statistics shown in the display. Instead, you should specify the DSCPs to be monitored by using the **mls qos monitor dscp** *dscp1 ... dscp8* interface configuration command, and then you should use the **show mls qos interface** *interface-id* **statistics** privileged EXEC command. For more information about these commands, see the command reference for this release.

Enabling QoS Globally

By default, QoS is disabled on the switch, which means that the switch offers best-effort service to each packet regardless of the packet contents or size. All CoS values map to egress queue 1 with both tail-drop thresholds set to 100 percent of the total queue size for Gigabit-capable Ethernet ports. On 10/100 Ethernet ports, all CoS values map to egress queue 1, which uses minimum-reserve level 1 and can hold up to 100 packets. When the buffer is full, packets are dropped.

Beginning in privileged EXEC mode, follow these steps to enable QoS:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface range** *port-range*	Enter interface configuration mode, and execute a command on multiple interfaces. You can define up to five interface ranges with a single command, with each range separated by a comma. All interfaces in a range must be the same type; that is, all Fast Ethernet ports or all Gigabit Ethernet ports.
Step 3	**flowcontrol receive off** **flowcontrol send off**	Disable flow control on all interfaces.
Step 4	**exit**	Return to global configuration mode.
Step 5	**mls qos**	Enable QoS globally.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show mls qos**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After QoS is enabled, the default settings are as shown in Table 28-4 on page 28-26.

To disable QoS, use the **no mls qos** global configuration command.

Configuring Classification By Using Port Trust States

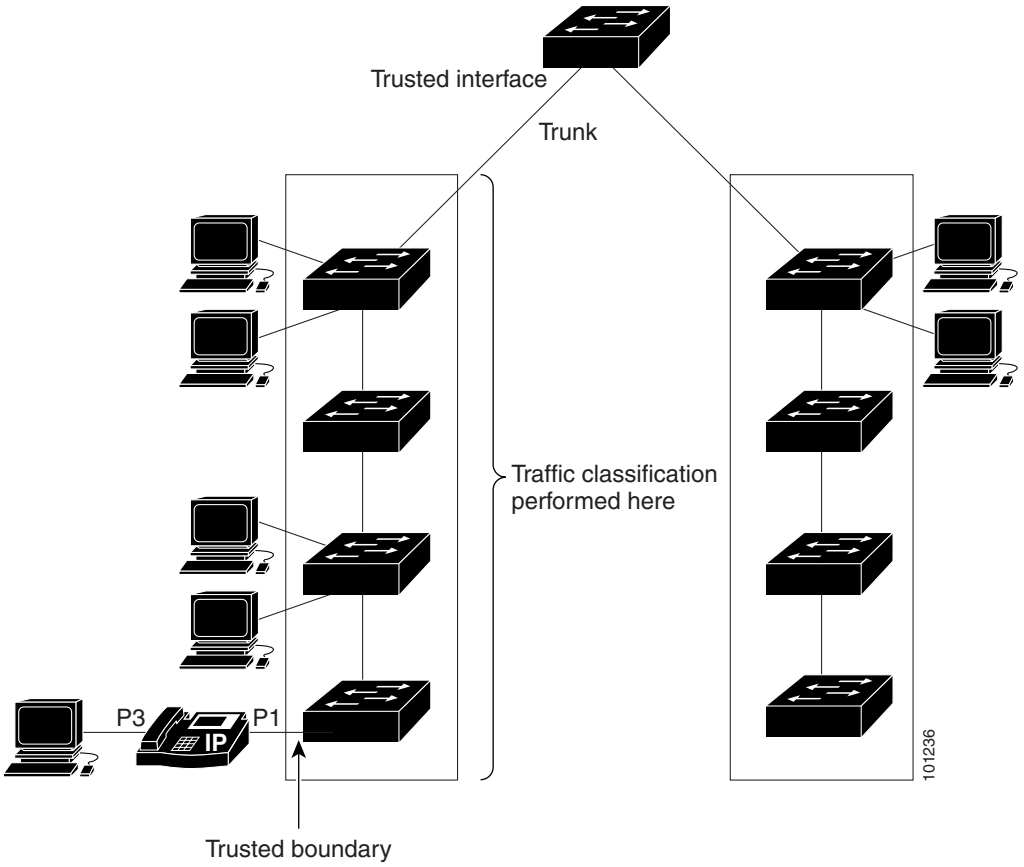
These sections describe how to classify incoming traffic by using port trust states:

- Configuring the Trust State on Ports within the QoS Domain, page 28-30
- Configuring the CoS Value for an Interface, page 28-32
- Configuring a Trusted Boundary to Ensure Port Security, page 28-32
- Enabling Pass-Through Mode, page 28-34
- Configuring the DSCP Trust State on a Port Bordering Another QoS Domain, page 28-34

Configuring the Trust State on Ports within the QoS Domain

Packets entering a QoS domain are classified at the edge of the QoS domain. When the packets are classified at the edge, the switch port within the QoS domain can be configured to one of the trusted states because there is no need to classify the packets at every switch within the QoS domain. Figure 28-9 shows a sample network topology.

Figure 28-9 Port Trusted States within the QoS Domain



Beginning in privileged EXEC mode, follow these steps to configure the port to trust the classification of the traffic that it receives:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS globally.
Step 3	**interface** *interface-id*	Specify the interface to be trusted, and enter interface configuration mode. Valid interfaces include physical interfaces.
Step 4	**mls qos trust** {**cos** | **dscp** | **ip-precedence**}	Configure the port trust state. By default, the port is not trusted. The keywords have these meanings: • **cos**—Classifies ingress packets with the packet CoS values. For untagged packets, the port default CoS value is used. The default port CoS value is 0. • **dscp**—Classifies ingress packets with packet DSCP values. For non-IP packets, the packet CoS value is used if the packet is tagged; for untagged packets, the default port CoS is used. Internally, the switch maps the CoS value to a DSCP value by using the CoS-to-DSCP map. • **ip-precedence**—Classifies ingress packets with the packet IP-precedence values. For non-IP packets, the packet CoS value is used if the packet is tagged; for untagged packets, the default port CoS is used. Internally, the switch maps the CoS value to a DSCP value by using the CoS-to-DSCP map. Use the **cos** keyword setting if your network is composed of Ethernet LANs, Catalyst 3500 XL and 2900 XL switches, and has no more than two types of traffic. Recall that on Catalyst 3500 XL and 2900 XL switches, CoS configures each transmitting port with a normal-priority transmit queue and a high-priority transmit queue. Use the **dscp** or **ip-precedence** keyword if your network is not composed of only Ethernet LANs and if you are familiar with sophisticated QoS features and implementations.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show mls qos interface**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return a port to its untrusted state, use the **no mls qos trust** interface configuration command.

For information on how to change the default CoS value, see the "Configuring the CoS Value for an Interface" section on page 28-32. For information on how to configure the CoS-to-DSCP map, see the "Configuring the CoS-to-DSCP Map" section on page 28-52.

Configuring the CoS Value for an Interface

QoS assigns the CoS value specified with the **mls qos cos** interface configuration command to untagged frames received on trusted and untrusted ports.

Beginning in privileged EXEC mode, follow these steps to define the default CoS value of a port or to assign the default CoS to all incoming packets on the port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS globally.
Step 3	**interface** *interface-id*	Specify the interface to be trusted, and enter interface configuration mode. Valid interfaces include physical interfaces.
Step 4	**mls qos cos** {*default-cos* | **override**}	Configure the default CoS value for the port. • For *default-cos,* specify a default CoS value to be assigned to a port. If the port is CoS trusted and packets are untagged, the default CoS value becomes the CoS value for the packet. The CoS range is 0 to 7. The default is 0. • Use the **override** keyword to override the previously configured trust state of the incoming packets and to apply the default port CoS value to all incoming packets. By default, CoS override is disabled. Use the **override** keyword when all incoming packets on certain ports deserve higher or lower priority than packets entering from other ports. Even if a port was previously set to trust DSCP, CoS, or IP precedence, this command overrides the previously configured trust state, and all the incoming CoS values are assigned the default CoS value configured with this command. If an incoming packet is tagged, the CoS value of the packet is modified with the default CoS of the port at the ingress port.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show mls qos interface**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no mls qos cos** {*default-cos* | **override**} interface configuration command.

Configuring a Trusted Boundary to Ensure Port Security

In a typical network, you connect a Cisco IP Phone to a switch port as shown in Figure 28-9 on page 28-30, and cascade devices that generate data packets from the back of the telephone. The Cisco IP Phone guarantees the voice quality through a shared data link by marking the CoS level of the voice packets as high priority (CoS = 5) and by marking the data packets as low priority (CoS = 0). Traffic sent from the telephone to the switch is typically marked with a tag that uses the IEEE 802.1Q header. The header contains the VLAN information and the class of service (CoS) 3-bit field, which is the priority of the packet.

For most Cisco IP Phone configurations, the traffic sent from the telephone to the switch should be trusted to ensure that voice traffic is properly prioritized over other types of traffic in the network. By using the **mls qos trust cos** interface configuration command, you configure the switch port to which the telephone is connected to trust the CoS labels of all traffic received on that port. Use the **mls qos trust dscp** interface configuration command to configure a routed port to which the telephone is connected to trust the DSCP labels of all traffic received on that port.

With the trusted setting, you also can use the trusted boundary feature to prevent misuse of a high-priority queue if a user bypasses the telephone and connects the PC directly to the switch. Without trusted boundary, the CoS labels generated by the PC are trusted by the switch (because of the trusted CoS setting). By contrast, trusted boundary uses CDP to detect the presence of a Cisco IP Phone (such as the Cisco IP Phone 7910, 7935, 7940, and 7960) on a switch port. If the telephone is not detected, the trusted boundary feature disables the trusted setting on the switch port and prevents misuse of a high-priority queue. Note that the trusted boundary feature is not effective if the PC and Cisco IP Phone are connected to a hub that is connected to the switch.

In some situations, you can prevent a PC connected to the Cisco IP Phone from taking advantage of a high-priority data queue. You can use the **switchport priority extend cos** interface configuration command to configure the telephone through the switch CLI to override the priority of the traffic received from the PC.

Beginning in privileged EXEC mode, follow these steps to enable trusted boundary on a port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS globally.
Step 3	**cdp run**	Enable CDP globally. By default, CDP is enabled.
Step 4	**interface** *interface-id*	Specify the interface connected to the IP phone, and enter interface configuration mode. Valid interfaces include physical interfaces.
Step 5	**cdp enable**	Enable CDP on the interface. By default, CDP is enabled.
Step 6	**mls qos trust cos** or mls qos trust dscp	Configure the switch port to trust the CoS value in traffic received from the Cisco IP Phone. or Configure the routed port to trust the DSCP value in traffic received from the Cisco IP Phone. By default, the port is not trusted.
Step 7	**mls qos trust device cisco-phone**	Specify that the Cisco IP Phone is a trusted device. You cannot enable both trusted boundary and auto-QoS (**auto qos voip** interface configuration command) at the same time; they are mutually exclusive.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show mls qos interface**	Verify your entries.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the trusted boundary feature, use the **no mls qos trust device** interface configuration command.

Enabling Pass-Through Mode

You can use the pass-through mode to enable the CoS and DSCP setting to be independent for packets that contain both values. Use the pass-through mode when you do not want the other value (CoS or DSCP) to be modified when using the **mls qos trust** [**cos** | **dscp**] interface configuration command.

By default, in software releases earlier than Cisco IOS Release 12.1(11)EA1, if you configure the interface to trust the DSCP, the switch does not modify the DSCP field of the IP packet. However, the switch modifies the CoS value of the packet according to the DSCP-to-CoS map. If you configure the interface to trust the CoS, the switch does not modify the CoS field of the packet. However, the switch modifies the DSCP according to the CoS-to-DSCP map if the packet is an IP packet.

In Cisco IOS Release 12.1(11)EA1 or later, you configure the interface for pass-through mode. The interface trusts the DSCP, and the switch sends the packet without modifying the CoS value (the DSCP-to-CoS map is ignored). Otherwise, the interface trusts the CoS, and the switch sends the packet without modifying the DSCP value. The CoS-to-DSCP map is ignored.

Beginning in privileged EXEC mode, follow these steps to enable pass-through mode on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface on which pass-through mode is enabled, and enter interface configuration mode. Valid interfaces include physical interfaces.
Step 3	**mls qos trust cos pass-through dscp** or mls qos trust dscp pass-through cos	Enable pass-through mode. The interface is configured to trust the CoS value of the incoming packets. The switch does not modify the DSCP value. or Enable pass-through mode. The interface is configured to trust the DSCP value of the incoming packets. The switch does not modify the CoS value.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show mls qos interface** [*interface-id*]	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable pass-through mode, use the **no mls qos trust cos pass-through dscp** or the **no mls qos trust dscp pass-through cos** interface configuration command.

If you configure the **mls qos trust** [**cos pass-through dscp** | **dscp pass-through cos**] interface configuration command and then configure the **mls qos trust** [**cos** | **dscp**] interface configuration command, pass-through mode is disabled.

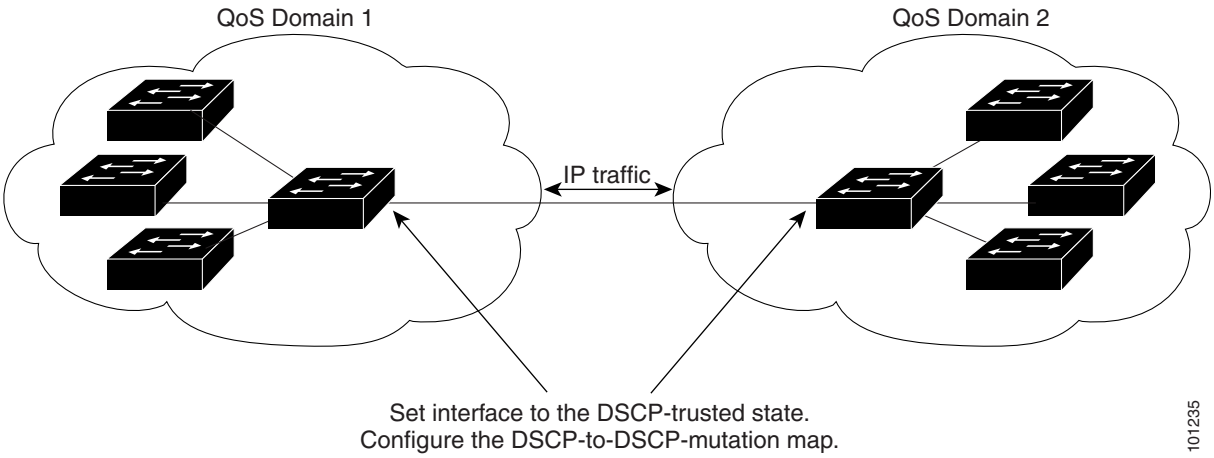
Note If you configure an interface for DSCP pass-through mode by using the **mls qos trust cos pass-through dscp** interface configuration command and apply the DSCP-to-DSCP mutation map to the same interface, the DSCP value changes according to the mutation map.

Configuring the DSCP Trust State on a Port Bordering Another QoS Domain

If you are administering two separate QoS domains between which you want to implement QoS features for IP traffic, you can configure the switch ports bordering the domains to a DSCP-trusted state as shown in Figure 28-10. Then the receiving port accepts the DSCP-trusted value and avoids the classification

stage of QoS. If the two domains use different DSCP values, you can configure the DSCP-to-DSCP-mutation map to translate a set of DSCP values to match the definition in the other domain.

Figure 28-10 DSCP-Trusted State on a Port Bordering Another QoS Domain



Beginning in privileged EXEC mode, follow these steps to configure the DSCP-trusted state on a port and modify the DSCP-to-DSCP-mutation map. To ensure a consistent mapping strategy across both QoS domains, you must perform this procedure on the ports in both domains:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**mls qos map dscp-mutation** *dscp-mutation-name in-dscp* **to** *out-dscp*	Modify the DSCP-to-DSCP-mutation map. The default DSCP-to-DSCP-mutation map is a null map, which maps an incoming DSCP value to the same DSCP value. • For *dscp-mutation-name*, enter the mutation map name. You can create more than one map by specifying a new name. • For *in-dscp*, enter up to eight DSCP values separated by spaces. Then enter the **to** keyword. • For *out-dscp*, enter a single DSCP value. The DSCP range is 0 to 63.
Step 4	**interface** *interface-id*	Specify the interface to be trusted, and enter interface configuration mode. Valid interfaces include physical interfaces.
Step 5	**mls qos trust dscp**	Configure the ingress port as a DSCP-trusted port.

	Command	Purpose
Step 6	**mls qos dscp-mutation** *dscp-mutation-name*	Apply the map to the specified ingress DSCP-trusted port. You can apply the map to different Gigabit-capable Ethernet ports. However, on 10/100 Ethernet ports, you can attach only one DSCP-to-DSCP-mutation map to a group of twelve ports. For example, Fast Ethernet ports 1 to 12 are a group, Fast Ethernet ports 13 to 24 are a group, Gigabit Ethernet 1 is a group, and Gigabit Ethernet 2 is a group. When applying a mutation map to any port in a group, all ports in the same group are automatically configured with the same map.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show mls qos maps dscp-mutation**	Verify your entries.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return a port to its non-trusted state, use the **no mls qos trust** interface configuration command. To return to the default DSCP-to-DSCP-mutation map values, use the **no mls qos map dscp-mutation** *dscp-mutation-map-name* global configuration command.

This example shows how to configure an interface to the DSCP-trusted state and to modify the DSCP-to-DSCP-mutation map (named *gi0/2-mutation*) so that incoming DSCP values 10 to 13 are mapped to DSCP values 30:

```
Switch# configure terminal
Switch(config)# mls qos map dscp-mutation gi0/2-mutation 10 11 12 13 to 30
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# mls qos trust dscp
Switch(config-if)# mls qos dscp-mutation gi0/2-mutation
Switch(config-if)# end
```

Configuring a QoS Policy

Configuring a QoS policy typically requires classifying traffic into classes, configuring policies applied to those traffic classes, and attaching policies to interfaces.

For background information, see the "Classification" section on page 28-5 and the "Policing and Marking" section on page 28-8.

These sections how to configure a QoS policy:

- Classifying Traffic by Using ACLs, page 28-36
- Classifying Traffic on a Physical-Port Basis by Using Class Maps, page 28-40
- Classifying Traffic on a Per-Port Per-VLAN Basis by Using Class Maps, page 28-41
- Classifying, Policing, and Marking Traffic by Using Policy Maps, page 28-44
- Classifying, Policing, and Marking Traffic by Using Aggregate Policers, page 28-50

Classifying Traffic by Using ACLs

You can classify IP traffic by using IP standard or IP extended ACLs; you can classify non-IP traffic by using Layer 2 MAC ACLs.

Beginning in privileged EXEC mode, follow these steps to create an IP standard ACL for IP traffic:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create an IP standard ACL, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number. The range is 1 to 99 and 1300 to 1999. • Use the **permit** keyword to permit a certain type of traffic if the conditions are matched. Use the **deny** keyword to deny a certain type of traffic if conditions are matched. • For *source*, enter the network or host from which the packet is being sent. You can use the **any** keyword as an abbreviation for 0.0.0.0 255.255.255.255. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. **Note** When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show access-lists**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an access list, use the **no access-list** *access-list-number* global configuration command.

This example shows how to allow access for only those hosts on the three specified networks. The wildcard bits apply to the host portions of the network addresses. Any host with a source address that does not match the access list statements is rejected.

```
Switch(config)# access-list 1 permit 192.5.255.0 0.0.0.255
Switch(config)# access-list 1 permit 128.88.0.0 0.0.255.255
Switch(config)# access-list 1 permit 36.0.0.0 0.0.0.255
! (Note: all other access implicitly denied)
```

Beginning in privileged EXEC mode, follow these steps to create an IP extended ACL for IP traffic:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.

	Command	Purpose
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*	Create an IP extended ACL, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number. The range is 100 to 199 and 2000 to 2699. • Use the **permit** keyword to permit a certain type of traffic if the conditions are matched. Use the **deny** keyword to deny a certain type of traffic if conditions are matched. • For *protocol*, enter the name or number of an IP protocol. Use the question mark (?) to see a list of available protocol keywords. • For *source*, enter the network or host from which the packet is being sent. You specify this by using dotted decimal notation, by using the **any** keyword as an abbreviation for *source* 0.0.0.0 *source-wildcard* 255.255.255.255, or by using the **host** keyword for *source* 0.0.0.0. • For *source-wildcard*, enter the wildcard bits by placing ones in the bit positions that you want to ignore. You specify the wildcard by using dotted decimal notation, by using the **any** keyword as an abbreviation for *source* 0.0.0.0 *source-wildcard* 255.255.255.255, or by using the **host** keyword for *source* 0.0.0.0. • For *destination*, enter the network or host to which the packet is being sent. You have the same options for specifying the *destination* and *destination-wildcard* as those described by *source* and *source-wildcard*. **Note** When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show access-lists**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an access list, use the **no access-list** *access-list-number* global configuration command.

This example shows how to create an ACL that permits IP traffic with a DSCP value set to 32 from any source to any destination:

```
Switch(config)# access-list 100 permit ip any any dscp 32
```

This example shows how to create an ACL that permits IP traffic with a precedence value of 5 from a source host at 10.1.1.1 to a destination host at 10.1.1.2:

```
Switch(config)# access-list 100 permit ip host 10.1.1.1 host 10.1.1.2 precedence 5
```

This example shows how to create an ACL that permits PIM traffic with a DSCP set to 32 from any source to a destination group address of 224.0.0.2:

```
Switch(config)# access-list 102 permit pim any 224.0.0.2 dscp 32
```

Beginning in privileged EXEC mode, follow these steps to create a Layer 2 MAC ACL for non-IP traffic:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**mac access-list extended** *name*	Create a Layer 2 MAC ACL by specifying the name of the list. After entering this command, the mode changes to extended MAC ACL configuration.
Step 4	{**permit** | **deny**} {**host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*]	Specify the type of traffic to permit or deny if the conditions are matched, entering the command as many times as necessary. • For *src-MAC-addr*, enter the MAC address of the host from which the packet is being sent. You specify this by using the hexadecimal format (H.H.H), by using the **any** keyword as an abbreviation for *source* 0.0.0, *source-wildcard* ffff.ffff.ffff, or by using the **host** keyword for *source* 0.0.0. • For *mask,* enter the wildcard bits by placing ones in the bit positions that you want to ignore. • For *dst-MAC-addr*, enter the MAC address of the host to which the packet is being sent. You specify this by using the hexadecimal format (H.H.H), by using the **any** keyword as an abbreviation for *source* 0.0.0, *source-wildcard* ffff.ffff.ffff, or by using the **host** keyword for *source* 0.0.0. • (Optional) For *type mask*, specify the Ethertype number of a packet with Ethernet II or SNAP encapsulation to identify the protocol of the packet. For *type*, the range is from 0 to 65535, typically specified in hexadecimal. For *mask*, enter the *don't care* bits applied to the Ethertype before testing for a match. **Note** When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show access-lists** [*access-list-number* | *access-list-name*]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an access list, use the **no mac access-list extended** *access-list-name* global configuration command.

This example shows how to create a Layer 2 MAC ACL with two permit statements. The first statement allows traffic from the host with MAC address 0001.0000.0001 to the host with MAC address 0002.0000.0001. The second statement allows only Ethertype XNS-IDP traffic from the host with MAC address 0001.0000.0002 to the host with MAC address 0002.0000.0002.

```
Switch(config)# mac access-list extended maclist1
Switch(config-ext-macl)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
Switch(config-ext-macl)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
! (Note: all other access implicitly denied)
```

Classifying Traffic on a Physical-Port Basis by Using Class Maps

You use the **class-map** global configuration command to name and to isolate a specific traffic flow (or class) from all other traffic. The class map defines the criteria to use to match against a specific traffic flow to further classify it. Match statements can include criterion such as an ACL, IP precedence values, or DSCP values. The match criterion is defined with one match statement entered within the class-map configuration mode.

You cannot configure both port-based classification and VLAN-based classification at the same time.

Note You can also create class-maps during policy map creation by using the **class** policy-map configuration command. For more information, see the "Classifying, Policing, and Marking Traffic by Using Policy Maps" section on page 28-44.

Beginning in privileged EXEC mode, follow these steps to create a class map and to define the match criterion to classify traffic on a physical-port basis:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] or **access-list** *access-list-number* {**deny** | **permit**} *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] or **mac access-list extended** *name* {**permit** | **deny**} {**host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*]	Create an IP standard or extended ACL for IP traffic or a Layer 2 MAC ACL for non-IP traffic, repeating the command as many times as necessary. For more information, see the "Classifying Traffic by Using ACLs" section on page 28-36. **Note** When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.
Step 4	**class-map** [**match-all** | **match-any**] *class-map-name*	Create a class map, and enter class-map configuration mode. By default, no class maps are defined. • (Optional) Use the **match-all** keyword to perform a logical-AND of all matching statements under this class map. All match criteria in the class map must be matched. • (Optional) Use the **match-any** keyword to perform a logical-OR of all matching statements under this class map. One or more match criteria must be matched. • For *class-map-name*, specify the name of the class map. If neither the **match-all** or **match-any** keyword is specified, the default is **match-all**. **Note** Because only one **match** command per class map is supported, the **match-all** and **match-any** keywords function the same.

	Command	Purpose
Step 5	**match** {**access-group** *acl-index-or-name* | **ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*}	Define the match criterion to classify traffic. By default, no match criterion is defined. Only one match criterion per class map is supported, and only one ACL per class map is supported. • For **access-group** *acl-index-or-name,* specify the number or name of the ACL created in Step 3. • For **ip dscp** *dscp-list*, enter a list of up to eight IP DSCP values to match against incoming packets. Separate each value with a space. The range is 0 to 63. • For **ip precedence** *ip-precedence-list*, enter a list of up to eight IP-precedence values to match against incoming packets. Separate each value with a space. The range is 0 to 7. **Note** The only match criterion in a policy map that can be attached to an egress interface is the **match ip dscp** *dscp-list* class-map configuration command.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show class-map**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an existing class map, use the **no class-map** [**match-all** | **match-any**] *class-map-name* global configuration command. To remove a match criterion, use the **no match** {**access-group** *acl-index-or-name* | **ip dscp** | **ip precedence**} class-map configuration command.

This example shows how to configure the class map called *class1*. The *class1* has one match criterion, which is access list 103. It permits traffic that matches a DSCP value of 10 from any host to any destination.

```
Switch(config)# access-list 103 permit any any dscp 10
Switch(config)# class-map class1
Switch(config-cmap)# match access-group 103
Switch(config-cmap)# end
```

This example shows how to create a class map called *class2*, which matches incoming traffic with DSCP values of 10, 11, and 12.

```
Switch(config)# class-map class2
Switch(config-cmap)# match ip dscp 10 11 12
Switch(config-cmap)# end
```

This example shows how to create a class map called *class3*, which matches incoming traffic with IP-precedence values of 5, 6, and 7:

```
Switch(config)# class-map class3
Switch(config-cmap)# match ip precedence 5 6 7
Switch(config-cmap)# end
```

Classifying Traffic on a Per-Port Per-VLAN Basis by Using Class Maps

You use the **class-map** global configuration command to name and to isolate a specific traffic flow (or class) from all other traffic. To further classify the traffic flow, the class map defines the matching criteria to use.

To define packet classification on a per-port per-VLAN basis, follow these guidelines:

- You must use the **match-all** keyword with the **class-map** global configuration command.
- Per-port per-VLAN classification is a per-port feature and does not work on redundant links. It is supported only on an ingress port configured as a trunk or as a static-access port.
- The class map must have two **match** commands in this order: one **match vlan** *vlan-list* class-map configuration command and one **match class-map** *class-map-name* class-map configuration command. The class map specified in the **match class-map** *class-map-name* command must be predefined and cannot contain the **match vlan** *vlan-list* and the **match class-map** *class-map-name* commands.
- You cannot configure both port-based classification and VLAN-based classification at the same time. When you configure the **match vlan** *vlan-list* command, the class map becomes per-port per-VLAN based. If you configure a policy map that contains both port-based and VLAN-based class maps, the switch rejects the policy map when you attach it to an interface.
- With per-port per-VLAN classification, unmatched VLANs are treated similarly to the default class, which means that the unmatched VLANs share the remaining bandwidth from those used by the matched VLAN classes. You cannot modify this default-class behavior. If necessary, you can use VLAN map filters to block these VLANs.
- Within a policy map, when you use the **match vlan** *vlan-list* command, all other class maps must use the **match vlan** *vlan-list* command.
- If you want to modify the VLAN list, first remove the previous configuration in the class map by using the **no match vlan** *vlan-list* command and the **no match class-map** *class-map-name* command. Then reconfigure the class map, and specify the new VLAN list. If the policy map is attached to an interface and you modify the class map by using any other method, the policy map detaches from the interface.

Note When you use the **match vlan** *vlan-list* class-map configuration command, you can enter up to 30 VLAN IDs. When you enter a range of VLANs, such as *10-15*, the VLAN range is counted as two VLAN IDs.

Note You can also create class-maps during policy map creation by using the **class** policy-map configuration command. For more information, see the "Classifying, Policing, and Marking Traffic by Using Policy Maps" section on page 28-44.

Beginning in privileged EXEC mode, follow these steps to create a class map and to define the match criterion to classify traffic on a per-port per-VLAN basis:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**class-map match-any** *class-map-name*	Create a class map, and enter class-map configuration mode. By default, no class maps are defined. • Use the **match-any** keyword to perform a logical-OR of all matching statements under this class map. One or more match criteria must be matched. • For *class-map-name*, specify the name of the class map.

	Command	Purpose
Step 4	**match** {**access-group** *acl-index-or-name* | **ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*}	Define the match criterion to classify traffic. By default, no match criterion is defined. • For **access-group** *acl-index-or-name,* specify the number or name of the ACL. • For **ip dscp** *dscp-list*, enter a list of up to eight IP DSCP values to match against incoming packets. Separate each value with a space. The range is 0 to 63. • For **ip precedence** *ip-precedence-list*, enter a list of up to eight IP-precedence values to match against incoming packets. Separate each value with a space. The range is 0 to 7.
Step 5	**exit**	Return to global configuration mode.
Step 6	**class-map match-all** *class-map-name*	Create a class map, and enter class-map configuration mode. By default, no class maps are defined. • Use the **match-all** keyword to perform a logical-AND of all matching statements under this class map. All match criteria in the class map must be matched. • For *class-map-name*, specify the name of the class map created in Step 3.
Step 7	**match vlan** *vlan-list*	Define the match criterion to classify traffic. By default, no match criterion is defined. For *vlan-list*, specify a list of VLANs to match against incoming packets. You can enter up to 30 VLAN IDs. Use a hyphen for a range of VLANs; the VLAN range is counted as two VLAN IDs. Use a space to separate individual VLANs. The range is 1 to 4094. You can enter only one **match vlan** command, and you must enter it before the **match class-map** command.
Step 8	**match class-map** *class-map-name*	Specify the name of the class map created in Step 3.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show class-map**	Verify your entries.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an existing class map, use the **no class-map** [**match-all** | **match-any**] *class-map-name* global configuration command. To remove a match criterion, use the **no match** {**access-group** *acl-index-or-name* | **ip dscp** | **ip precedence**} class-map configuration command.

This example shows how to configure a class map called *dscp_class* whose match criterion is to match IP DSCP 9. A second class map, called *vlan_class*, matches traffic on VLANs 10, 20 to 30, and 40 to class map *dscp_class*:

```
Switch(config)# class-map match-any dscp_class
Switch(config-cmap)# match ip dscp 9
Switch(config-cmap)# exit
Switch(config)# class-map match-all vlan_class
Switch(config-cmap)# match vlan 10 20-30 40
Switch(config-cmap)# match class-map dscp_class
Switch(config-cmap)# exit
```

Classifying, Policing, and Marking Traffic by Using Policy Maps

A policy map specifies which traffic class to act on. Actions can include trusting the CoS, DSCP, or IP precedence values in the traffic class; setting a specific CoS, DSCP, or IP precedence value in the traffic class; and specifying the traffic bandwidth limitations for each matched traffic class (policer) and the action to take (marking) when the traffic is out of profile.

A policy map also has these characteristics:

- A policy map can contain multiple class statements, each with different match criteria and policers.
- A separate policy-map class can exist for each type of traffic received through an interface.
- A policy-map trust state supersedes an interface trust state.

Follow these guidelines when configuring policy maps:

- Only one policy map per interface per direction is supported. You can apply the same policy map to multiple interfaces and directions.
- If you configure the IP-precedence-to-DSCP map by using the **mls qos map ip-prec-dscp** *dscp1...dscp8* global configuration command, the settings only affect packets on ingress interfaces that are configured to trust the IP precedence value. In a policy map, if you set the packet IP precedence value to a new value by using the **set ip precedence** *new-precedence* policy-map class configuration command, the egress DSCP value is not affected by the IP-precedence-to-DSCP map. If you want to egress DSCP value to be different than the ingress value, use the **set dscp** *new-dscp* policy-map class configuration command.
- When you apply a policy map defined by the **policy-map** global configuration command to the output of an interface or remove the policy map and interface association, the interface goes down. To re-enable the interface, use the **shutdown** and then the **no shutdown** interface configuration commands.

Beginning in privileged EXEC mode, follow these steps to create a policy map:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] or **access-list** *access-list-number* {**deny** | **permit**} *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] or **mac access-list extended** *access-list name* {**permit** | **deny**} {*source-MAC-addr mask* | **any** | **host**} {*destination-MAC-addr mask* | **any** | **host**} [*ethertype*]	Create an IP standard or extended ACL for IP traffic or a Layer 2 MAC ACL for non-IP traffic, repeating the command as many times as necessary. For more information, see the "Classifying Traffic by Using ACLs" section on page 28-36. **Note** When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.

	Command	Purpose
Step 4	**class-map** [**match-all** | **match-any**] *class-map-name*	Create a class map to classify traffic as necessary. For more information, see the "Classifying Traffic on a Physical-Port Basis by Using Class Maps" section on page 28-40 and the "Classifying Traffic on a Per-Port Per-VLAN Basis by Using Class Maps" section on page 28-41.
Step 5	**mls qos cos policy-map**	(Optional) Define the CoS value of a port in a policy map. When you enter this command, you must also enter the **trust dscp** policy-map configuration command in Step 8 and the **set cos** *new-cos* policy-map configuration command in Step 9.
Step 6	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode. By default, no policy maps are defined. The default behavior of a policy map is to set the DSCP to 0 if the packet is an IP packet and to set the CoS to 0 if the packet is tagged. No policing is performed.
Step 7	**class** *class-map-name*	Define a traffic classification, and enter policy-map class configuration mode. By default, no policy map class-maps are defined. If a traffic class has already been defined by using the **class-map** global configuration command, specify its name for *class-map-name* in this command.

	Command	Purpose
Step 8	**trust** [**cos** | **dscp** | **ip-precedence**]	Configure the trust state, which selects the value that QoS uses as the source of the internal DSCP value. **Note** This command is mutually exclusive with the **set** command within the same policy map. If you enter the **trust** command, then skip Step 7. By default, the port is not trusted. If no keyword is specified when the command is entered, the default is **dscp**. The keywords have these meanings: • **cos—**QoS derives the internal DSCP value by using the received or default port CoS value and the CoS-to-DSCP map. • **dscp—**QoS derives the internal DSCP value by using the DSCP value from the ingress packet. For non-IP packets that are tagged, QoS derives the internal DSCP value by using the received CoS value; for non-IP packets that are untagged, QoS derives the internal DSCP value by using the default port CoS value. In either case, the internal DSCP value is derived from the CoS-to-DSCP map. **Note** If you use the **mls qos cos policy-map** global configuration command, you must use the **dscp** keyword. • **ip-precedence—**QoS derives the internal DSCP value by using the IP precedence value from the ingress packet and the IP-precedence-to-DSCP map. For non-IP packets that are tagged, QoS derives the internal DSCP value by using the received CoS value; for non-IP packets that are untagged, QoS derives the internal DSCP value by using the default port CoS value. In either case, the internal DSCP value is derived from the CoS-to-DSCP map. For more information, see the "Configuring the CoS-to-DSCP Map" section on page 28-52.
Step 9	**set** {**cos** *new-cos* | **dscp** *new-dscp* | **ip precedence** *new-precedence*}	Classify IP traffic by setting a new value in the packet. • For **cos** *new-cos,* enter a new CoS value to be assigned to the classified traffic. The range is 0 to 7. **Note** If you use the **mls qos cos policy-map** global configuration command, you must use the **cos** *new-cos* keyword. • For **dscp** *new-dscp*, enter a new DSCP value to be assigned to the classified traffic. The range is 0 to 63. • For **ip precedence** *new-precedence,* enter a new IP-precedence value to be assigned to the classified traffic. The range is 0 to 7.

	Command	Purpose
Step 10	**police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}]	Define a policer for the classified traffic. You can configure up to 128 policers on ingress Gigabit-capable Ethernet ports, up to 8 policers on ingress 10/100 Ethernet ports, and up to 8 policers on egress ports. • For *rate-bps,* specify average traffic rate in bits per second (bps). The range is 8000 to 1000000000. • For *burst-byte,* specify the normal burst size in bytes. The range is 8000 to 20000000. **Note** Although the command-line help strings show a large range of values, the *rate-bps* option cannot exceed the configured port speed, and the *burst-byte* option cannot exceed 2000000 bytes. If you enter a larger value, the switch rejects the policy map when you attach it to an interface. • (Optional) Specify the action to take when the rates are exceeded. Use the **exceed-action drop** keywords to drop the packet. Use the **exceed-action policed-dscp-transmit** keywords to mark down the DSCP value (by using the policed-DSCP map) and send the packet. For more information, see the "Configuring the Policed-DSCP Map" section on page 28-54**.**
Step 11	**exit**	Return to policy map configuration mode.
Step 12	**exit**	Return to global configuration mode.
Step 13	**interface** *interface-id*	Specify the interface to attach to the policy map, and enter interface configuration mode. Valid interfaces include physical interfaces.
Step 14	**service-policy** {**input** *policy-map-name* | **output** *policy-map-name*}	Apply a policy map to the input or output of a particular interface. Only one policy map per interface per direction is supported. • Use **input** *policy-map-name* to apply the specified policy-map to the input of an interface. • Use **output** *policy-map-name* to apply the specified policy-map to the output of an interface. You cannot use the **service-policy** interface configuration command to attach policy maps that contain these elements to an egress interface: • **set** or **trust** policy-map class configuration commands. Instead, you can use the **police** policy-map class configuration command to mark down (reduce) the DSCP value at the egress interface. • Access control list (ACL) classification. • Per-port per-VLAN classification. The only match criterion in a policy map that can be attached to an egress interface is the **match ip dscp** *dscp-list* class-map configuration command. Per-port per-VLAN policing is not supported on routed ports or on virtual (logical) interfaces.

	Command	Purpose
Step 15	**end**	Return to privileged EXEC mode.
Step 16	**show policy-map** [*policy-map-name* [**class** *class-name*]]	Verify your entries.
Step 17	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an existing policy map, use the **no policy-map** *policy-map-name* global configuration command. To delete an existing class map, use the **no class** *class-map-name* policy-map configuration command. To use the DSCP-to-CoS map to define the CoS value, use the **no mls qos cos policy-map** global configuration command. To return to the default trust state, use the **no trust** [**cos** | **dscp** | **ip-precedence**] policy-map configuration command. To remove an assigned CoS, DSCP, or IP precedence value, use the **no set** {**cos** *new-cos* | **dscp** *new-dscp* | **ip precedence** *new-precedence*} policy-map configuration command. To remove an existing policer, use the **no police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}] policy-map configuration command. To remove the policy map and interface association, use the **no service-policy** {**input** *policy-map-name* | **output** *policy-map-name*} interface configuration command.

This example shows how to create a policy map and attach it to an ingress interface. In the configuration, the IP standard ACL permits traffic from network 10.1.0.0. For traffic matching this classification, the DSCP value in the incoming packet is trusted. If the matched traffic exceeds an average traffic rate of 48000 bps and a normal burst size of 8000 bytes, its DSCP is marked down (based on the policed-DSCP map) and sent:

```
Switch(config)# access-list 1 permit 10.1.0.0 0.0.255.255
Switch(config)# class-map ipclass1
Switch(config-cmap)# match access-group 1
Switch(config-cmap)# exit
Switch(config)# policy-map flow1t
Switch(config-pmap)# class ipclass1
Switch(config-pmap-c)# trust dscp
Switch(config-pmap-c)# police 48000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# service-policy input flow1t
```

This example shows how to create a policy map and attach it to an ingress interface. In the configuration, the IP extended ACL permits TCP traffic with an IP precedence of 4 from any host destined for the host at 224.0.0.5. For traffic matching this classification, the DSCP value in the incoming packet is set to 63.

```
Switch(config)# access-list 104 permit tcp any host 224.0.0.5 precedence 4
Switch(config)# class-map ipclass1
Switch(config-cmap)# match access-group 104
Switch(config-cmap)# exit
Switch(config)# policy-map ip104
Switch(config-pmap)# class ipclass1
Switch(config-pmap-c)# set dscp 63
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# service-policy input ip104
```

This example shows how to create a Layer 2 MAC ACL with two permit statements and attach it to an ingress interface. The first permit statement allows traffic from the host with MAC address 0001.0000.0001 destined for the host with MAC address 0002.0000.0001. The second permit statement allows only Ethertype XNS-IDP traffic from the host with MAC address 0001.0000.0002 destined for the host with MAC address 0002.0000.0002.

```
Switch(config)# mac access-list extended maclist1
Switch(config-ext-mac)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
Switch(config-ext-mac)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
Switch(config-ext-mac)# exit
Switch(config)# mac access-list extended maclist2
Switch(config-ext-mac)# permit 0001.0000.0003 0.0.0 0002.0000.0003 0.0.0
Switch(config-ext-mac)# permit 0001.0000.0004 0.0.0 0002.0000.0004 0.0.0 aarp
Switch(config-ext-mac)# exit
Switch(config)# class-map macclass1
Switch(config-cmap)# match access-group maclist1
Switch(config-cmap)# exit
Switch(config)# class-map macclass2
Switch(config-cmap)# match access-group maclist2
Switch(config-cmap)# exit
Switch(config)# policy-map macpolicy1
Switch(config-pmap)# class macclass1
Switch(config-pmap-c)# set dscp 63
Switch(config-pmap-c)# exit
Switch(config-pmap)# class macclass2
Switch(config-pmap-c)# set dscp 45
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# mls qos trust cos
Switch(config-if)# service-policy input macpolicy1
```

This example shows how to create a policy map that contains per-port per-VLAN classification and attach it to an ingress interface. A class map, called *vlan_class*, matches traffic received on VLANs 10, 20 to 30, and 40 that contains IP DSCP 9 (defined in class map *dscp_class*). If the specified average traffic rates and the burst sizes are exceeded, the switch drops the packet.

```
Switch(config)# class-map match-any dscp_class
Switch(config-cmap)# match ip dscp 9
Switch(config-cmap)# exit
Switch(config)# class-map match-all vlan_class
Switch(config-cmap)# match vlan 10 20-30 40
Switch(config-cmap)# match class-map dscp_class
Switch(config-cmap)# exit
Switch(config)# policy-map policymap2
Switch(config-pmap)# class vlan_class
Switch(config-pmap-c)# police 80000 8000 exceed-action drop
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# service-policy input policymap2
```

This example shows how to create a policy map that defines the CoS value for a port and how to attach it to an ingress interface. A class map, called *class1*, matches traffic received on VLANs 10, 20 to 30, and 40.

```
Switch (config)# mls qos cos policy-map
Switch (config)# class-map match-all class1
Switch (config-cmap)# match vlan 10 20-30 40
Switch (config-cmap)# match class-map some_class
Switch (config-cmap)# exit
Switch (config)# policy-map policymap1
Switch (config-pmap)# class class1
Switch (config-pmap-c)# trust dscp
Switch (config-pmap-c)# set cos 3
Switch (config-pmap-c)# exit
Switch (config-pmap)# exit
Switch (config)# interface gigabitethernet0/1
Switch(config-if)# service-policy input policymap1
```

Classifying, Policing, and Marking Traffic by Using Aggregate Policers

By using an aggregate policer, you can create a policer that is shared by multiple traffic classes within the same policy map. However, you cannot use the aggregate policer across different policy maps or interfaces.

Beginning in privileged EXEC mode, follow these steps to create an aggregate policer:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**mls qos aggregate-police** *aggregate-policer-name rate-bps burst-byte* **exceed-action** {**drop** | **policed-dscp-transmit**}	Define the policer parameters that can be applied to multiple traffic classes within the same policy map. By default, no aggregate policer is defined. You can configure up to 128 policers on ingress Gigabit-capable Ethernet ports, up to 8 policers on ingress 10/100 Ethernet ports, and up to 8 policers on egress ports. • For *aggregate-policer-name*, specify the name of the aggregate policer. • For *rate-bps,* specify average traffic rate in bits per second (bps). The range is 8000 to 1000000000. • For *burst-byte,* specify the normal burst size in bytes. The range is 8000 to 2000000. **Note** Although the command-line help strings show a large range of values, the *rate-bps* option cannot exceed the configured port speed, and the *burst-byte* option cannot exceed 2000000 bytes. If you enter a larger value, the switch rejects the policy map when you attach it to an interface. • (Optional) Specify the action to take when the rates are exceeded. Use the **exceed-action drop** keywords to drop the packet. Use the **exceed-action policed-dscp-transmit** keywords to mark down the DSCP value (by using the policed-DSCP map) and to send the packet. For more information, see the "Configuring the Policed-DSCP Map" section on page 28-54**.**
Step 4	**class-map** [**match-all** | **match-any**] *class-map-name*	Create a class map to classify traffic as necessary. For more information, see the "Classifying Traffic on a Physical-Port Basis by Using Class Maps" section on page 28-40 and the "Classifying Traffic on a Per-Port Per-VLAN Basis by Using Class Maps" section on page 28-41.
Step 5	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode. For more information, see the "Classifying, Policing, and Marking Traffic by Using Policy Maps" section on page 28-44.

	Command	Purpose
Step 6	**class** *class-map-name*	Define a traffic classification, and enter policy-map class configuration mode. By default, no policy map class-maps are defined. If a traffic class has already been defined by using the **class-map** global configuration command, specify its name for *class-map-name* in this command.
Step 7	**police aggregate** *aggregate-policer-name*	Apply an aggregate policer to multiple classes in the same policy map. For *aggregate-policer-name*, enter the name specified in Step 3.
Step 8	**exit**	Return to global configuration mode.
Step 9	**interface** *interface-id*	Specify the interface to attach to the policy map, and enter interface configuration mode. Valid interfaces include physical interfaces.
Step 10	**service-policy** {**input** *policy-map-name* | **output** *policy-map-name*}	Apply a policy map to the input or output of a particular interface. Only one policy map per interface per direction is supported. • Use **input** *policy-map-name* to apply the specified policy-map to the input of an interface. • Use **output** *policy-map-name* to apply the specified policy-map to the output of an interface. You cannot use the **service-policy** interface configuration command to attach policy maps that contain these elements to an egress interface: • **set** or **trust** policy-map class configuration commands. Instead, you can use the **police** policy-map class configuration command to mark down (reduce) the DSCP value at the egress interface. • Access control list (ACL) classification. • Per-port per-VLAN classification. The only match criterion in a policy map that can be attached to an egress interface is the **match ip dscp** *dscp-list* class-map configuration command. Per-port per-VLAN policing is not supported on routed ports or on virtual (logical) interfaces.
Step 11	**end**	Return to privileged EXEC mode.
Step 12	**show mls qos aggregate-policer** [*aggregate-policer-name*]	Verify your entries.
Step 13	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the specified aggregate policer from a policy map, use the **no police aggregate** *aggregate-policer-name* policy map configuration mode. To delete an aggregate policer and its parameters, use the **no mls qos aggregate-policer** *aggregate-policer-name* global configuration command.

This example shows how to create an aggregate policer and attach it to multiple classes within a policy map. In the configuration, the IP ACLs permit traffic from network 10.1.0.0 and from host 11.3.1.1. For traffic coming from network 10.1.0.0, the DSCP in the incoming packets is trusted. For traffic coming

from host 11.3.1.1, the DSCP in the packet is changed to 56. The traffic rate from the 10.1.0.0 network and from host 11.3.1.1 is policed. If the traffic exceeds an average rate of 48000 bps and a normal burst size of 8000 bytes, its DSCP is marked down (based on the policed-DSCP map) and sent. The policy map is attached to an ingress interface.

```
Switch(config)# access-list 1 permit 10.1.0.0 0.0.255.255
Switch(config)# access-list 2 permit 11.3.1.1
Switch(config)# mls qos aggregate-police transmit1 48000 8000 exceed-action
policed-dscp-transmit
Switch(config)# class-map ipclass1
Switch(config-cmap)# match access-group 1
Switch(config-cmap)# exit
Switch(config)# class-map ipclass2
Switch(config-cmap)# match access-group 2
Switch(config-cmap)# exit
Switch(config)# policy-map aggflow1
Switch(config-pmap)# class ipclass1
Switch(config-pmap-c)# trust dscp
Switch(config-pmap-c)# police aggregate transmit1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class ipclass2
Switch(config-pmap-c)# set dscp 56
Switch(config-pmap-c)# police aggregate transmit1
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# service-policy input aggflow1
Switch(config-if)# exit
```

Configuring DSCP Maps

These sections describe how to configure the DSCP maps:

- Configuring the CoS-to-DSCP Map, page 28-52
- Configuring the IP-Precedence-to-DSCP Map, page 28-53
- Configuring the Policed-DSCP Map, page 28-54
- Configuring the DSCP-to-CoS Map, page 28-55
- Configuring the DSCP-to-DSCP-Mutation Map, page 28-57

All the maps, except the DSCP-to-DSCP-mutation map, are globally defined and are applied to all ports.You can have multiple DSCP-to-DSCP-mutation maps and apply them to different Gigabit-capable Ethernet ports. However, on 10/100 Ethernet ports, you can attach only one DSCP-to-DSCP-mutation map to a group of twelve ports.

Configuring the CoS-to-DSCP Map

You use the CoS-to-DSCP map to map CoS values in incoming packets to a DSCP value that QoS uses internally to represent the priority of the traffic.

Table 28-6 shows the default CoS-to-DSCP map.

Table 28-6 Default CoS-to-DSCP Map

CoS Value	DSCP Value
0	0
1	8
2	16
3	24
4	32
5	40
6	48
7	56

If these values are not appropriate for your network, you need to modify them.

Beginning in privileged EXEC mode, follow these steps to modify the CoS-to-DSCP map:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos map cos-dscp** *dscp1...dscp8*	Modify the CoS-to-DSCP map. For *dscp1...dscp8*, enter eight DSCP values that correspond to CoS values 0 to 7. Separate each DSCP value with a space. The range is 0 to 63.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mls qos maps cos-dscp**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default map, use the **no mls qos map cos-dscp** global configuration command.

This example shows how to modify and display the CoS-to-DSCP map:

```
Switch# configure terminal
Switch(config)# mls qos map cos-dscp 10 15 20 25 30 35 40 45
Switch(config)# end
Switch# show mls qos maps cos-dscp

Cos-dscp map:
        cos:   0  1  2  3  4  5  6  7
      --------------------------------
       dscp:   10 15 20 25 30 35 40 45
```

Configuring the IP-Precedence-to-DSCP Map

You use the IP-precedence-to-DSCP map to map IP precedence values in incoming packets to a DSCP value that QoS uses internally to represent the priority of the traffic.

Table 28-7 shows the default IP-precedence-to-DSCP map:

Table 28-7 Default IP-Precedence-to-DSCP Map

IP Precedence Value	DSCP Value
0	0
1	8
2	16
3	24
4	32
5	40
6	48
7	56

If these values are not appropriate for your network, you need to modify them.

Beginning in privileged EXEC mode, follow these steps to modify the IP-precedence-to-DSCP map:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos map ip-prec-dscp** *dscp1...dscp8*	Modify the IP-precedence-to-DSCP map. For *dscp1...dscp8*, enter eight DSCP values that correspond to the IP precedence values 0 to 7. Separate each DSCP value with a space. The range is 0 to 63.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mls qos maps ip-prec-dscp**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default map, use the **no mls qos map ip-prec-dscp** global configuration command.

This example shows how to modify and display the IP-precedence-to-DSCP map:

```
Switch# configure terminal
Switch(config)# mls qos map ip-prec-dscp 10 15 20 25 30 35 40 45
Switch(config)# end
Switch# show mls qos maps ip-prec-dscp

IpPrecedence-dscp map:
     ipprec:   0  1  2  3  4  5  6  7
     --------------------------------
       dscp:   10 15 20 25 30 35 40 45
```

Configuring the Policed-DSCP Map

You use the policed-DSCP map to mark down a DSCP value to a new value as the result of a policing and marking action.

The default policed-DSCP map is a null map, which maps an incoming DSCP value to the same DSCP value.

Beginning in privileged EXEC mode, follow these steps to modify the policed-DSCP map:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos map policed-dscp** *dscp-list* **to** *mark-down-dscp*	Modify the policed-DSCP map. • For *dscp-list*, enter up to eight DSCP values, with each value separated by a space. Then enter the **to** keyword. • For *mark-down-dscp*, enter the corresponding policed (marked down) DSCP value. The range is 0 to 63.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mls qos maps policed-dscp**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default map, use the **no mls qos map policed-dscp** global configuration command.

This example shows how to map DSCP values 50 to 57 to a marked-down DSCP value of 0:

```
Switch# configure terminal
Switch(config)# mls qos map policed-dscp 50 51 52 53 54 55 56 57 to 0
Switch(config)# end
Switch# show mls qos maps policed-dscp
Policed-dscp map:
     d1 :  d2 0  1  2  3  4  5  6  7  8  9
     ----------------------------------------
      0 :    00 01 02 03 04 05 06 07 08 09
      1 :    10 11 12 13 14 15 16 17 18 19
      2 :    20 21 22 23 24 25 26 27 28 29
      3 :    30 31 32 33 34 35 36 37 38 39
      4 :    40 41 42 43 44 45 46 47 48 49
      5 :    00 00 00 00 00 00 00 00 58 59
      6 :    60 61 62 63
```

Note In the policed-DSCP map, the marked-down DSCP values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the original DSCP; the d2 row specifies the least-significant digit of the original DSCP. The intersection of the d1 and d2 values gives the marked-down value. For example, an original DSCP value of 53 corresponds to a marked-down DSCP value of 0.

Configuring the DSCP-to-CoS Map

You use the DSCP-to-CoS map to generate a CoS value, which is used to select one of the four egress queues.

Table 28-8 shows the default DSCP-to-CoS map.

Table 28-8 Default DSCP-to-CoS Map

DSCP Value	CoS Value
0–7	0
8–15	1

Table 28-8 Default DSCP-to-CoS Map (continued)

DSCP Value	CoS Value
16–23	2
24–31	3
32–39	4
40–47	5
48–55	6
56–63	7

If these values are not appropriate for your network, you need to modify them.

Beginning in privileged EXEC mode, follow these steps to modify the DSCP-to-CoS map:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos map dscp-cos** *dscp-list* **to** *cos*	Modify the DSCP-to-CoS map. • For *dscp-list*, enter up to eight DSCP values, with each value separated by a space. Then enter the **to** keyword. The range is 0 to 63. • For *cos*, enter the CoS value to which the DSCP values correspond. The range is 0 to 7.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mls qos maps dscp-to-cos**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default map, use the **no mls qos map dscp-cos** global configuration command.

This example shows how to map DSCP values 0, 8, 16, 24, 32, 40, 48, and 50 to CoS value 0:

```
Switch# configure terminal
Switch(config)# mls qos map dscp-cos 0 8 16 24 32 40 48 50 to 0
Switch(config)# end
Switch# show mls qos maps dscp-cos
Dscp-cos map:
     d1 :  d2 0  1  2  3  4  5  6  7  8  9
     ---------------------------------------
      0 :    00 00 00 00 00 00 00 00 00 01
      1 :    01 01 01 01 01 01 00 02 02 02
      2 :    02 02 02 02 00 03 03 03 03 03
      3 :    03 03 00 04 04 04 04 04 04 04
      4 :    00 05 05 05 05 05 05 05 00 06
      5 :    00 06 06 06 06 06 07 07 07 07
      6 :    07 07 07 07
```

Note In the DSCP-to-CoS map, the CoS values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the DSCP; the d2 row specifies the least-significant digit of the DSCP. The intersection of the d1 and d2 values gives the CoS value. For example, in the DSCP-to-CoS map, a DSCP value of 08 corresponds to a CoS value of 0.

Configuring the DSCP-to-DSCP-Mutation Map

You apply the DSCP-to-DSCP-mutation map to a port at the boundary of a QoS administrative domain. If the two domains have different DSCP definitions between them, use the DSCP-to-DSCP-mutation map to translate a set of DSCP values to match the definition of the other domain.

The default DSCP-to-DSCP-mutation map is a null map, which maps an incoming DSCP value to the same DSCP value.

Beginning in privileged EXEC mode, follow these steps to modify the DSCP-to-DSCP-mutation map:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos map dscp-mutation** *dscp-mutation-name in-dscp* **to** *out-dscp*	Modify the DSCP-to-DSCP-mutation map. • For *dscp-mutation-name*, enter the mutation map name. You can create more than one map by specifying a new name. • For *in-dscp*, enter up to eight DSCP values, with each value separated by a space. Then enter the **to** keyword. • For *out-dscp*, enter a single DSCP value. The range is 0 to 63.
Step 3	**interface** *interface-id*	Specify the interface to which to attach the map, and enter interface configuration mode. Valid interfaces include physical interfaces.
Step 4	**mls qos trust dscp**	Configure the ingress port as a DSCP-trusted port.
Step 5	**mls qos dscp-mutation** *dscp-mutation-name*	Apply the map to the specified ingress DSCP-trusted port. For *dscp-mutation-name*, enter the mutation map name specified in Step 2. You can apply the map to different Gigabit-capable Ethernet ports. However, on 10/100 Ethernet ports, you can attach only one DSCP-to-DSCP-mutation map to a group of twelve ports. For example, Fast Ethernet ports 1 to 12 are a group, Fast Ethernet ports 13 to 24 are a group, Gigabit Ethernet port 1 is a group, and Gigabit Ethernet port 2 is a group. When applying a mutation map to any port in a group, all ports in the same group are automatically configured with the same map.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show mls qos maps dscp-mutation**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default map, use the **no mls qos map dscp-mutation** *dscp-mutation-name* global configuration command.

This example shows how to define the DSCP-to-DSCP-mutation map. All the entries that are not explicitly configured are not modified (remain as specified in the null map):

```
Switch# configure terminal
Switch(config)# mls qos map dscp-mutation mutation1 1 2 3 4 5 6 7 to 0
Switch(config)# mls qos map dscp-mutation mutation1 8 9 10 11 12 13 to 10
Switch(config)# mls qos map dscp-mutation mutation1 20 21 22 to 20
```

```
Switch(config)# mls qos map dscp-mutation mutation1 30 31 32 33 34 to 30
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# mls qos trust dscp
Switch(config-if)# mls qos dscp-mutation mutation1
Switch(config-if)# end
Switch# show mls qos maps dscp-mutation mutation1
Dscp-dscp mutation map:
   mutation1:
     d1 :  d2 0  1  2  3  4  5  6  7  8  9
     ---------------------------------------
      0 :    00 00 00 00 00 00 00 00 10 10
      1 :    10 10 10 10 14 15 16 17 18 19
      2 :    20 20 20 23 24 25 26 27 28 29
      3 :    30 30 30 30 30 35 36 37 38 39
      4 :    40 41 42 43 44 45 46 47 48 49
      5 :    50 51 52 53 54 55 56 57 58 59
      6 :    60 61 62 63
```

Note In the DSCP-to-DSCP-mutation map, the mutated values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the original DSCP; the d2 row specifies the least-significant digit of the original DSCP. The intersection of the d1 and d2 values gives the mutated value. For example, a DSCP value of 12 corresponds to a mutated value of 10.

Configuring Egress Queues on Gigabit-Capable Ethernet Ports

This section describes how to configure the egress queues on Gigabit-capable Ethernet ports. For information on configuring 10/100 Ethernet ports, see "Configuring Egress Queues on 10/100 Ethernet Ports" section on page 28-65.

Depending on the complexity of your network and your QoS solution, you might need to perform all of the tasks in the next sections. You will need to make decisions about these characteristics:

- Which packets are assigned (by CoS value) to each queue?
- How much of the available buffer space (limit) is allotted to each queue?
- What drop percentage thresholds apply to each queue and which DSCP values map to each threshold?
- Is one of the queues the expedite (high-priority) egress queue?
- How much of the available bandwidth is allotted to each queue?

These sections contain this configuration information:

- Mapping CoS Values to Select Egress Queues, page 28-59
- Configuring the Egress Queue Size Ratios, page 28-59
- Configuring Tail-Drop Threshold Percentages, page 28-60
- Configuring WRED Drop Thresholds Percentages, page 28-62
- Configuring the Egress Expedite Queue, page 28-63
- Allocating Bandwidth among Egress Queues, page 28-64

Mapping CoS Values to Select Egress Queues

Beginning in privileged EXEC mode, follow these steps to map CoS ingress values to select one of the egress queues:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress Gigabit-capable Ethernet interface, and enter interface configuration mode.
Step 4	**wrr-queue cos-map** *queue-id cos1 ... cos8*	Map assigned CoS values to select one of the egress queues. The default map has these values: CoS value 0, 1 selects queue 1. CoS value 2, 3 selects queue 2. CoS value 4, 5 selects queue 3. CoS value 6, 7 selects queue 4. • For *queue-id*, specify the ID of the egress queue. The range is 1 to 4, where 4 can be configured as the expedite queue. For more information, see the "Configuring the Egress Expedite Queue" section on page 28-63. • For *cos1 ... cos8*, specify the CoS values that select a queue. Enter up to eight CoS values. Separate each value with a space. The range is 0 to 7.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show mls qos interface queueing**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the default CoS-to-egress-queue map, use the **no wrr-queue cos-map** interface configuration command.

This example shows how to map CoS values 6 and 7 to queue 1, 4 and 5 to queue 2, 2 and 3 to queue 3, 0 and 1 to queue 4.

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# wrr-queue cos-map 1 6 7
Switch(config-if)# wrr-queue cos-map 2 4 5
Switch(config-if)# wrr-queue cos-map 3 2 3
Switch(config-if)# wrr-queue cos-map 4 0 1
```

Configuring the Egress Queue Size Ratios

Beginning in privileged EXEC mode, follow these steps to configure the egress queue size ratios:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.

	Command	Purpose
Step 3	**interface** *interface-id*	Specify the egress Gigabit-capable Ethernet interface, and enter interface configuration mode.
Step 4	**wrr-queue queue-limit** *weight1 weight2 weight3 weight4*	Configure the egress queue size ratios. The defaults weights are 25 (1/4 of the buffer size is allocated to each queue). For *weight1*, *weight2*, *weight3*, and *weight4*, specify a weight from 1 to 100. Separate each value with a space. The relative size difference in the numbers show the relative differences in the queue sizes. When you enter this command, the queue is temporarily shutdown during the hardware reconfiguration, and the switch drops newly arrived packets to this queue.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show mls qos interface buffers**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default weights, use the **no wrr-queue queue-limit** interface configuration command.

This example shows how to configure the size ratio of the four queues. The ratio of the size allocated for each queue is 1/10, 2/10, 3/10, and 4/10 for queues 1, 2, 3, and 4. (Queue 4 is four times larger than queue 1, twice as large as queue 2, and 1.33 times as large as queue 3.)

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# wrr-queue queue-limit 1 2 3 4
```

Configuring Tail-Drop Threshold Percentages

Tail drop is the default congestion-avoidance technique on Gigabit-capable Ethernet ports. With tail drop, packets are queued until the thresholds are exceeded. For example, all packets with DSCPs assigned to the first threshold are dropped until the threshold is no longer exceeded. However, packets assigned to a second threshold continue to be queued and sent as long as the second threshold is not exceeded.

You modify the DSCP-to-threshold map to determine which DSCPs are mapped to which threshold ID by using the **wrr-queue dscp-map** interface configuration command. By default, all DSCPs are mapped to threshold 1, and when this threshold is exceeded, all the packets are dropped.

If you use tail-drop thresholds, you cannot use WRED, and vice versa.

Beginning in privileged EXEC mode, follow these steps to configure the tail-drop threshold percentage values on Gigabit-capable Ethernet ports:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress Gigabit-capable Ethernet interface, and enter interface configuration mode.

	Command	Purpose
Step 4	**wrr-queue threshold** *queue-id threshold-percentage1 threshold-percentage2*	Configure tail-drop threshold percentages on each egress queue. The default threshold is 100 percent for thresholds 1 and 2. • For *queue-id*, specify the ID of the egress queue. The range is 1 to 4. • For *threshold-percentage1 threshold-percentage2*, specify the tail-drop threshold percentage values. Separate each value with a space. The range is 1 to 100.
Step 5	**exit**	Return to global configuration mode.
Step 6	**interface** *interface-id*	Specify the ingress Gigabit-capable Ethernet interface, and enter interface configuration mode.
Step 7	**wrr-queue dscp-map** *threshold-id dscp1 ... dscp8*	Map DSCP values to the tail-drop thresholds of the egress queues. By default, all DSCP values are mapped to threshold 1. • For *threshold-id*, specify the threshold ID of the queue. The range is 1 to 2. • For *dscp1 ... dscp8*, specify the DSCP values that are mapped to the threshold ID. Enter up to eight DSCP values per command. Separate each value with a space. The range is 0 to 63.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show running-config** or **show mls qos interface** *interface-id* **queueing**	Verify the DSCP-to-threshold map.
Step 10	**show mls qos interface buffers**	Verify the thresholds.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default thresholds, use the **no wrr-queue threshold** *queue-id* interface configuration command. To return to the default DSCP-to-threshold map, use the **no wrr-queue dscp-map** [*threshold-id*] interface configuration command.

This example shows how to configure the tail-drop queue threshold values for queue 1 to 10 percent and 100 percent, for queue 2 to 40 percent and 100 percent, for queue 3 to 60 percent and 100 percent, and for queue 4 to 80 percent and 100 percent on the egress interface (Gigabit Ethernet port 1). The ingress interface (Gigabit Ethernet port 2) is configured to trust the DSCP in the incoming packets, to map DSCPs 0, 8, 16, 24, 32, 40, 48, and 56 to threshold 1, and to map DSCPs 10, 20, 30, 40, 50, and 60 to threshold 2:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# switchport mode access
Switch(config-if)# wrr-queue threshold 1 10 100
Switch(config-if)# wrr-queue threshold 2 40 100
Switch(config-if)# wrr-queue threshold 3 60 100
Switch(config-if)# wrr-queue threshold 4 80 100
Switch(config-if)# exit
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# mls qos trust dscp
Switch(config-if)# wrr-queue dscp-map 1 0 8 16 24 32 40 48 56
Switch(config-if)# wrr-queue dscp-map 2 10 20 30 40 50 60
```

As a result of this configuration, when queue 1 is filled above 10 percent, packets with DSCPs 0, 8, 16, 24, 32, 40, 48, and 56 are dropped. The same packets are dropped when queue 2 is filled above 40 percent, queue 3 above 60 percent, and queue 4 above 80 percent. When the second threshold (100 percent) is exceeded, all queues drop packets with DSCPs 10, 20, 30, 40, 50, and 60.

Configuring WRED Drop Thresholds Percentages

WRED reduces the chances of tail drop by selectively dropping packets when the output interface begins to show signs of congestion. By dropping some packets early rather than waiting until the queue is full, WRED avoids dropping large numbers of packets at once.

All packets with DSCPs assigned to the first threshold are randomly dropped when the first threshold is exceeded. However, packets with DSCPs assigned to the second threshold continue to be queued and sent as long as the second threshold is not exceeded. Each threshold percentage represents where WRED starts to randomly drop packets. By default, WRED is disabled.

If you use WRED, you cannot use tail-drop thresholds, and vice versa.

Beginning in privileged EXEC mode, follow these steps to configure the WRED drop threshold percentage values on Gigabit-capable Ethernet ports:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress Gigabit-capable Ethernet interface, and enter interface configuration mode.
Step 4	**wrr-queue random-detect max-threshold** *queue-id threshold-percentage1 threshold-percentage2*	Configure WRED drop threshold percentages on each egress queue. The default, WRED is disabled, and no thresholds are configured. • For *queue-id*, specify the ID of the egress queue. The range is 1 to 4, where queue 4 can be configured as the expedite queue. For more information, see the "Configuring the Egress Expedite Queue" section on page 28-63. • For *threshold-percentage1 threshold-percentage2*, specify the threshold percentage values. Separate each value with a space. The range is 1 to 100.
Step 5	**exit**	Return to global configuration mode.
Step 6	**interface** *interface-id*	Specify the ingress Gigabit-capable Ethernet interface, and enter interface configuration mode.
Step 7	**wrr-queue dscp-map** *threshold-id dscp1 ... dscp8*	Map DSCP values to the WRED drop thresholds of the egress queues. By default, all DSCP values are mapped to threshold 1. • For *threshold-id*, specify the threshold ID of the queue. The range is 1 to 2. • For *dscp1 ... dscp8*, specify the DSCP values that are mapped to the threshold ID. Enter up to eight DSCP values per command. Separate each value with a space. The range is 0 to 63.

	Command	Purpose
Step 8	**show running-config** or **show mls qos interface** *interface-id* **queueing**	Verify the DSCP-to-threshold map.
Step 9	**show mls qos interface buffers**	Verify the thresholds.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable WRED, use the **no wrr-queue random-detect max-threshold** *queue-id* interface configuration command. To return to the default DSCP-to-threshold map, use the **no wrr-queue dscp-map** [*threshold-id*] interface configuration command.

This example shows how to configure the WRED queue threshold values for queue 1 to 50 percent and 100 percent, for queue 2 to 70 percent and 100 percent, for queue 3 to 50 percent and 100 percent, and for queue 4 to 70 percent and 100 percent on the egress interface (Gigabit Ethernet port 1). The ingress interface (Gigabit Ethernet port 2) is configured to trust the DSCP in the incoming packets, to map DSCPs 0, 8, 16, 24, 32, 40, 48, and 56 to threshold 1, and to map DSCPs 10, 20, 30, 40, 50, and 60 to threshold 2.

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# wrr-queue random-detect max-threshold 1 50 100
Switch(config-if)# wrr-queue random-detect max-threshold 2 70 100
Switch(config-if)# wrr-queue random-detect max-threshold 3 50 100
Switch(config-if)# wrr-queue random-detect max-threshold 4 70 100
Switch(config-if)# exit
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# mls qos trust dscp
Switch(config-if)# wrr-queue dscp-map 1 0 8 16 24 32 40 48 56
Switch(config-if)# wrr-queue dscp-map 2 10 20 30 40 50 60
```

As a result of this configuration, when the queues 1 and 3 are filled above 50 percent, packets with DSCPs 0, 8, 16, 24, 32, 40, 48, and 56 are randomly dropped. The same packets are randomly dropped when queues 2 and 4 are filled above 70 percent. When the second threshold (100 percent) is exceeded, all queues randomly drop packets with DSCPs 10, 20, 30, 40, 50, and 60.

Configuring the Egress Expedite Queue

You can ensure that certain packets have priority over all others by queuing them in the egress expedite queue. WRR services this queue until it is empty before servicing the other queues.

Beginning in privileged EXEC mode, follow these steps to enable the egress expedite queue:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress Gigabit-capable Ethernet interface, and enter interface configuration mode.

	Command	Purpose
Step 4	**priority-queue out**	Enable the egress expedite queue, which is disabled by default. When you configure this command, the WRR weight and queue size ratios are affected because there is one fewer queue participating in WRR. This means that *weight4* in the **wrr-queue bandwidth** command is ignored (not used in the ratio calculation).
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the egress expedite queue, use the **no priority-queue out** interface configuration command.

Allocating Bandwidth among Egress Queues

You need to specify how much of the available bandwidth is allocated to each queue. The ratio of the weights is the ratio of frequency in which the WRR scheduler dequeues packets from each queue.

Beginning in privileged EXEC mode, follow these steps to allocate bandwidth to each queue:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress Gigabit-capable Ethernet interface, and enter interface configuration mode.
Step 4	**wrr-queue bandwidth** *weight1 weight2 weight3 weight4*	Assign WRR weights to the egress queues. By default, all the weights are set to 25 (1/4 of the bandwidth is allocated to each queue). For *weight1 weight2 weight3 weight4*, enter the ratio, which determines the ratio of the frequency in which the WRR scheduler dequeues packets. Separate each value with a space. The range is 1 to 65536. All four queues participate in the WRR unless the expedite queue (queue 4) is enabled, in which case *weight4* is ignored (not used in the ratio calculation). The expedite queue is a priority queue, and it is serviced until empty before the other queues are serviced. A weight of 1 means that the minimum bandwidth is allocated for that queue. To allocate no bandwidth for a queue, use the **wrr-queue cos-map** interface configuration command so that the available bandwidth is shared among the remaining queues.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show mls qos interface queueing**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default bandwidth setting, use the **no wrr-queue bandwidth** interface configuration command.

This example shows how to configure the weight ratio of the WRR scheduler running on the egress queues. In this example, four queues are used (no expedite queue), and the ratio of the bandwidth allocated for each queue is 1/(1+2+3+4), 2/(1+2+3+4), 3/(1+2+3+4), and 4/(1+2+3+4), which is 1/10, 1/5, 3/10, and 2/5 for queues 1, 2, 3, and 4.

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# wrr-queue bandwidth 1 2 3 4
```

Configuring Egress Queues on 10/100 Ethernet Ports

This section describes how to configure the egress queues on 10/100 Ethernet ports. For information on configuring Gigabit-capable Ethernet ports, see the "Configuring Egress Queues on Gigabit-Capable Ethernet Ports" section on page 28-58.

Depending on the complexity of your network and your QoS solution, you might need to perform all of the tasks in the next sections. You will need to make decisions about these characteristics:

- Which packets are assigned (by CoS value) to each queue?
- How much of the available buffer space is allotted to each queue?
- Is one of the queues the expedite (high-priority) egress queue?
- How much of the available bandwidth is allotted to each queue?

These sections contain this configuration information:

- Mapping CoS Values to Select Egress Queues, page 28-65
- Configuring the Minimum-Reserve Levels, page 28-66
- Configuring the Egress Expedite Queue, page 28-67
- Allocating Bandwidth among Egress Queues, page 28-68

Mapping CoS Values to Select Egress Queues

Beginning in privileged EXEC mode, follow these steps to map CoS ingress values to select one of the egress queues:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress 10/100 Ethernet interface, and enter interface configuration mode.

	Command	Purpose
Step 4	**wrr-queue cos-map** *queue-id cos1 ... cos8*	Map assigned CoS values to select one of the egress queues. Theses are the default map values: CoS value 0, 1 selects queue 1. CoS value 2, 3 selects queue 2. CoS value 4, 5 selects queue 3. CoS value 6, 7 selects queue 4. • For *queue-id*, specify the ID of the egress queue. The range is 1 to 4, where 4 can be configured as the expedite queue. For more information, see the "Configuring the Egress Expedite Queue" section on page 28-67. • For *cos1 ... cos8*, specify the CoS values that select a queue. Enter up to eight CoS values. Separate each value with a space. The range is 0 to 7.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show mls qos interface queueing**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default CoS-to-egress-queue map, use the **no wrr-queue cos-map** interface configuration command.

This example shows how to map CoS values 6 and 7 to queue 1, 4 and 5 to queue 2, 2 and 3 to queue 3, and 0 and 1 to queue 4.

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# wrr-queue cos-map 1 6 7
Switch(config-if)# wrr-queue cos-map 2 4 5
Switch(config-if)# wrr-queue cos-map 3 2 3
Switch(config-if)# wrr-queue cos-map 4 0 1
```

Configuring the Minimum-Reserve Levels

You can configure the buffer size of the minimum-reserve levels on all 10/100 ports and assign the minimum-reserve level to an egress queue on a 10/100 Ethernet port.

Beginning in privileged EXEC mode, follow these steps to configure the egress queue sizes:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.

	Command	Purpose
Step 3	**mls qos min-reserve** *min-reserve-level min-reserve-buffersize*	Configure the buffer size of the minimum-reserve level, if necessary, for all the 10/100 Ethernet ports. By default, the buffer size for all eight minimum-reserve levels is 100 packets. • For *min-reserve-level*, specify the minimum-reserve level number. The range is 1 to 8. • For *min-reserve-buffersize*, specify the buffer size. The range is 10 to 170 packets. When you enter this command, the queue is temporarily shutdown during the hardware reconfiguration, and the switch drops newly arrived packets to this queue.
Step 4	**interface** *interface-id*	Specify the egress 10/100 Ethernet interface, and enter interface configuration mode.
Step 5	**wrr-queue min-reserve** *queue-id min-reserve-level*	Assign a minimum-reserve level number to a particular egress queue. By default, queue 1 selects minimum-reserve level 1, queue 2 selects minimum-reserve level 2, queue 3 selects minimum-reserve level 3, and queue 4 selects minimum-reserve level 4. • For *queue-id*, specify the ID of the egress queue. The range is 1 to 4, where 4 can be configured as the expedite queue. For more information, see the "Configuring the Egress Expedite Queue" section on page 28-67. • For *min-reserve-level*, specify the minimum-reserve level configured in Step 3.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show mls qos interface buffers**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default minimum-reserve buffer size, use the **no mls qos min-reserve** *min-reserve-level* global configuration command. To return to the default queue selection of the minimum-reserve level, use the **no wrr-queue min-reserve** *queue-id* interface configuration command.

This example shows how to configure minimum-reserve level 5 to 20 packets and to assign minimum-reserve level 5 to egress queue 1 on an interface:

```
Switch(config)# mls qos min-reserve 5 20
Switch(config)# interface fastethernet0/1
Switch(config-if)# wrr-queue min-reserve 1 5
```

Configuring the Egress Expedite Queue

You can ensure that certain packets have priority over all others by queuing them in the egress expedite queue. WRR services this queue until it is empty before servicing the other queues.

Beginning in privileged EXEC mode, follow these steps to enable the egress expedite queue:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress 10/100 Ethernet interface, and enter interface configuration mode.
Step 4	**priority-queue out**	Enable the egress expedite queue, which is disabled by default. When you configure this command, the WRR weight is affected because there is one fewer queue participating in WRR. This means that *weight4* in the **wrr-queue bandwidth** command is ignored (not used in the ratio calculation).
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the egress expedite queue, use the **no priority-queue out** interface configuration command.

Allocating Bandwidth among Egress Queues

You need to specify how much of the available bandwidth is allocated to each queue. The ratio of the weights is the ratio of frequency in which the WRR scheduler dequeues packets from each queue.

Beginning in privileged EXEC mode, follow these steps to allocate bandwidth to each queue:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress 10/100 Ethernet interface, and enter interface configuration mode.

	Command	Purpose
Step 4	**wrr-queue bandwidth** *weight1 weight2 weight3 weight4*	Assign WRR weights to the egress queues. By default, all the weights are set to 25 (1/4 of the bandwidth is allocated to each queue). For *weight1 weight2 weight3 weight4*, enter the ratio, which determines the ratio of the frequency in which the WRR scheduler dequeues packets. Separate each value with a space. The range is 1 to 65536. All four queues participate in the WRR unless the expedite queue (queue 4) is enabled, in which case *weight4* is ignored (not used in the ratio calculation). The expedite queue is a priority queue, and it is serviced until empty before the other queues are serviced. A weight of 1 means that the minimum bandwidth is allocated for that queue. To allocate no bandwidth for a queue, use the **wrr-queue cos-map** interface configuration command so that the available bandwidth is shared among the remaining queues.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show mls qos interface queueing**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default bandwidth setting, use the **no wrr-queue bandwidth** interface configuration command.

This example shows how to configure the weight ratio of the WRR scheduler running on the egress queues. In this example, four queues are used (no expedite queue), and the ratio of the bandwidth allocated for each queue is 1/(1+2+3+4), 2/(1+2+3+4), 3/(1+2+3+4), and 4/(1+2+3+4), which is 1/10, 2/10, 3/10, and 4/10 for queues 1, 2, 3, and 4.

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# wrr-queue bandwidth 1 2 3 4
```

Displaying Standard QoS Information

To display standard QoS information, use one or more of the privileged EXEC commands in Table 28-9:

Table 28-9 Commands for Displaying Standard QoS Information

Command	Purpose
show class-map [*class-map-name*]	Display QoS class maps, which define the match criteria to classify traffic.
show mls qos aggregate-policer [*aggregate-policer-name*]	Display the aggregate policer configuration.
show mls qos interface [*interface-id*] [**buffers** | **policers** | **queueing** | **statistics**]	Display QoS information at the interface level, including the configuration of the egress queues and the CoS-to-egress-queue map, which interfaces have configured policers, and ingress and egress statistics (including the number of bytes dropped).[1]

Table 28-9 Commands for Displaying Standard QoS Information (continued)

Command	Purpose
show mls qos maps [**cos-dscp** | **dscp-cos** | **dscp-mutation** | **ip-prec-dscp** | **policed-dscp**]	Display QoS mapping information. Maps are used to generate an internal DSCP value, which represents the priority of the traffic.
show policy-map [*policy-map-name* [**class** *class-map-name*]	Display QoS policy maps, which define classification criteria for incoming traffic.

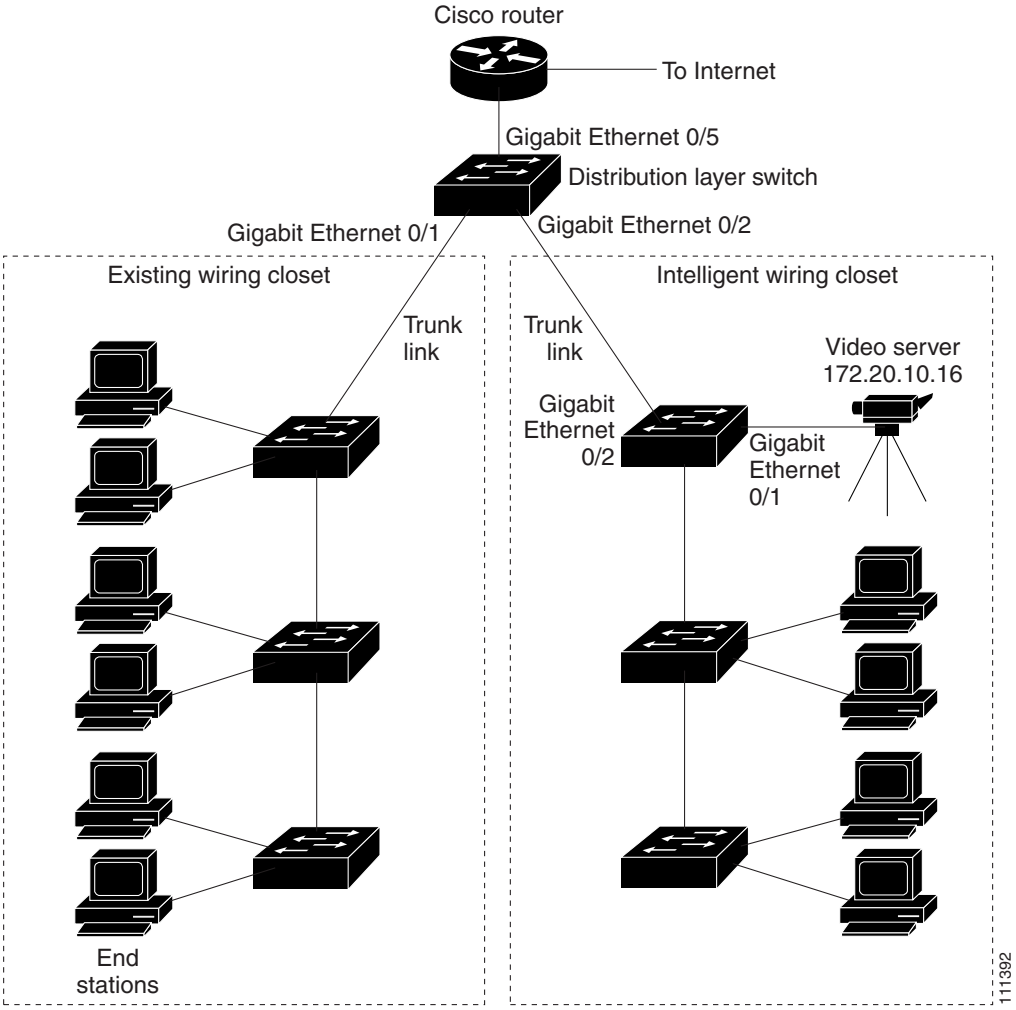
1. You can define up to 16 DSCP values for which byte or packet statistics are gathered by hardware by using the **mls qos monitor** {**bytes** \| **dscp** *dscp1* ... *dscp8* \| **packets**} interface configuration command and the **show mls qos interface statistics** privileged EXEC command.

Standard QoS Configuration Examples

This section shows a QoS migration path to help you quickly implement QoS features based on your existing network and planned changes to your network, as shown in Figure 28-11. It contains this information:

- QoS Configuration for the Existing Wiring Closet, page 28-71
- QoS Configuration for the Intelligent Wiring Closet, page 28-72
- QoS Configuration for the Distribution Layer, page 28-73

Figure 28-11 QoS Configuration Example Network



QoS Configuration for the Existing Wiring Closet

Figure 28-11 shows an existing wiring closet with Catalyst 3500 XL and 2900 XL switches, for example. These switches are running Cisco IOS Release 12.0(5)XP or later, which supports the QoS-based IEEE 802.1p CoS values. QoS classifies frames by assigning priority-indexed CoS values to them and gives preference to higher-priority traffic.

Recall that on the Catalyst 3500 XL and 2900 XL switches, you can classify untagged (native) Ethernet frames at the ingress ports by setting a default CoS priority (**switchport priority default** *default-priority-id* interface configuration command) for each port. For ISL or IEEE 802.1Q frames with tag information, the priority value from the header frame is used. On the Catalyst 3524-PWR XL and 3548 XL switches, you can override this priority with the default value by using the **switchport priority default override** interface configuration command. For Catalyst 3500 XL, 2950, other 2900 XL models that do not have the override feature, the Catalyst 3550-12T switch at the distribution layer can override the IEEE 802.1p CoS value by using the **mls qos cos override** interface configuration command.

For the Catalyst 3500 XL and 2900 XL switches, CoS configures each egress port with a normal-priority transmit queue and a high-priority transmit queue, depending on the frame tag or the port information. Frames in the normal-priority queue are forwarded only after frames in the high-priority queue are forwarded. Frames that have IEEE 802.1p CoS values of 0 to 3 are placed in the normal-priority transmit queue whereas frames with CoS values of 4 to 7 are placed in the expedite (high-priority) queue.

QoS Configuration for the Intelligent Wiring Closet

Figure 28-11 shows an intelligent wiring closet with Catalyst 3550 multilayer switches, for example. One of the switches is connected to a video server, which has an IP address of 172.20.10.16.

The object of this example is to prioritize the video traffic over all other traffic. To do so, a DSCP of 56 is assigned to the video traffic. This traffic is stored in the expedite queue (queue 4), which is serviced until empty before the other queues are serviced. The appropriate CoS value selects queue 4 in the CoS-to-egress-queue map.

Beginning in privileged EXEC mode, follow these steps to configure the switch to prioritize video packets over all other traffic:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list 1 permit 172.20.10.16**	Define an IP standard ACL, and permit traffic from the video server at 172.20.10.16.
Step 3	**class-map videoclass**	Create a class map called *videoclass*, and enter class-map configuration mode.
Step 4	**match access-group 1**	Define the match criterion by matching the traffic specified by access list 1.
Step 5	**exit**	Return to global configuration mode.
Step 6	**policy-map videopolicy**	Create a policy map called *videopolicy*, and enter policy-map configuration mode.
Step 7	**class videoclass**	Specify the class on which to act, and enter policy-map class configuration mode.
Step 8	**set dscp 56**	For traffic matching ACL 1, set the DSCP of incoming packets to 56.
Step 9	**police 5000000 2000000 exceed-action drop**	Define a policer for the classified video traffic to drop traffic that exceeds 5-Mbps average traffic rate with a 2-MB burst size.
Step 10	**exit**	Return to policy-map configuration mode.
Step 11	**exit**	Return to global configuration mode.
Step 12	**interface** *interface-id*	Specify the switch ingress interface that is connected to the video server, and enter interface configuration mode.
Step 13	**service-policy input videopolicy**	Apply the policy to the ingress interface.
Step 14	**exit**	Return to global configuration mode.
Step 15	**interface gigabitethernet0/2**	Enter interface configuration mode, and specify the egress interface (to configure the queues).
Step 16	**priority-queue out**	Enable the expedite queue.

	Command	Purpose
Step 17	**wrr-queue cos-map 4 6 7**	Configure the CoS-to-egress-queue map so that CoS values 6 and 7 select queue 4 (this is the default setting). Because the default DSCP-to-CoS map has DSCP values 56 to 63 mapped to CoS value 7, the matched traffic that is set to DSCP 56 goes to the queue 4, the priority queue.
Step 18	**end**	Return to privileged EXEC mode.
Step 19	**show class-map videoclass** **show policy-map videopolicy** **show mls qos maps** [**cos-dscp** | **dscp-cos**] **show mls qos interface queueing**	Verify your entries.
Step 20	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

QoS Configuration for the Distribution Layer

Figure 28-11 shows a distribution layer switch, for example, a Catalyst 3550 switch. This example focuses on the configuration steps for the distribution layer switch. Because the classification was performed by the switches at the edge of the network, fewer classification steps are needed at the distribution layer switch.

For the connection to the existing wiring closet, Gigabit Ethernet interface 1 on the distribution layer switch is configured to trust the received CoS value. In this situation, the default CoS-to-DSCP map on the multilayer switch is sufficient. For information on the default map settings, see the "Configuring the CoS-to-DSCP Map" section on page 28-52.

For the connection to the intelligent wiring closet, Gigabit Ethernet interface 2 on the distribution layer switch is configured to trust the received DSCP value. The DSCP-to-threshold map also needs to be configured on this ingress interface so that on the egress interface, WRED can provide congestion avoidance control. By default, all DSCP values are mapped to threshold 1.

You need to configure several of the switch maps from their default settings. The object of the configuration is to have only DSCP value 56 sent to the expedite queue (queue 4). The default CoS-to-egress-queue map is sufficient; however, you need to configure the DSCP-to-CoS map so that DSCP values 57 to 63 map to CoS 5.

For the egress interface, Gigabit Ethernet interface 5, WRR weights need to be configured by using the **wrr-queue bandwidth** interface configuration command. WRED needs to be enabled and the threshold percentages configured for each queue. The bandwidth allocated to each queue must be configured to determine the ratio of the frequency at which packets are dequeued.

Beginning in privileged EXEC mode, follow these steps to configure the switch at the distribution layer:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the ingress interface that is connected to the existing wiring closet, and enter interface configuration mode.
Step 4	**mls qos trust cos**	Classify incoming packets on this port by using the packet CoS value.

	Command	Purpose
Step 5	**switchport mode trunk**	Configure this port as a trunk port.
Step 6	**exit**	Return to global configuration mode.
Step 7	**interface** *interface-id*	Specify the ingress interface connected to the intelligent wiring closet, and enter interface configuration mode.
Step 8	**mls qos trust dscp**	Classify incoming packets on this port by trusting the packet DSCP value.
Step 9	**wrr-queue dscp-map** *threshold-id dscp1 ... dscp8*	Map the ingress DSCP values to the WRED thresholds of the egress queues. In the default DSCP-to-threshold map, all DSCP values are mapped to threshold 1. • For *threshold-id*, specify the threshold ID of the queue. The range is 1 to 2. • For *dscp1 ... dscp8*, specify the DSCP values that are mapped to a threshold ID. Enter up to eight DSCP values per command. Separate each value with a space. The DSCP range is 0 to 63.
Step 10	**switchport mode trunk**	Configure this port as a trunk port.
Step 11	**exit**	Return to global configuration mode.
Step 12	**mls qos map dscp-cos** *dscp-list* **to** *cos*	Modify the DSCP-to-CoS map. You can enter up to eight DSCP values separated by spaces in the DSCP-to-CoS map. For example, to map DSCP values 57 to 63 to CoS 5, enter: **mls qos map dscp-cos 57 58 59 60 61 62 63 to 5**
Step 13	**interface** *interface-id*	Specify the egress interface connected to the upstream router, and enter interface configuration mode.
Step 14	**priority-queue out**	Enable the expedite queue.
Step 15	**wrr-queue bandwidth** *weight1 weight2 weight3 weight4*	Configure WRR weights to the egress queues to determine the ratio of the frequency at which packets are dequeued. Separate each value with a space. The weight range is 0 to 65536. In this example, to configure the weights so that queue 4 is serviced more frequently than the other queues, enter: **wrr-queue bandwidth 1 2 3 4** Because the expedite queue is enabled, only the first three weights are used in the ratio calculation.
Step 16	**wrr-queue random-detect max-threshold** *queue-id threshold-percentage1 threshold-percentage2*	Enable WRED and assign two WRED threshold values to an egress queue of a Gigabit-capable Ethernet port. • For *queue-id*, the range is 1 to 4. • For *threshold-percentage1 threshold-percentage2*, the range is 1 to 100 percent. In this example, to configure the thresholds, enter: **wrr-queue random-detect max-threshold 1 20 100** **wrr-queue random-detect max-threshold 2 40 100** **wrr-queue random-detect max-threshold 3 60 100** **wrr-queue random-detect max-threshold 4 80 100**

	Command	Purpose
Step 17	**end**	Return to privileged EXEC mode.
Step 18	**show mls qos interface** and **show interfaces**	Verify your entries.
Step 19	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

CHAPTER 29

Configuring Network Security with ACLs

This chapter describes how to configure network security on your Catalyst 3550 switch by using access control lists (ACLs), which are also referred to in commands and tables as access lists.

For complete syntax and usage information for the commands used in this chapter, see the command reference for this release, see the "Configuring IP Services" section in the "IP Addressing and Services" chapter of the *Cisco IOS IP Configuration Guide, Release 12.2*, and to these software configuration guides and command references:

- *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2*
- *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*
- *Cisco IOS IP Command Reference, Volume 3 of 3: Multicast, Release 12.2*

This chapter consists of these sections:

- Understanding ACLs, page 29-1
- Configuring IP ACLs, page 29-6
- Configuring Named MAC Extended ACLs, page 29-27
- Configuring VLAN Maps, page 29-30
- Using VLAN Maps with Router ACLs, page 29-36
- Displaying ACL Information, page 29-40

Note To allocate system resources to maximize the number of security access control entries (ACEs) allowed on the switch, you can use the **sdm prefer access** global configuration command to set the Switch Database Management (sdm) feature to the access template. For more information on the SDM templates, see the "Optimizing System Resources for User-Selected Features" section on page 6-26. For information about determining resource usage for your configuration, see the "Displaying ACL Resource Usage and Configuration Problems" section on page 29-42.

Understanding ACLs

Packet filtering can help limit network traffic and restrict network use by certain users or devices. ACLs can filter traffic as it passes through a router and permit or deny packets at specified interfaces. An ACL is a sequential collection of permit and deny conditions that apply to packets. When a packet is received on an interface, the switch compares the fields in the packet against any applied ACLs to verify that the packet has the required permissions to be forwarded, based on the criteria specified in the access lists. It tests packets against the conditions in an access list one by one. The first match determines whether

the switch accepts or rejects the packets. Because the switch stops testing conditions after the first match, the order of conditions in the list is critical. If no conditions match, the switch rejects the packets. If there are no restrictions, the switch forwards the packet; otherwise, the switch drops the packet.

Switches traditionally operate at Layer 2 only, switching traffic within a VLAN, whereas routers route traffic between VLANs. The Catalyst 3550 switch can accelerate packet routing between VLANs by using Layer 3 switching. The switch bridges the packet, the packet is then routed internally without going to an external router, and then the packet is bridged again to send it to its destination. During this process, the switch can access-control all packets it switches, including packets bridged within a VLAN.

You configure access lists on a router or switch to provide basic security for your network. If you do not configure ACLs, all packets passing through the switch could be allowed onto all parts of the network. You can use ACLs to control which hosts can access different parts of a network or to decide which types of traffic are forwarded or blocked at router interfaces. For example, you can allow e-mail traffic to be forwarded but not Telnet traffic. ACLs can be configured to block inbound traffic, outbound traffic, or both. However, on Layer 2 interfaces, you can only apply ACLs in the inbound direction.

An ACL contains an ordered list of access control entries (ACEs). Each ACE specifies *permit* or *deny* and a set of conditions the packet must satisfy in order to match the ACE. The meaning of *permit* or *deny* depends on the context in which the ACL is used.

The switch supports two types of ACLs:

- IP ACLs filter IP traffic, including TCP, User Datagram Protocol (UDP), Internet Group Management Protocol (IGMP), and Internet Control Message Protocol (ICMP).
- Ethernet or MAC ACLs filter non-IP traffic.

Supported ACLs

The switch supports three applications of ACLs to filter traffic:

- Router ACLs access-control routed traffic between VLANs and are applied to Layer 3 interfaces. You can apply one router ACL in each direction on an interface.
- Port ACLs access-control traffic entering a Layer 2 interface. The switch does not support port ACLs in the outbound direction. You can apply only one IP access list and one MAC access list to a Layer 2 interface.
- VLAN ACLs or VLAN maps access-control all packets (bridged and routed). You can use VLAN maps to filter traffic between devices in the same VLAN. VLAN maps are configured to provide access-control based on Layer 3 addresses for IP. Unsupported protocols are access-controlled through MAC addresses by using Ethernet ACEs. After a VLAN map is applied to a VLAN, all packets (routed or bridged) entering the VLAN are checked against the VLAN map. Packets can either enter the VLAN through a switch port or through a routed port after being routed.

You can use both router ACLs and VLAN maps on the same switch. However, you cannot use port ACLs on a switch that contains input router ACLs or VLAN maps.

- When a switch has a Layer 2 interface with an applied IP access list or MAC access list, you can create IP access lists and VLAN maps, but you cannot apply an IP access list to an input Layer 3 interface on that switch, and you cannot apply a VLAN map to any of the switch VLANs. An error message is generated if you attempt to do so. You can still apply an IP access list to an output Layer 3 interface on a switch with port ACLs.
- When a switch has an input Layer 3 ACL or a VLAN map applied to it, you cannot apply an IP access list or MAC access list to a Layer 2 interface on that switch. An error message is generated if you attempt to do so. You can apply a port ACL if the switch has an ACL applied to an output Layer 3 interface.

If IEEE 802.1Q tunneling is configured on an interface, any IEEE 802.1Q encapsulated IP packets received on the tunnel port can be filtered by MAC ACLs, but not by IP ACLs. This is because the switch does not recognize the protocol inside the IEEE 802.1Q header. This restriction applies to router ACLs, port ACLs, and VLAN maps. For more information about IEEE 802.1Q tunneling, see Chapter 14, "Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling."

This switch also supports Quality of Service (QoS) classification ACLs. For more information, see the "Classification Based on QoS ACLs" section on page 28-7.

Router ACLs

You can apply router ACLs on switch virtual interfaces (SVIs), which are Layer 3 interfaces to VLANs; on physical Layer 3 interfaces; and on Layer 3 EtherChannel interfaces. Router ACLs are applied on interfaces for specific directions (inbound or outbound). You can apply one IP access list in each direction.

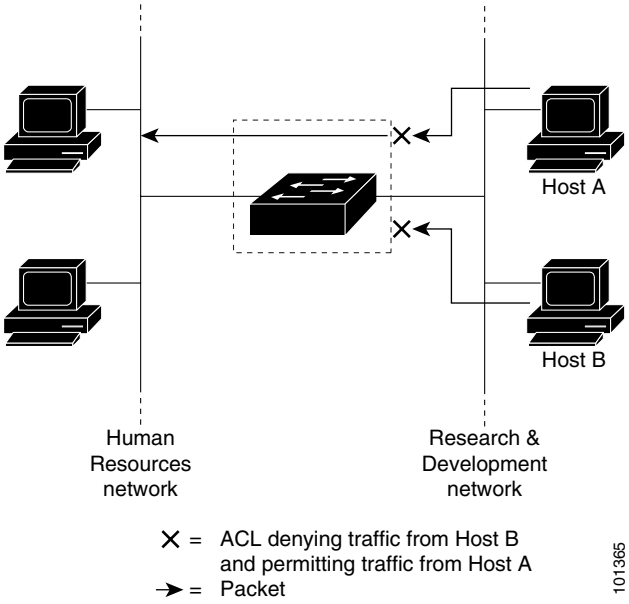
One ACL can be used with multiple features for a given interface, and one feature can use multiple ACLs. When a single router ACL is used by multiple features, it is examined multiple times.

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses and optional protocol type information for matching operations.

The switch examines ACLs associated with features configured on a given interface and a direction. As packets enter the switch on an interface, ACLs associated with all inbound features configured on that interface are examined. After packets are routed and before they are forwarded to the next hop, all ACLs associated with outbound features configured on the egress interface are examined.

ACLs permit or deny packet forwarding based on how the packet matches the entries in the ACL. For example, you can use access lists to allow one host to access a part of a network, but prevent another host from accessing the same part. In Figure 29-1, ACLs applied at the router input allow Host A to access the Human Resources network, but prevent Host B from accessing the same network.

Figure 29-1 Using ACLs to Control Traffic to a Network



Port ACLs

You can also apply ACLs to Layer 2 interfaces on a switch. Port ACLs are supported on physical interfaces only and not on EtherChannel interfaces. Port ACLs are applied on interfaces for inbound traffic only. These access lists are supported on Layer 2 interfaces:

- Standard IP access lists using source addresses
- Extended IP access lists using source and destination addresses and optional protocol type information
- MAC extended access lists using source and destination MAC addresses and optional protocol type information

As with router ACLs, the switch examines ACLs associated with features configured on a given interface and permits or denies packet forwarding based on how the packet matches the entries in the ACL. However, ACLs can only be applied to Layer 2 interfaces in the inbound direction. In the example in Figure 29-1, if all workstations were in the same VLAN, ACLs applied at the Layer 2 input would allow Host A to access the Human Resources network, but prevent Host B from accessing the same network.

When you apply a port ACL to a trunk port, the ACL filters traffic on all VLANs present on the trunk port. When you apply a port ACL to a port with voice VLAN, the ACL filters traffic on both data and voice VLANs.

With port ACLs, you can filter IP traffic by using IP access lists and non-IP traffic by using MAC addresses. You can filter both IP and non-IP traffic on the same Layer 2 interface by applying both an IP access list and a MAC access list to the interface.

Note You cannot apply more than one IP access list and one MAC access list to a Layer 2 interface. If an IP access list or MAC access list is already configured on a Layer 2 interface and you apply a new IP access list or MAC access list to the interface, the new ACL replaces the previously configured one.

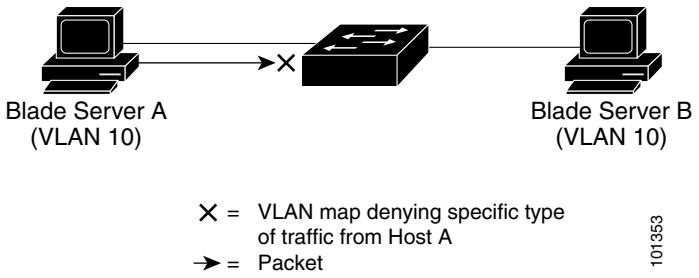
VLAN Maps

VLAN maps can access-control *all* traffic. You can apply VLAN maps on the switch to all packets that are routed into or out of a VLAN or are bridged within a VLAN. VLAN maps are used strictly for security packet filtering. Unlike router ACLs, VLAN maps are not defined by direction (input or output).

You can configure VLAN maps to match Layer 3 addresses for IP traffic. All non-IP protocols are access-controlled through MAC addresses and Ethertype using MAC VLAN maps. (IP traffic *is not* access controlled by MAC VLAN maps.) You can enforce VLAN maps only on packets going through the switch; you cannot enforce VLAN maps on traffic between hosts on a hub or on another switch connected to this switch.

With VLAN maps, forwarding of packets is permitted or denied, based on the action specified in the map. Figure 29-2 illustrates how a VLAN map is applied to deny a specific type of traffic from Host A in VLAN 10 from being forwarded.

Figure 29-2 Using VLAN Maps to Control Traffic



Handling Fragmented and Unfragmented Traffic

IP packets can be fragmented as they cross the network. When this happens, only the fragment containing the beginning of the packet contains the Layer 4 information, such as TCP or UDP port numbers, ICMP type and code, and so on. All other fragments are missing this information.

Some ACEs do not check Layer 4 information and therefore can be applied to all packet fragments. ACEs that do test Layer 4 information cannot be applied in the standard manner to most of the fragments in a fragmented IP packet. When the fragment contains no Layer 4 information and the ACE tests some Layer 4 information, the matching rules are modified:

- Permit ACEs that check the Layer 3 information in the fragment (including protocol type, such as TCP, UDP, and so on) are considered to match the fragment regardless of what the missing Layer 4 information might have been.
- Deny ACEs that check Layer 4 information never match a fragment unless the fragment contains Layer 4 information.

Consider access list 102, configured with these commands, applied to three fragmented packets:

```
Switch (config)# access-list 102 permit tcp any host 10.1.1.1 eq smtp
Switch (config)# access-list 102 deny tcp any host 10.1.1.2 eq telnet
Switch (config)# access-list 102 permit tcp any host 10.1.1.2
Switch (config)# access-list 102 deny tcp any any
```

Note In the first two ACEs, the *eq* keyword after the destination address means to test for the TCP destination-port well-known numbers equaling Simple Mail Transfer Protocol (SMTP) and Telnet, respectively.

- Packet A is a TCP packet from host 10.2.2.2., port 65000, going to host 10.1.1.1 on the SMTP port. If this packet is fragmented, the first fragment matches the first ACE (a permit) as if it were a complete packet because all Layer 4 information is present. The remaining fragments also match the first ACE, even though they do not contain the SMTP port information, because the first ACE only checks Layer 3 information when applied to fragments. The information in this example is that the packet is TCP and that the destination is 10.1.1.1.
- Packet B is from host 10.2.2.2, port 65001, going to host 10.1.1.2 on the Telnet port. If this packet is fragmented, the first fragment matches the second ACE (a deny) because all Layer 3 and Layer 4 information is present. The remaining fragments in the packet do not match the second ACE because they are missing Layer 4 information. Instead, they match the third ACE (a permit).

Because the first fragment was denied, host 10.1.1.2 cannot reassemble a complete packet, so packet B is effectively denied. However, the later fragments that are permitted will consume bandwidth on the network and resources of host 10.1.1.2 as it tries to reassemble the packet.

- Fragmented packet C is from host 10.2.2.2, port 65001, going to host 10.1.1.3, port ftp. If this packet is fragmented, the first fragment matches the fourth ACE (a deny). All other fragments also match the fourth ACE because that ACE does not check any Layer 4 information and because Layer 3 information in all fragments shows that they are being sent to host 10.1.1.3, and the earlier permit ACEs were checking different hosts.

Configuring IP ACLs

Configuring IP ACLs on Layer 2 or Layer 3 switch or VLAN interfaces is the same as configuring ACLs on other Cisco routers. The process is briefly described here. For more detailed information on configuring router ACLs, see the "Configuring IP Services" section in the "IP Addressing and Services" chapter of the *Cisco IOS IP Configuration Guide, Release 12.2*. For detailed information about the commands, see these documents:

- *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2*
- *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*
- *Cisco IOS IP Command Reference, Volume 3 of 3: Multicast, Release 12.2*

For a list of IOS features not supported on the Catalyst 3550 switch, see the "Unsupported Features" section on page 29-8.

Caution By default, the router sends Internet Control Message Protocol (ICMP) unreachable messages when a packet is denied by an access group; these access-group denied packets are not dropped in hardware but are bridged to the switch CPU so that it can generate the ICMP-unreachable message. To drop access-group denied packets in hardware, you must disable ICMP unreachables by using the **no ip unreachables** interface configuration command. The **ip unreachables** command is enabled by default.

This section includes the following information:

- Hardware and Software Handling of Router ACLs, page 29-6
- Configuration Guidelines for Input Router ACLs, page 29-7
- Unsupported Features, page 29-8
- Creating Standard and Extended IP ACLs, page 29-8
- Applying an IP ACL to an Interface or Terminal Line, page 29-21
- IP ACL Configuration Examples, page 29-23

Hardware and Software Handling of Router ACLs

ACL processing is primarily accomplished in hardware, but requires forwarding of some traffic flows to the CPU for software processing. The forwarding rate for software-forwarded traffic is substantially less than for hardware-forwarded traffic. When traffic flows are both logged and forwarded, forwarding is done by hardware, but logging must be done by software. Because of the difference in packet handling capacity between hardware and software, if the sum of all flows being logged (both permitted flows and denied flows) is of great enough bandwidth, not all of the packets that are forwarded can be logged.

These factors can cause packets to be sent to the CPU:

- Using the **log** keyword
- Enabling ICMP unreachables
- Hardware reaching its capacity to store ACL configurations

If ACLs cause large numbers of packets to be sent to the CPU, the switch performance can be negatively affected.

Note After the ACL configuration is stable for a specified interval, the system loads the configuration into hardware. Forwarding is blocked on any affected interfaces while the hardware is being updated. To change this behavior, you can use the **mls aclmerge delay** and the **access-list hardware program nonblocking** global configuration commands. For descriptions of these commands, see the command reference for this release.

When you enter the **show ip access-lists** privileged EXEC command, the match count displayed does not account for packets that are access controlled in hardware. Use the **show access-lists hardware counters** privileged EXEC command to obtain some basic hardware ACL statistics for switched and routed packets.

IP ACLs are handled as follows:

- The hardware controls permit and deny actions of standard and extended ACLs (input and output) for security access control.
- If **log** has not been specified, the flows that match a *deny* statement in a security ACL are dropped by the hardware if *ip unreachables* is disabled. The flows matching a *permit* statement are switched in hardware. Logging is not supported for port ACLs.
- Adding the **log** keyword to an ACE in a router ACL causes a copy of the packet to be sent to the CPU only for logging. If the ACE is a *permit* statement, the packet is still switched and routed in hardware.

Note Logging is not supported on Layer 2 interfaces (port ACLs).

Configuration Guidelines for Input Router ACLs

With input router ACLs, there can be a large expansion in the number of TCAM entries when the ACL is applied. If the number of TCAM entries exceeds the allocated resources, ACL filtering is done in software instead of hardware, which can have a negative impact on performance.

There are several ways to prevent excessive TCAM usage:

- Use the **sdm prefer access** global configuration command to change the switch database management (SDM) template to allow more access lists.
- Use output router ACLs instead of input router ACLs.
- Minimize the TCAM usage of input router ACLs by configuring explicit permits or denies.

  When an input router ACL is applied, it is automatically merged with an implicit ACL that matches against routing protocol packets and sends them to the protocol queue. This merge results in additional TCAM entries. To minimize the number of entries, you can configure router ACLs to explicitly permit or deny routing protocols, such as RIP, EIGRP, OSPF, BGP, and PIM, by configuring permit or deny ACEs at the beginning of the ACL.

This is an example of how to configure an input router ACL to minimize TCAM usage:

```
Switch(config)# access-list 100 [permit|deny] tcp any any eq bgp
Switch(config)# access-list 100 [permit|deny] eigrp any any
Switch(config)# access-list 100 [permit|deny] pim any any
Switch(config)# access-list 100 [permit|deny] ospf any any
Switch(config)# access-list 100 [permit|deny] udp any any eq rip
Switch(config)# access-list 100 ...... ACL 100's ACE(s)
Switch(config)# exit
```

Unsupported Features

The Catalyst 3550 switch does not support these Cisco IOS router ACL-related features:

- Non-IP protocol ACLs (see Table 29-1 on page 29-9).
- Bridge-group ACLs.
- IP accounting.
- Inbound and outbound rate limiting (except with QoS ACLs).
- IP packets with a header length of less than five are not access controlled (results in an ICMP parameter error).
- Reflexive ACLs.
- Dynamic ACLs (except for certain specialized dynamic ACLs used by the switch clustering feature).
- For Layer 2 port ACLs, the switch does not support logging or outbound ACLs.

Creating Standard and Extended IP ACLs

This section summarizes how to create router IP ACLs. An ACL is a sequential collection of permit and deny conditions. The switch tests packets against the conditions in an access list one by one. The first match determines whether the switch accepts or rejects the packet. Because the switch stops testing conditions after the first match, the order of the conditions is critical. If no conditions match, the switch denies the packet.

These are the steps to use IP ACLs:

Step 1 Create an ACL by specifying an access list number or name and access conditions.

Step 2 Apply the ACL to interfaces or terminal lines. You can also apply standard and extended IP ACLs to VLAN maps.

The software supports these styles of ACLs or access lists for IP:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses for matching operations and optional protocol-type information for finer granularity of control.

These sections describe access lists and the steps for using them:

- Access List Numbers, page 29-9
- Creating a Numbered Standard ACL, page 29-10

- Creating a Numbered Extended ACL, page 29-11
- Creating Named Standard and Extended IP ACLs, page 29-16
- Using Time Ranges with ACLs, page 29-18
- Including Comments in ACLs, page 29-20

Access List Numbers

The number you use to denote your ACL shows the type of access list that you are creating. Table 29-1 lists the access-list number and corresponding access list type and shows whether or not they are supported in the switch. The switch supports IP standard and IP extended access lists, numbers 1 to 199 and 1300 to 2699.

Table 29-1 Access List Numbers

Access List Number	Type	Supported
1–99	IP standard access list	Yes
100–199	IP extended access list	Yes
200–299	Protocol type-code access list	No
300–399	DECnet access list	No
400–499	XNS standard access list	No
500–599	XNS extended access list	No
600–699	AppleTalk access list	No
700–799	48-bit MAC address access list	No
800–899	IPX standard access list	No
900–999	IPX extended access list	No
1000–1099	IPX SAP access list	No
1100–1199	Extended 48-bit MAC address access list	No
1200–1299	IPX summary address access list	No
1300–1999	IP standard access list (expanded range)	Yes
2000–2699	IP extended access list (expanded range)	Yes

Note In addition to numbered standard and extended ACLs, you can also create standard and extended named IP ACLs using the supported numbers. That is, the name of a standard IP ACL can be 1 to 99; the name of an extended IP ACL can be 100 to 199. The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

Creating a Numbered Standard ACL

Beginning in privileged EXEC mode, follow these steps to create a numbered standard ACL:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] [**log**]	Define a standard IP access list by using a source address and wildcard. The *access-list-number* is a decimal number from 1 to 99 or 1300 to 1999. Enter **deny** or **permit** to specify whether to deny or permit access if conditions are matched. The *source* is the source address of the network or host from which the packet is being sent specified as: • The 32-bit quantity in dotted-decimal format. • The keyword **any** as an abbreviation for *source* and *source-wildcard* of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard. • The keyword **host** as an abbreviation for source and source-wildcard of *source* 0.0.0.0. (Optional) The *source-wildcard* applies wildcard bits to the source. (Optional) Enter **log** to create an informational logging message about the packet that matches the entry to be sent to the console. **Note** The **log** keyword is ignored on ACLs applied to Layer 2 interfaces.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show access-lists** [*number* | *name*]	Show the access list configuration.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no access-list** *access-list-number* global configuration command to delete the entire ACL. You cannot delete individual ACEs from numbered access lists.

Note When creating an ACL, remember that, by default, the end of the ACL contains an implicit deny statement for all packets that it did not find a match for before reaching the end. With standard access lists, if you omit the mask from an associated IP host address ACL specification, 0.0.0.0 is assumed to be the mask.

This example shows how to create a standard ACL to deny access to IP host 171.69.198.102, permit access to any others, and display the results.

```
Switch (config)# access-list 2 deny host 171.69.198.102
Switch (config)# access-list 2 permit any
Switch(config)# end
Switch# show access-lists
Standard IP access list 2
    10 deny   171.69.198.102
    20 permit any
```

The switch always rewrites the order of standard access lists so that entries with **host** matches and entries with matches having a *don't care* mask of 0.0.0.0 are moved to the top of the list, above any entries with non-zero *don't care* masks. Therefore, in **show** command output and in the configuration file, the ACEs do not necessarily appear in the order in which they were entered.

The switch software can provide logging messages about packets permitted or denied by a standard IP access list. That is, any packet that matches the ACL causes an informational logging message about the packet to be sent to the console. The level of messages logged to the console is controlled by the logging console commands controlling the syslog messages.

Note Because routing is done in hardware and logging is done in software, if a large number of packets match a *permit* or *deny* ACE containing a **log** keyword, the software might not be able to match the hardware processing rate, and not all packets will be logged.

The first packet that triggers the ACL causes a logging message right away, and subsequent packets are collected over 5-minute intervals before they are displayed or logged. The logging message includes the access list number, whether the packet was permitted or denied, the source IP address of the packet, and the number of packets from that source permitted or denied in the prior 5-minute interval.

Note An output ACL cannot log multicast packets. Logging is not supported for ACLs applied to Layer 2 interfaces.

After creating an ACL, you must apply it to a line or interface, as described in the "Applying an IP ACL to an Interface or Terminal Line" section on page 29-21.

Creating a Numbered Extended ACL

Although standard ACLs use only source addresses for matching, you can use extended ACL source and destination addresses for matching operations and optional protocol type information for finer granularity of control. When you are creating ACEs in numbered extended access lists, remember that after you create the ACL, any additions are placed at the end of the list. You cannot reorder the list or selectively add or remove ACEs from a numbered list.

Some protocols also have specific parameters and keywords that apply to that protocol.

Extended ACLs support these IP protocols (protocol keywords are in bold):

Authentication Header Protocol (**ahp**), Enhanced Interior Gateway Routing Protocol (**eigrp**), Encapsulation Security Payload (**esp**), generic routing encapsulation (**gre**), Internet Control Message Protocol (**icmp**), Internet Group Management Protocol (**igmp**), Interior Gateway Routing Protocol (**igrp**), any Interior Protocol (**ip**), IP in IP tunneling (**ipinip**), KA9Q NOS-compatible IP over IP tunneling (**nos**), Open Shortest Path First routing (**ospf**), Payload Compression Protocol (**pcp**), Protocol Independent Multicast (**pim**), Transmission Control Protocol (**tcp**), or User Datagram Protocol (**udp**).

Note ICMP echo-reply cannot be filtered. All other ICMP codes or types can be filtered.

For more details on the specific keywords relative to each protocol, see these software configuration guides and command references:

- *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2*
- *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*
- *Cisco IOS IP Command Reference, Volume 3 of 3: Multicast, Release 12.2*

Note The switch does not support dynamic or reflexive access lists. It also does not support filtering based on the type of service (ToS) minimize-monetary-cost bit.

Supported parameters can be grouped into these categories: TCP, UDP, ICMP, IGMP, or other IP.

Beginning in privileged EXEC mode, follow these steps to create an extended ACL:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2a	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*] **Note** If you enter a **dscp** value, you cannot enter **tos** or **precedence**. You can enter both a **tos** and a **precedence** value with no **dscp**.	Define an extended IP access list and the access conditions. The *access-list-number* is a decimal number from 100 to 199 or 2000 to 2699. Enter **deny** or **permit** to specify whether to deny or permit the packet if conditions are matched. For *protocol*, enter the name or number of an IP protocol: **ahp**, **eigrp**, **esp**, **gre**, **icmp**, **igmp**, **igrp**, **ip**, **ipinip**, **nos**, **ospf**, **pcp**, **pim**, **tcp**, or **udp**, or an integer in the range 0 to 255 representing an IP protocol number. To match any Internet protocol (including ICMP, TCP, and UDP) use the keyword **ip**. **Note** This step includes options for most IP protocols. For additional specific parameters for TCP, UDP, ICMP, and IGMP, see steps 2b through 2e. Specific parameters for IGMP apply to IGMP Version 1 and Version 2. Extended ACLs do not support IGMP Version 3. The *source* is the number of the network or host from which the packet is sent. The *source-wildcard* applies wildcard bits to the source. The *destination* is the network or host number to which the packet is sent. The *destination-wildcard* applies wildcard bits to the destination. Source, source-wildcard, destination, and destination-wildcard can be specified as: • The 32-bit quantity in dotted-decimal format. • The keyword **any** for 0.0.0.0 255.255.255.255 (any host). • The keyword **host** for a single host 0.0.0.0. The other keywords are optional and have these meanings: • **precedence**—Enter to match packets with a precedence level specified as a number from 0 to 7 or by name: **routine** (**0**), ***priority*** (**1**), **immediate** (**2**), **flash** (**3**), **flash-override** (**4**), **critical** (**5**), **internet** (6), **network** (**7**). • **fragments**—Enter to check non-initial fragments. • **tos**—Enter to match by type of service level, specified by a number from 0 to 15 or a name: **normal** (**0**), **max-reliability** (**2**), **max-throughput** (**4**), **min-delay** (**8**). • **log**—Enter to create an informational logging message to be sent to the console about the packet that matches the entry or **log-input** to include the input interface in the log entry. Logging is not supported for ACLs applied to Layer 2 interfaces (port ACLs). • **time-range**—For an explanation of this keyword, see the "Using Time Ranges with ACLs" section on page 29-18. • **dscp**—Enter to match packets with the DSCP value specified by a number from 0 to 63, or use the question mark (?) to see a list of available values.

	Command	Purpose
or	**access-list** *access-list-number* {**deny** | **permit**} *protocol* **any any** [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	In access-list configuration mode, define an extended IP access list using an abbreviation for a source and source wildcard of 0.0.0.0 255.255.255.255 and an abbreviation for a destination and destination wildcard of 0.0.0.0 255.255.255.255. You can use the **any** keyword in place of source and destination address and wildcard.
or	**access-list** *access-list-number* {**deny** | **permit**} *protocol* **host** *source* **host** *destination* [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	Define an extended IP access list using an abbreviation for a source and source wildcard of *source* 0.0.0.0 and an abbreviation for a destination and destination wildcard of *destination* 0.0.0.0. You can use the **host** keyword in place of source and destination wildcard or mask.
Step 2b	**access-list** *access-list-number* {**deny** | **permit**} **tcp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**established**] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*] [*flag*]	(Optional) Define an extended TCP access list and the access conditions. Enter **tcp** for Transmission Control Protocol. The parameters are the same as those described in Step 2a with these exceptions: (Optional) Enter an *operator* and *port* to compare source (if positioned after *source source-wildcard*) or destination (if positioned after *destination destination-wildcard*) port. Possible operators include **eq** (equal), **gt** (greater than), **lt** (less than), **neq** (not equal), and **range** (inclusive range). Operators require a port number (**range** requires two port numbers separated by a space). Enter the *port* number as a decimal number (from 0 to 65535) or the name of a TCP port. To see TCP port names, use the ? or see "Configuring IP Services" section in the "IP Addressing and Services" chapter of the *Cisco IOS IP Configuration Guide, Release 12.2*. Use only TCP port numbers or names when filtering TCP. The additional optional keywords have these meanings: • **established**—Enter to match an established connection. This has the same function as matching on the **ack** or **rst** flag. • *flag*—Enter one of these flags to match by the specified TCP header bits: **ack** (acknowledge), **fin** (finish), **psh** (push), **rst** (reset), **syn** (synchronize), or **urg** (urgent).
Step 2c	**access-list** *access-list-number* {**deny** | **permit**} **udp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	(Optional) Define an extended UDP access list and the access conditions. Enter **udp** for the User Datagram Protocol. The UDP parameters are the same as those described for TCP except that [*operator* [*port*]] port number or name must be a UDP port number or name, and the **flag** and **established** parameters are not valid for UDP.

	Command	Purpose
Step 2d	**access-list** *access-list-number* {**deny** | **permit**} **icmp** *source source-wildcard destination destination-wildcard* [*icmp-type* | [[*icmp-type icmp-code*] | *[icmp-message]*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	(Optional) Define an extended ICMP access list and the access conditions. Enter **icmp** for Internet Control Message Protocol. The ICMP parameters are the same as those described for most IP protocols in Step 2a, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings: • *icmp-type*—Enter to filter by ICMP message type, a number from 0 to 255. • *icmp-code*—Enter to filter ICMP packets that are filtered by ICMP message type by the ICMP message code, a number from 0 to 255. • *icmp-message*—Enter to filter ICMP packets by ICMP message type name or ICMP message type and code name. To see a list of ICMP message type names and ICMP message type and code names, use the ? or see the "Configuring IP Services" section of the *Cisco IOS IP Configuration Guide, Release 12.2*
Step 2e	**access-list** *access-list-number* {**deny** | **permit**} **igmp** *source source-wildcard destination destination-wildcard* [*igmp-type*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	(Optional) Define an extended IGMP access list and the access conditions. Enter **igmp** for Internet Group Management Protocol. The IGMP parameters are the same as those described for most IP protocols in Step 2a, with the addition of this optional parameter. **Note** When configuring extended ACLs for the IGMP protocol using the command **access-list** *access-list-number* {**permit**} **igmp**, IGMP Version 1 and Version 2 are supported. IGMP Version 3 is not supported. *igmp-type*—To match IGMP message type, use one of these values when filtering IGMP Version 1 or IGMP Version 2 protocol packets: • 1—IGMP query • 2—IGMP Version 1 report • 3—IGMP DVMRP packets • 4—IGMP PIM Version1 packets • 6—IGMP Version 2 report • 7—IGMP leave • e—IGMP MTRACE response packets • f—IGMP MTRACE request packets
Step 3	**show access-lists** [*number* | *name*]	Verify the access list configuration.
Step 4	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no access-list** *access-list-number* global configuration command to delete the entire access list. You cannot delete individual ACEs from numbered access lists.

This example shows how to create and display an extended access list to deny Telnet access from any host in network 171.69.198.0 to any host in network 172.20.52.0 and permit any others. (The **eq** keyword after the destination address means to test for the TCP destination port number equaling Telnet.)

```
Switch(config)# access-list 102 deny tcp 171.69.198.0 0.0.0.255 172.20.52.0 0.0.0.255 eq
telnet
Switch(config)# access-list 102 permit tcp any any
Switch(config)# end
Switch# show access-lists
Extended IP access list 102
    10 deny tcp 171.69.198.0 0.0.0.255 172.20.52.0 0.0.0.255 eq telnet
    20 permit tcp any any
```

After an ACL is created, any additions (possibly entered from the terminal) are placed at the end of the list. You cannot selectively add or remove access list entries from a numbered access list.

Note When you are creating an ACL, remember that, by default, the end of the access list contains an implicit deny statement for all packets if it did not find a match before reaching the end.

After creating an ACL, you must apply it to a line or interface, as described in the "Applying an IP ACL to an Interface or Terminal Line" section on page 29-21.

Resequencing ACEs in an ACL

In Cisco IOS Release 12.2(18)SE and later, sequence numbers for the entries in an access list are automatically generated when you create a new ACL.You can use the **ip access-list resequence** global configuration command to edit the sequence numbers in an ACL and change the order in which ACEs are applied. For example, if you add a new ACE to an ACL, it is placed at the bottom of the list. By changing the sequence number, you can move the ACE to a different position in the ACL.

For more information about the **ip access-list resequence** command, see this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122s/122snwft/release/122s14/fsaclseq.htm

Creating Named Standard and Extended IP ACLs

You can identify IP ACLs with an alphanumeric string (a name) rather than a number. You can use named ACLs to configure more IP access lists in a switch than if you were to use numbered access lists. If you identify your access list with a name rather than a number, the mode and command syntax are slightly different. However, not all commands that use IP access lists accept a named access list.

Note The name you give to a standard or extended ACL can also be a number in the supported range of access list numbers. That is, the name of a standard IP ACL can be 1 to 99; the name of an extended IP ACL can be 100 to 199. The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

Consider these guidelines and limitations before configuring named ACLs:

- Not all commands that accept a numbered ACL accept a named ACL. ACLs for packet filters and route filters on interfaces can use a name. VLAN maps also accept a name.
- A standard ACL and an extended ACL cannot have the same name.

- Numbered ACLs are also available, as described in the "Creating Standard and Extended IP ACLs" section on page 29-8.
- You can apply standard and extended ACLs (named or numbered) to VLAN maps.

Beginning in privileged EXEC mode, follow these steps to create a standard ACL using names:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip access-list standard** *name*	Define a standard IP access list using a name, and enter access-list configuration mode. **Note** The name can be a number from 1 to 99.
Step 3	**deny** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**] or **permit** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]	In access-list configuration mode, specify one or more conditions denied or permitted to determine if the packet is forwarded or dropped. • **host** *source*—A source and source wildcard of *source* 0.0.0.0. • **any**—A source and source wildcard of 0.0.0.0 255.255.255.255. **Note** The **log** keyword is not supported for ACLs applied to Layer 2 interfaces (port ACLs).
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show access-lists** [*number* | *name*]	Show the access list configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a named standard ACL, use the **no ip access-list standard** *name* global configuration command.

Beginning in privileged EXEC mode, follow these steps to create an extended ACL using names:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip access-list extended** *name*	Define an extended IP access list using a name and enter access-list configuration mode. **Note** The name can be a number from 100 to 199.

	Command	Purpose
Step 3	{**deny** | **permit**} *protocol* {*source* [*source-wildcard*] | **host** *source* | **any**} {*destination* [*destination-wildcard*] | **host** *destination* | **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log**] [**time-range** *time-range-name*]	In access-list configuration mode, specify the conditions allowed or denied. Use the **log** keyword to get access list logging messages, including violations. See the "Creating a Numbered Extended ACL" section on page 29-11 for definitions of protocols and other keywords. • **host** *source*—A source and source wildcard of *source* 0.0.0.0. • ***host*** *destination*—A destination and destination wildcard of *destination* 0.0.0.0. • **any**—A source and source wildcard or destination and destination wildcard of 0.0.0.0 255.255.255.255. **Note** The **log** keyword is not supported for ACLs applied to Layer 2 interfaces (port ACLs).
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show access-lists** [*number* | *name*]	Show the access list configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a named extended ACL, use the **no ip access-list extended** *name* global configuration command.

When you are creating standard extended ACLs, remember that, by default, the end of the ACL contains an implicit deny statement for everything if it did not find a match before reaching the end. For standard ACLs, if you omit the mask from an associated IP host address access list specification, 0.0.0.0 is assumed to be the mask.

After you create an ACL, any additions are placed at the end of the list. You cannot selectively add ACL entries to a specific ACL. However, you can use **no permit** and **no deny** access-list configuration mode commands to remove entries from a named ACL. This example shows how you can delete individual ACEs from the named access list *border-list*:

```
Switch(config)# ip access-list extended border-list
Switch(config-ext-nacl)# no permit ip host 10.1.1.3 any
```

Being able to selectively remove lines from a named ACL is one reason you might use named ACLs instead of numbered ACLs.

After creating an ACL, you must apply it to a line or interface, as described in the "Applying an IP ACL to an Interface or Terminal Line" section on page 29-21.

Using Time Ranges with ACLs

You can selectively apply extended ACLs based on the time of day and week by using the **time-range** global configuration command. First, define a time-range name and set the times and the dates or the days of the week in the time range. Then enter the time-range name when applying an ACL to set restrictions to the access list. You can use the time range to define when the permit or deny statements in the ACL are in effect, for example, during a specified time period or on specified days of the week. The **time-range** keyword and argument are referenced in the named and numbered extended ACL task tables in the previous sections, the "Creating Standard and Extended IP ACLs" section on page 29-8 and the "Creating Named Standard and Extended IP ACLs" section on page 29-16.

These are two of the many benefits of using time ranges:

- You have more control over permitting or denying a user access to resources, such as an application (identified by an IP address/mask pair and a port number).
- You can control logging messages. ACL entries can be set to log traffic only at certain times of the day. Therefore, you can simply deny access without needing to analyze many logs generated during peak hours.

Note The time range relies on the switch system clock; therefore, you need a reliable clock source. We recommend that you use Network Time Protocol (NTP) to synchronize the switch clock. For more information, see the "Managing the System Time and Date" section on page 6-1.

Beginning in privileged EXEC mode, follow these steps to configure a time-range parameter for an ACL:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**time-range** *time-range-name*	Assign a meaningful name (for example, *workhours*) to the time range to be created, and enter time-range configuration mode. The name cannot contain a space or quotation mark and must begin with a letter.
Step 3	**absolute** [**start** *time date*] [**end** *time date*] or **periodic** *day-of-the-week hh:mm to* [*day-of-the-week*] *hh:mm* or **periodic** {**weekdays** | **weekend** | **daily**} *hh:mm to hh:mm*	Specify when the function it will be applied to is operational. • You can use only one **absolute** statement in the time range. If you configure more than one absolute statement, only the one configured last is executed • You can enter multiple **periodic** statements. For example, you could configure different hours for weekdays and weekends. See the example configurations.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show time-range**	Verify the time-range configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a configured time-range limitation, use the **no time-range** *time-range-name* global configuration command.

Repeat the steps if you have multiple items that you want operational at different times.

This example shows how to configure time ranges for *workhours* and for January 1, 2005 as a company holiday, and how to verify your configuration.

```
Switch(config)# time-range workhours
Switch(config-time-range)# periodic weekdays 8:00 to 12:00
Switch(config-time-range)# periodic weekdays 13:00 to 17:00
Switch(config-time-range)# exit
Switch(config)# time-range new_year_day_2005
Switch(config-time-range)# absolute start 00:00 1 Jan 2005 end 23:59 1 Jan 2005
```

```
Switch(config-time-range)# end
Switch# show time-range
time-range entry: new_year_day_2005 (inactive)
   absolute start 00:00 01 January 2005 end 23:59 01 January 2005
time-range entry: workhours (inactive)
   periodic weekdays 8:00 to 12:00
   periodic weekdays 13:00 to 17:00
```

For a time range to be applied, you must enter the time-range name in an extended ACL that can implement time ranges. This example shows how to create and verify extended access list 188 that denies TCP traffic from any source to any destination during the defined holiday time ranges and permits all TCP traffic during work hours.

```
Switch(config)# access-list 188 deny tcp any any time-range new_year_day_2005
Switch(config)# access-list 188 permit tcp any any time-range workhours
Switch(config)# end
Switch# show access-lists
Extended IP access list 188
    10 deny tcp any any time-range new_year_day_2005 (inactive)
    20 permit tcp any any time-range workhours (inactive)
```

This example uses named ACLs to permit and deny the same traffic.

```
Switch(config)# ip access-list extended deny_access
Switch(config-ext-nacl)# deny tcp any any time-range new_year_day_2005
Switch(config-ext-nacl)# exit
Switch(config)# ip access-list extended may_access
Switch(config-ext-nacl)# permit tcp any any time-range workhours
Switch(config-ext-nacl)# end
Switch# show ip access-lists
Extended IP access list lpip_default
    10 permit ip any any
Extended IP access list deny_access
    10 deny tcp any any time-range new_year_day_2005 (inactive)
Extended IP access list may_access
    40 permit tcp any any time-range workhours (inactive)
```

Including Comments in ACLs

You can use the **remark** keyword to include comments (remarks) about entries in any IP standard or extended ACL. The remarks make the ACL easier for you to understand and scan. Each remark line is limited to 100 characters.

The remark can go before or after a permit or deny statement. You should be consistent about where you put the remark so that it is clear which remark describes which permit or deny statement. For example, it would be confusing to have some remarks before the associated permit or deny statements and some remarks after the associated statements.

For IP numbered standard or extended ACLs, use the **access-list** *access-list number* **remark** *remark* global configuration command to include a comment about an access list. To remove the remark, use the **no** form of this command.

In this example, the workstation belonging to Jones is allowed access, and the workstation belonging to Smith is not allowed access:

```
Switch(config)# access-list 1 remark Permit only Jones workstation through
Switch(config)# access-list 1 permit 171.69.2.88
Switch(config)# access-list 1 remark Do not allow Smith workstation through
Switch(config)# access-list 1 deny 171.69.3.13
```

For an entry in a named IP ACL, use the **remark** access-list configuration command. To remove the remark, use the **no** form of this command.

In this example, the Jones subnet is not allowed to use outbound Telnet:

```
Switch(config)# ip access-list extended telnetting
Switch(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Switch(config-ext-nacl)# deny tcp host 171.69.2.88 any eq telnet
```

Applying an IP ACL to an Interface or Terminal Line

After you create an IP ACL, you can apply it to one or more interfaces or terminal lines. ACLs can be applied on *either* outbound or inbound Layer 3 interfaces, but only to inbound Layer 2 interfaces. This section describes how to accomplish this task for both terminal lines and network interfaces. Note these guidelines:

- When controlling access to a line, you must use a number. Only numbered ACLs can be applied to lines.
- When controlling access to an interface, you can use a name or number.
- Set identical restrictions on all the virtual terminal lines because a user can attempt to connect to any of them.
- If you apply an ACL to a Layer 3 interface and routing is not enabled on your switch, the ACL only filters packets that are intended for the CPU, such as SNMP, Telnet, or Web traffic. You do not have to enable routing to apply ACLs to Layer 2 interfaces.
- Port ACLs are not supported on the same switch with input router ACLs and VLAN maps.
  - If you try to apply an ACL to a Layer 2 interface on a switch that has an input Layer 3 ACL or a VLAN map applied to it, a *conflict* error message is generated. You *can* apply an ACL to a Layer 2 interface if the switch has output Layer 3 ACLs applied.
  - If you try to apply an ACL to an input Layer 3 interface on a switch that has a Layer 2 ACL applied to it, a *conflict* error message is generated. You *can* apply an ACL to an output Layer 3 interface if the switch has Layer 2 ACLs applied.
- A Layer 2 interface can have one IP access list applied to the input; a Layer 3 interface can have one IP access list applied to the input and one IP access list applied to the output. If you apply an IP ACL to an interface that already has an IP ACL configured (in that direction), the new ACL replaces the previously configured one.
- You can apply a port ACL only to a physical Layer 2 interface; you cannot apply port ACLs to EtherChannel interfaces.

Beginning in privileged EXEC mode, follow these steps to restrict incoming and outgoing connections between a virtual terminal line and the addresses in an ACL:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**line** [**console** | **vty**] *line-number*	Identify a specific line for configuration, and enter in-line configuration mode. • **console**—Enter to specify the console terminal line. The console port is DCE. • **vty**—Enter to specify a virtual terminal for remote console access. The *line-number* is the first line number in a contiguous group that you want to configure when the line type is specified. The range is from 0 to 16.

	Command	Purpose
Step 3	**access-class** *access-list-number* {**in** | **out**}	Restrict incoming or outgoing connections between a virtual terminal line (into a device) by using the conditions in the specified access list.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Display the access list configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove access restrictions on a terminal line, use the **no access-class** *access-list-number* {**in** | **out**} line configuration command.

Beginning in privileged EXEC mode, follow these steps to apply an IP access list to control access to a Layer 2 or Layer 3 interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Identify a specific interface for configuration, and enter interface configuration mode. The interface can be a Layer 2 interface (port ACL) or a Layer 3 interface (router ACL).
Step 3	**ip access-group** {*access-list-number* | *name*} {**in** | **out**}	Control access to the specified interface by using the IP access list. You can enter a standard or extended IP access number or name. **Note** The **out** keyword is not valid for Layer 2 interfaces. Port ACLs are supported only in the inbound direction.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Display the access list configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the specified access group, use the **no ip access-group** {*access-list-number* | *name*} {**in** | **out**} interface configuration command.

This example shows how to apply access list 2 on Gigabit Ethernet interface 0/3 to filter packets entering the interface:

```
Switch(config)# interface gigabitethernet0/3
Router(config-if)# ip access-group 2 in
```

Note When you apply the **ip access-group** interface configuration command to a Layer 3 interface (an SVI, a Layer 3 EtherChannel, or a routed port), the interface must have been configured with an IP address. Layer 3 access groups filter packets that are routed or are received by Layer 3 processes on the CPU. They do not affect packets bridged within a VLAN.

For inbound ACLs, after receiving a packet, the switch checks the packet against the ACL. If the ACL permits the packet, the switch continues to process the packet. If the ACL rejects the packet, the switch discards the packet.

For outbound ACLs (Layer 3 interfaces only), after receiving and routing a packet to a controlled interface, the switch checks the packet against the ACL. If the ACL permits the packet, the switch sends the packet. If the ACL rejects the packet, the switch discards the packet.

If the input interface is configured to send ICMP Unreachable messages, these messages are sent whenever a packet is discarded, regardless of whether the packet was discarded because of an ACL on the input interface or because of an ACL on the output interface. ICMP Unreachables are normally limited to no more than one every one-half second per input interface, but this can be changed by using the **ip icmp rate-limit unreachable** global configuration command.

When you apply an undefined ACL to an interface, the switch acts as if the ACL has not been applied to the interface and permits all packets. Remember this behavior if you use undefined ACLs for network security.

IP ACL Configuration Examples

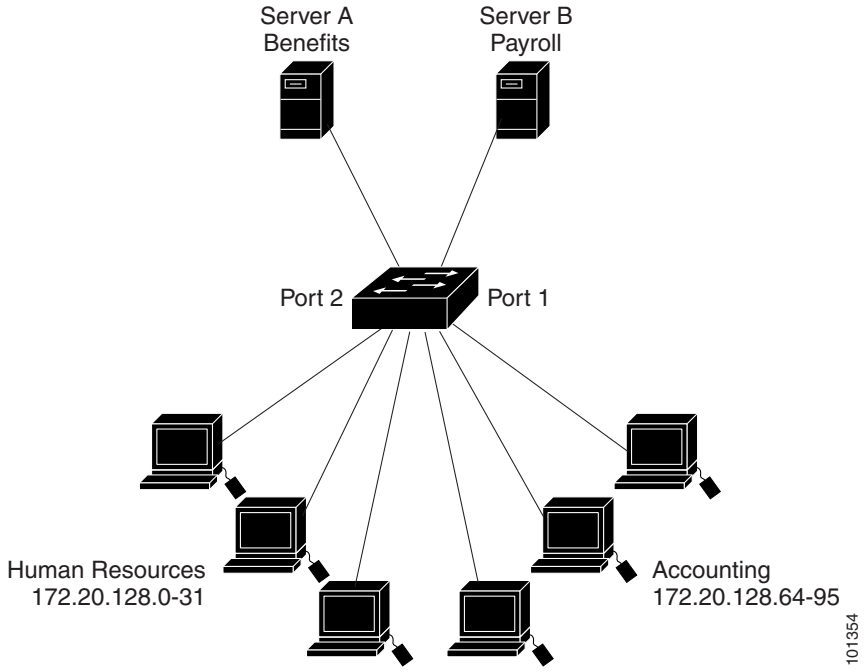
This section provides examples of configuring IP ACLs. For detailed information about compiling ACLs, see the *Cisco IOS Security Configuration Guide, Release 12.2* and to the "*Configuring IP* Services" section in the "IP Addressing and Services" chapter of the *Cisco IOS IP Configuration Guide, Release 12.2*.

Figure 29-3 shows a small networked office environment with the routed Port 2 connected to Server A, containing benefits and other information that all employees can access, and routed Port 1 connected to Server B, containing confidential payroll data. All users can access Server A, but Server B has restricted access.

Use router ACLs to do this in one of these ways:

- Create a standard IP ACL, and filter traffic coming to the server from Port 1.
- Create an extended IP ACL, and filter traffic coming from the server into Port 1.

Figure 29-3 Using Router ACLs to Control Traffic



This example uses a standard ACL to filter traffic coming into Server B from an interface, permitting traffic only from Accounting's source addresses 172.20.128.64 to 172.20.128.95. The ACL is applied to traffic coming out of routed Port 1 from the specified source address.

```
Switch(config)# access-list 6 permit 172.20.128.64 0.0.0.31
Switch(config)# end
Switch# show access-lists
Standard IP access list 6
    permit 172.20.128.64, wildcard bits 0.0.0.31
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group 6 out
```

This example uses an extended ACL to filter traffic coming from Server B into Port 1, permitting traffic from any source address (in this case Server B) to only the Accounting destination addresses 172.20.128.64 to 172.20.128.95. The ACL is then applied to traffic going into routed Port 1, permitting it to go only to the specified destination addresses. Note that with extended ACLs, you must enter the protocol (IP) before the source and destination information.

```
Switch(config)# access-list 106 permit ip any 172.20.128.64 0.0.0.31
Switch(config)# end
Switch# show access-lists
Extended IP access list 106
    permit ip any 172.20.128.64 0.0.0.31
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group 106 in
```

Numbered ACLs

In this example, network 36.0.0.0 is a Class A network whose second octet specifies a subnet; that is, its subnet mask is 255.255.0.0. The third and fourth octets of a network 36.0.0.0 address specify a particular host. Using access list 2, the switch accepts one address on subnet 48 and reject all others on that subnet. The last line of the list shows that the switch accepts addresses on all other network 36.0.0.0 subnets. The ACL is then applied to packets entering Gigabit Ethernet interface 0/1.

```
Switch(config)# access-list 2 permit 36.48.0.3
Switch(config)# access-list 2 deny 36.48.0.0 0.0.255.255
Switch(config)# access-list 2 permit 36.0.0.0 0.255.255.255
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group 2 in
```

Extended ACLs

In this example, the first line permits any incoming TCP connections with destination ports greater than 1023. The second line permits incoming TCP connections to the Simple Mail Transfer Protocol (SMTP) port of host 128.88.1.2. The third line permits incoming ICMP messages for error feedback.

```
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 gt 1023
Switch(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Switch(config)# access-list 102 permit icmp any any
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group 102 in
```

For another example of using an extended ACL, suppose that you have a network connected to the Internet, and you want any host on the network to be able to form TCP connections to any host on the Internet. However, you do not want IP hosts to be able to form TCP connections to hosts on your network, except to the mail (SMTP) port of a dedicated mail host.

SMTP uses TCP port 25 on one end of the connection and a random port number on the other end. The same port numbers are used throughout the life of the connection. Mail packets coming in from the Internet have a destination port of 25. Outbound packets have the port numbers reversed. Because the secure system behind the router always accepts mail connections on port 25, the incoming and outgoing services are separately controlled. The ACL must be configured as an input ACL on the outbound interface and an output ACL on the inbound interface.

In this example, the network is a Class B network with the address 128.88.0.0, and the mail host address is 128.88.1.2. The **established** keyword is used only for the TCP to show an established connection. A match occurs if the TCP datagram has the ACK or RST bits set, which show that the packet belongs to an existing connection. Gigabit Ethernet interface 0/1 is the interface that connects the router to the Internet.

```
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 established
Switch(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group 102 in
```

Named ACLs

The following configuration creates a standard ACL named *internet_filter* and an extended ACL named *marketing_group*. The *internet_filter* ACL allows all traffic from the source address 1.2.3.4.

```
Switch(config)# ip access-list standard Internet_filter
Switch(config-ext-nacl)# permit 1.2.3.4
Switch(config-ext-nacl)# exit
```

The *marketing_group* ACL allows any TCP Telnet traffic to the destination address and wildcard 171.69.0.0 0.0.255.255 and denies any other TCP traffic. It permits any ICMP traffic, denies UDP traffic from any source to the destination address range 171.69.0.0 through 179.69.255.255 with a destination port less than 1024, denies any other IP traffic, and provides a log of the result.

```
Switch(config)# ip access-list extended marketing_group
Switch(config-ext-nacl)# permit tcp any 171.69.0.0 0.0.255.255 eq telnet
Switch(config-ext-nacl)# deny tcp any any
Switch(config-ext-nacl)# permit icmp any any
Switch(config-ext-nacl)# deny udp any 171.69.0.0 0.0.255.255 lt 1024
Switch(config-ext-nacl)# deny ip any any log
Switch(config-ext-nacl)# exit
```

The ACLs are applied to Gigabit Ethernet port 0/5, which is configured as a Layer 3 port, with the *Internet_filter* ACL applied to incoming traffic and the *marketing_group* ACL applied to outgoing traffic.

```
Switch(config)# interface gigabitethernet0/5
Switch(config-if)# no switchport
Switch(config-if)# ip address 2.0.5.1 255.255.255.0
Switch(config-if)# ip access-group Internet_filter out
Switch(config-if)# ip access-group marketing_group in
...
```

Time Range Applied to an IP ACL

This example denies Hypertext Transfer Protocol (HTTP) traffic on IP on Monday through Friday between the hours of 8:00 a.m. and 6:00 p.m. The example allows UDP traffic only on Saturday and Sunday from noon to 8:00 p.m.

```
Switch(config)# time-range no-http
Switch(config)# periodic weekdays 8:00 to 18:00
!
```

```
Switch(config)# time-range udp-yes
Switch(config)# periodic weekend 12:00 to 20:00
!
Switch(config)# ip access-list extended strict
Switch(config-ext-nacl)# deny tcp any any eq www time-range no-http
Switch(config-ext-nacl)# permit udp any any time-range udp-yes
!
Switch(config-ext-nacl)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group strict in
```

Commented IP ACL Entries

In this example of a numbered ACL, the workstation belonging to Jones is allowed access, and the workstation belonging to Smith is not allowed access:

```
Switch(config)# access-list 1 remark Permit only Jones workstation through
Switch(config)# access-list 1 permit 171.69.2.88
Switch(config)# access-list 1 remark Do not allow Smith workstation through
Switch(config)# access-list 1 deny 171.69.3.13
```

In this example of a numbered ACL, the Winter and Smith workstations are not allowed to browse the Web:

```
Switch(config)# access-list 100 remark Do not allow Winter to browse the web
Switch(config)# access-list 100 deny host 171.69.3.85 any eq www
Switch(config)# access-list 100 remark Do not allow Smith to browse the web
Switch(config)# access-list 100 deny host 171.69.3.13 any eq www
```

In this example of a named ACL, the Jones subnet is not allowed access:

```
Switch(config)# ip access-list standard prevention
Switch(config-std-nacl)# remark Do not allow Jones subnet through
Switch(config-std-nacl)# deny 171.69.0.0 0.0.255.255
```

In this example of a named ACL, the Jones subnet is not allowed to use outbound Telnet:

```
Switch(config)# ip access-list extended telnetting
Switch(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Switch(config-ext-nacl)# deny tcp 171.69.0.0 0.0.255.255 any eq telnet
```

ACL Logging

Note Logging is not supported on port ACLs.

Two variations of logging are supported on router ACLs. The **log** keyword sends an informational logging message to the console about the packet that matches the entry; the **log-input** keyword includes the input interface in the log entry.

In this example, standard named access list *stan1* denies traffic from 10.1.1.0 0.0.0.255, allows traffic from all other sources, and includes the **log** keyword.

```
Switch(config)# ip access-list standard stan1
Switch(config-std-nacl)# deny 10.1.1.0 0.0.0.255 log
Switch(config-std-nacl)# permit any log
Switch(config-std-nacl)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group stan1 in
Switch(config-if)# end
```

```
Switch# show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Console logging: level debugging, 37 messages logged
    Monitor logging: level debugging, 0 messages logged
    Buffer logging: level debugging, 37 messages logged
    File logging: disabled
    Trap logging: level debugging, 39 message lines logged

Log Buffer (4096 bytes):

00:00:48: NTP: authentication delay calculation problems

<output truncated>

00:09:34:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
00:09:59:%SEC-6-IPACCESSLOGS:list stan1 denied 10.1.1.15 1 packet
00:10:11:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
00:15:33:%SEC-6-IPACCESSLOGS:list stan1 denied 10.1.1.15 2009 packets
```

This example is a named extended access list *ext1* that permits ICMP packets from any source to 10.1.1.0 0.0.0.255 and denies all UDP packets.

```
Switch(config)# ip access-list extended ext1
Switch(config-ext-nacl)# permit icmp any 10.1.1.0 0.0.0.255 log
Switch(config-ext-nacl)# deny udp any any log
Switch(config-std-nacl)# exit
Switch(config)# interface gigabitethernet0/3
Switch(config-if)# ip access-group ext1 in
```

This is a an example of a log for an extended IP ACL:

```
01:24:23:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 1
packet
01:25:14:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 7
packets
01:26:12:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 1
packet
01:31:33:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 8
packets
```

Note that all logging entries for IP ACLs start with `%SEC-6-IPACCESSLOG` with minor variations in format depending on the kind of ACL and the access entry that has been matched.

This is an example of an output message when the **log-input** keyword is entered:

```
00:04:21:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 (Vlan1
0001.42ef.a400) -> 10.1.1.61 (0/0), 1 packet
```

A log message for the same sort of packet using the **log** keyword does not include the input interface information:

```
00:05:47:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 -> 10.1.1.61 (0/0), 1
packet
```

Configuring Named MAC Extended ACLs

You can filter non-IP traffic on a VLAN and on a physical Layer 2 interface by using MAC addresses and named MAC extended ACLs. The procedure is similar to that of configuring other extended named ACLs. You can use a number to name the access list, but MAC access list numbers from 700 to 799 are not supported.

Note Named MAC extended ACLs cannot be applied to Layer 3 interfaces.

For more information about the supported non-IP protocols in the **mac access-list extended** command, see the command reference for this release.

Note Though visible in the command-line help strings, **appletalk** is not supported as a matching condition for the **deny** and **permit** MAC access-list configuration mode commands nor is matching on the EtherType of any SNAP-encapsulated packet with a nonzero Organizational Unique Identifier (OUI).

Beginning in privileged EXEC mode, follow these steps to create a named MAC extended ACL:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mac access-list extended** *name*	Define an extended MAC access list using a name.
Step 3	{**deny** | **permit**} {**any** | **host** *source MAC address* | *source MAC address mask*} {**any** | **host** *destination MAC address* | *destination MAC address mask*} [*type mask* | **lsap** *lsap mask* | **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** |**vines-ip** | **xns-idp** | *0-65535*] [**cos** *cos*]	In extended MAC access-list configuration mode, specify to **permit** or **deny any** source MAC address, a source MAC address with a mask, or a specific **host** source MAC address and **any** destination MAC address, destination MAC address with a mask, or a specific destination MAC address. (Optional) You can also enter these options: • *type mask*—An arbitrary EtherType number of a packet with Ethernet II or SNAP encapsulation in decimal, hex, or octal with optional mask of *don't care* bits applied to the EtherType before testing for a match. • **lsap** *lsap mask*—An LSAP number of a packet with IEEE 802.2 encapsulation in decimal, hex, or octal with optional mask of *don't care* bits. • **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** |**vines-ip** | **xns-idp**—A non-IP protocol. • **cos** *cos*—An IEEE 802.1Q cost of service number from 0 to 7 used to set priority.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show access-lists** [*number* | *name*]	Show the access list configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no mac access-list extended** *name* global configuration command to delete the entire ACL. You can also delete individual ACEs from named MAC extended ACLs.

This example shows how to create and display an access list named *mac1*, denying only EtherType DECnet Phase IV traffic, but permitting all other types of traffic.

```
Switch(config)# mac access-list extended mac1
Switch(config-ext-macl)# deny any any decnet-iv
Switch(config-ext-macl)# permit any any
Switch(config-ext-macl)# end
```

```
Switch # show access-lists
Extended MAC access list mac1
    10 deny   any any decnet-iv
    20 permit any any
```

Applying a MAC ACL to a Layer 2 Interface

After you create a MAC ACL, you can apply it to a Layer 2 interface to filter non-IP traffic coming into that interface. When you apply the MAC ACL, consider these guidelines:

- You cannot apply an ACL to a Layer 2 interface on a switch if the switch has an input Layer 3 ACL or a VLAN map applied to it. An error message is generated if you attempt to do so. You can apply an ACL to a Layer 2 interface if the switch has output Layer 3 ACLs applied.
- A Layer 2 interface can have only one MAC access list. If you apply a MAC access list to a Layer 2 interface that has a MAC ACL configured, the new ACL replaces the previously configured one.
- When a MAC ACL or VLAN filter is configured to permit MAC addresses, all control traffic, including bridge protocol data units (BPDUs) and Cisco Discovery Protocol (CDP) packets, is denied.

Beginning in privileged EXEC mode, follow these steps to apply a MAC access list to control access to a Layer 2 interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Identify a specific interface for configuration, and enter interface configuration mode. The interface must be a physical Layer 2 interface (port ACL).
Step 3	**mac access-group** {*name*} {**in**}	Control access to the specified interface by using the MAC access list. **Note** Port ACLs are supported only in the inbound direction.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show mac access-group** [**interface** *interface-id*]	Display the MAC access list applied to the interface or all Layer 2 interfaces.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the specified access group, use the **no mac access-group** {*name*} **in** interface configuration command.

This example shows how to apply MAC access list mac1 on Gigabit Ethernet interface 0/3 to filter packets entering the interface:

```
Switch(config)# interface gigabitethernet0/3
Router(config-if)# mac access-group mac1 in
```

Note The **mac access-group** interface configuration command is only valid when applied to a physical Layer 2 interface.

For inbound ACLs, after receiving a packet, the switch checks it against the ACL. If the ACL permits the packet, the switch continues to process the packet. If the ACL rejects the packet, the switch discards it.

When you apply an undefined ACL to an interface, the switch acts as if the ACL has not been applied to the interface and permits all packets. Remember this behavior if you use undefined ACLs for network security.

Configuring VLAN Maps

This section describes how to configure VLAN maps, which is the only way to control filtering within a VLAN. VLAN maps have no direction. To filter traffic in a specific direction by using a VLAN map, you need to include an ACL with specific source or destination addresses. If there is a match clause for that type of packet (IP or MAC) in the VLAN map, the default action is to drop the packet if the packet does not match any of the entries within the map. If there is no match clause for that type of packet, the default is to forward the packet.

Note For complete syntax and usage information for the commands used in this section, see the command reference for this release.

To create a VLAN map and apply it to one or more VLANs, perform these steps:

Step 1 Create the standard or extended IP ACLs or named MAC extended ACLs that you want to apply to the VLAN. See the "Creating Standard and Extended IP ACLs" section on page 29-8 and the "Configuring Named MAC Extended ACLs" section on page 29-27.

Step 2 Enter the **vlan access-map** global configuration command to create a VLAN ACL map entry.

Step 3 In access map configuration mode, optionally enter an **action**—**forward** (the default) or **drop**—and enter the **match** command to specify an IP packet or a non-IP packet (with only a known MAC address) and to match the packet against one or more ACLs (standard or extended).

Note If the VLAN map is configured with a match clause for a type of packet (IP or MAC) and the map action is drop, all packets that match the type are dropped. If the VLAN map has no match clause, and the configured action is drop, then all IP and Layer 2 packets are dropped.

Step 4 Use the **vlan filter** global configuration command to apply a VLAN map to one or more VLANs.

Note You cannot apply a VLAN map to a VLAN on a switch that has ACLs applied to Layer 2 interfaces (port ACLs).

This section contains these topics:

- VLAN Map Configuration Guidelines, page 29-31
- Creating a VLAN Map, page 29-31
- Applying a VLAN Map to a VLAN, page 29-34
- Using VLAN Maps in Your Network, page 29-34

VLAN Map Configuration Guidelines

Follow these guidelines when configuring VLAN maps:

- If there is no router ACL configured to deny traffic on a routed VLAN interface (input or output), and *no* VLAN map configured, all traffic is permitted.
- Each VLAN map consists of a series of entries. The order of entries in an VLAN map is important. A packet that comes into the switch is tested against the first entry in the VLAN map. If it matches, the action specified for that part of the VLAN map is taken. If there is no match, the packet is tested against the next entry in the map.
- If the VLAN map has at least one match clause for the type of packet (IP or MAC) and the packet does not match any of these match clauses, the default is to drop the packet. If there is no match clause for that type of packet in the VLAN map, the default is to forward the packet.
- The system might take longer to boot if you have configured a very large number of ACLs.
- For information about using both router ACLs and VLAN maps, see the "Guidelines for Using Router ACLs and VLAN Maps" section on page 29-37.
- See the "Using VLAN Maps in Your Network" section on page 29-34 for configuration examples.
- When a switch has an IP access list or MAC access list applied to a Layer 2 interface, you can create VLAN maps, but you cannot apply a VLAN map to any of the switch VLANs. An error message is generated if you attempt to do so.
- If you apply a nonexistent VLAN map to a VLAN, a warning message appears. Although you can apply a nonexistent VLAN map to a VLAN, it is not enabled until the VLAN map is defined. To avoid accidentally dropping packets and disabling connectivity in the middle of the configuration process, we recommend that you completely define the VLAN map before applying it to a VLAN.

Creating a VLAN Map

Each VLAN map consists of an ordered series of entries. Beginning in privileged EXEC mode, follow these steps to create, add to, or delete a VLAN map entry:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vlan access-map** *name* [*number*]	Create a VLAN map, and give it a name and (optionally) a number. The number is the sequence number of the entry within the map. When you create VLAN maps with the same name, numbers are assigned sequentially in increments of 10. When modifying or deleting maps, you can enter the number of the map entry that you want to modify or delete. Entering this command changes to access-map configuration mode.
Step 3	**action** {**drop** | **forward**}	(Optional) Set the action for the map entry. The default is to forward.
Step 4	**match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*]	Match the packet (using either the IP or MAC address) against one or more standard or extended access lists. Note that packets are only matched against access lists of the correct protocol type. IP packets are matched against standard or extended IP access lists. Non-IP packets are only matched against named MAC extended access lists.
Step 5	**end**	Return to global configuration mode.

	Command	Purpose
Step 6	**show running-config**	Display the access list configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no vlan access-map** *name* global configuration command to delete a map.

Use the **no vlan access-map** *name number* global configuration command to delete a single sequence entry from within the map.

Use the **no action** access-map configuration command to enforce the default action, which is to forward.

VLAN maps do not use the specific permit or deny keywords. To deny a packet by using VLAN maps, create an ACL that would match the packet, and set the action to drop. A permit in the ACL counts as a match. A deny in the ACL means no match.

Examples of ACLs and VLAN Maps

These examples show how to create ACLs and VLAN maps that for specific purposes.

Example 1

This example shows how to create an ACL and a VLAN map to deny a packet. In the first map, any packets that match the *ip1* ACL (TCP packets) would be dropped. You first create the *ip1*ACL to permit any TCP packet and no other packets. Because there is a match clause for IP packets in the VLAN map, the default action is to drop any IP packet that does not match any of the match clauses.

```
Switch(config)# ip access-list extended ip1
Switch(config-ext-nacl)# permit tcp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map_1 10
Switch(config-access-map)# match ip address ip1
Switch(config-access-map)# action drop
```

This example shows how to create a VLAN map to permit a packet. ACL *ip2* permits UDP packets and any packets that match the *ip2* ACL are forwarded.

```
Switch(config)# ip access-list extended ip2
Switch(config-ext-nacl)# permit udp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map_1 20
Switch(config-access-map)# match ip address ip2
Switch(config-access-map)# action forward
```

In this map, any IP packets that did not match any of the previous ACLs (that is, packets that are not TCP packets or UDP packets) would get dropped.

Example 2

In this example, the VLAN map has a default action of drop for IP packets and a default action of forward for MAC packets. Used with standard ACL 101 and extended named access lists **igmp-match** and **tcp-match**, the map will have the following results:

- Forward all UDP packets
- Drop all IGMP packets
- Forward all TCP packets

- Drop all other IP packets
- Forward all non-IP packets

```
Switch(config)# access-list 101 permit udp any any
Switch(config)# ip access-list extended igmp-match
Switch(config-ext-nacl)# permit igmp any any
Switch(config)# ip access-list extended tcp-match
Switch(config-ext-nacl)# permit tcp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map drop-ip-default 10
Switch(config-access-map)# match ip address 101
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-ip-default 20
Switch(config-access-map)# match ip address igmp-match
Switch(config-access-map)# action drop
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-ip-default 30
Switch(config-access-map)# match ip address tcp-match
Switch(config-access-map)# action forward
```

Example 3

In this example, the VLAN map has a default action of drop for MAC packets and a default action of forward for IP packets. Used with MAC extended access lists **good-hosts** and **good-protocols**, the map will have the following results:

- Forward MAC packets from hosts 0000.0c00.0111 and 0000.0c00.0211
- Forward MAC packets with decnet-iv or vines-ip protocols
- Drop all other non-IP packets
- Forward all IP packets

```
Switch(config)# mac access-list extended good-hosts
Switch(config-ext-macl)# permit host 000.0c00.0111 any
Switch(config-ext-macl)# permit host 000.0c00.0211 any
Switch(config-ext-nacl)# exit
Switch(config)# mac access-list extended good-protocols
Switch(config-ext-macl)# permit any any decnet-ip
Switch(config-ext-macl)# permit any any vines-ip
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map drop-mac-default 10
Switch(config-access-map)# match mac address good-hosts
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-mac-default 20
Switch(config-access-map)# match mac address good-protocols
Switch(config-access-map)# action forward
```

Example 4

In this example, the VLAN map has a default action of drop for all packets (IP and non-IP). Used with access lists **tcp-match** and **good-hosts** from Examples 2 and 3, the map will have the following results:

- Forward all TCP packets
- Forward MAC packets from hosts 0000.0c00.0111 and 0000.0c00.0211
- Drop all other IP packets
- Drop all other MAC packets

```
Switch(config)# vlan access-map drop-all-default 10
Switch(config-access-map)# match ip address tcp-match
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-all-default 20
Switch(config-access-map)# match mac address good-hosts
Switch(config-access-map)# action forward
```

Applying a VLAN Map to a VLAN

Beginning in privileged EXEC mode, follow these steps to apply a VLAN map to one or more VLANs:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vlan filter** *mapname* **vlan-list** *list*	Apply the VLAN map to one or more VLAN IDs. The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.
Step 3	**show running-config**	Display the access list configuration.
Step 4	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note You cannot apply a VLAN map to a VLAN on a switch that has ACLs applied to Layer 2 interfaces (port ACLs).

To remove the VLAN map, use the **no vlan filter** *mapname* **vlan-list** *list* global configuration command.

This example shows how to apply VLAN map 1 to VLANs 20 through 22:

```
Switch(config)# vlan filter map 1 vlan-list 20-22
```

Using VLAN Maps in Your Network

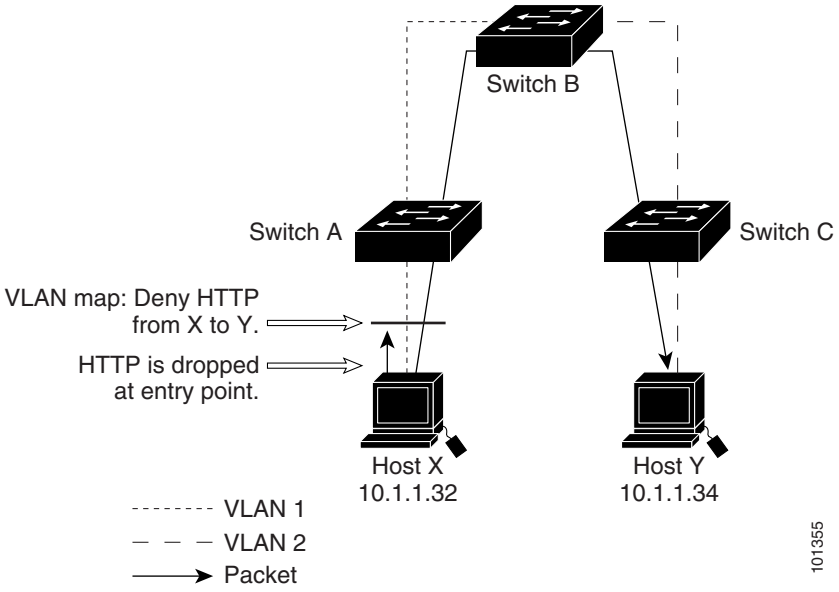
This section describes some typical uses for VLAN maps and includes these topics:

- Wiring Closet Configuration, page 29-34
- Denying Access to a Server on Another VLAN, page 29-35

Wiring Closet Configuration

In a wiring closet configuration, routing might not be enabled on the switch. In this configuration, the switch can still support a VLAN map and a QoS classification ACL. In Figure 29-4, assume that Host X and Host Y are in different VLANs and are connected to wiring closet switches A and C. Traffic from Host X to Host Y is eventually being routed by Switch B, which has routing enabled. Traffic from Host X to Host Y can be access-controlled at the traffic entry point, Switch A.

Figure 29-4 Wiring Closet Configuration



If you do not want HTTP traffic switched from Host X to Host Y, you can configure a VLAN map on Switch A to drop all HTTP traffic from Host X (IP address 10.1.1.32) to Host Y (IP address 10.1.1.34) at Switch A and not bridge it to Switch B.

First, define the IP access list *http* that permits (matches) any TCP traffic on the HTTP port.

```
Switch(config)# ip access-list extended http
Switch(config-ext-nacl)# permit tcp host 10.1.1.32 host 10.1.1.34 eq www
Switch(config-ext-nacl)# exit
```

Next, create VLAN access map *map2* so that traffic that matches the *http* access list is dropped and all other IP traffic is forwarded.

```
Switch(config)# vlan access-map map2 10
Switch(config-access-map)# match ip address http
Switch(config-access-map)# action drop
Switch(config-access-map)# exit
Switch(config)# ip access-list extended match_all
Switch(config-ext-nacl)# permit ip any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map2 20
Switch(config-access-map)# match ip address match_all
Switch(config-access-map)# action forward
```

Then, apply VLAN access map *map2*to VLAN 1.

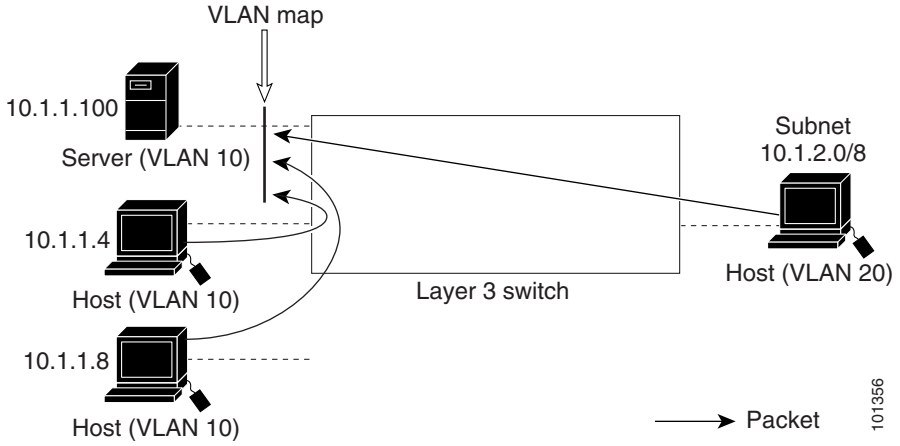
```
Switch(config)# vlan filter map2 vlan 1
```

Denying Access to a Server on Another VLAN

You can restrict access to a server on another VLAN. For example, server 10.1.1.100 in VLAN 10 needs to have access restricted as follows (see Figure 29-5):

- Hosts in subnet 10.1.2.0/8 in VLAN 20 should not have access.
- Hosts 10.1.1.4 and 10.1.1.8 in VLAN 10 should not have access.

Figure 29-5 Deny Access to a Server on Another VLAN



This example shows how to deny access to a server on another VLAN by creating the VLAN map SERVER 1 that denies access to hosts in subnet 10.1.2.0/8, host 10.1.1.4, and host 10.1.1.8 and permits other IP traffic. The final step is to apply the map SERVER1 to VLAN 10.

Step 1 Define the IP ACL that will match the correct packets.

```
Switch(config)# ip access-list extended SERVER1_ACL
Switch(config-ext-nacl))# permit ip 10.1.2.0 0.0.0.255 host 10.1.1.100
Switch(config-ext-nacl))# permit ip host 10.1.1.4 host 10.1.1.100
Switch(config-ext-nacl))# permit ip host 10.1.1.8 host 10.1.1.100
Switch(config-ext-nacl))# exit
```

Step 2 Define a VLAN map using this ACL that will drop IP packets that match SERVER1_ACL and forward IP packets that do not match the ACL.

```
Switch(config)# vlan access-map SERVER1_MAP
Switch(config-access-map)# match ip address SERVER1_ACL
Switch(config-access-map)# action drop
Switch(config)# vlan access-map SERVER1_MAP 20
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
```

Step 3 Apply the VLAN map to VLAN 10.

```
Switch(config)# vlan filter SERVER1_MAP vlan-list 10.
```

Using VLAN Maps with Router ACLs

To access control both bridged and routed traffic, you can use VLAN maps only or a combination of router ACLs and VLAN maps. You can define router ACLs on both input and output routed VLAN interfaces, and you can define a VLAN map to access control the bridged traffic.

Note You cannot combine VLAN maps or input router ACLs with port ACLs on a switch.

If a packet flow matches a VLAN-map deny clause in the ACL, regardless of the router ACL configuration, the packet flow is denied.

Note When you use router ACLs with VLAN maps, packets that require logging on the router ACLs are not logged if they are denied by a VLAN map.

If the VLAN map has a match clause for the type of packet (IP or MAC) and the packet does not match the type, the default is to drop the packet. If there is no match clause in the VLAN map, and no action specified, the packet is forwarded if it does not match any VLAN map entry.

This section includes this information about using VLAN maps with router ACLs:

- Guidelines for Using Router ACLs and VLAN Maps, page 29-37
- Examples of Router ACLs and VLAN Maps Applied to VLANs, page 29-38

Guidelines for Using Router ACLs and VLAN Maps

These guidelines are for configurations where you need to have an router ACL *and* a VLAN map on the same VLAN. These guidelines do not apply to configurations where you are mapping router ACLs and VLAN maps on different VLANs.

The switch hardware provides one lookup for security ACLs for each direction (input and output); therefore, you must merge a router ACL and a VLAN map when they are configured on the same VLAN. Merging the router ACL with the VLAN map might significantly increase the number of ACEs.

If you must configure a router ACL and a VLAN map on the same VLAN, use these guidelines for both router ACL and VLAN map configuration:

- Whenever possible, try to write the ACL with all entries having a single action except for the final, default action of the other type. That is, write the ACL using one of these two forms:

  permit...
  permit...
  permit...
  deny ip any any

  or

  deny...
  deny...
  deny...
  permit ip any any

- To define multiple actions in an ACL (permit, deny), group each action type together to reduce the number of entries.
- Avoid including Layer 4 information in an ACL; adding this information complicates the merging process. The best merge results are obtained if the ACLs are filtered based on IP addresses (source and destination) and not on the full flow (source IP address, destination IP address, protocol, and protocol ports). It is also helpful to use *don't care* bits in the IP address, whenever possible.

  If you need to specify the full-flow mode and the ACL contains both IP ACEs and TCP/UDP/ICMP ACEs with Layer 4 information, put the Layer 4 ACEs at the end of the list. This gives priority to the filtering of traffic based on IP addresses.

Note After the ACL configuration is stable for a specified interval, the system loads the configuration into hardware. Forwarding is blocked on any affected interfaces while the hardware is being updated. To change this behavior, you can use the **mls aclmerge delay** and the **access-list hardware program nonblocking** global configuration commands. For descriptions of these commands, see the command reference for this release.

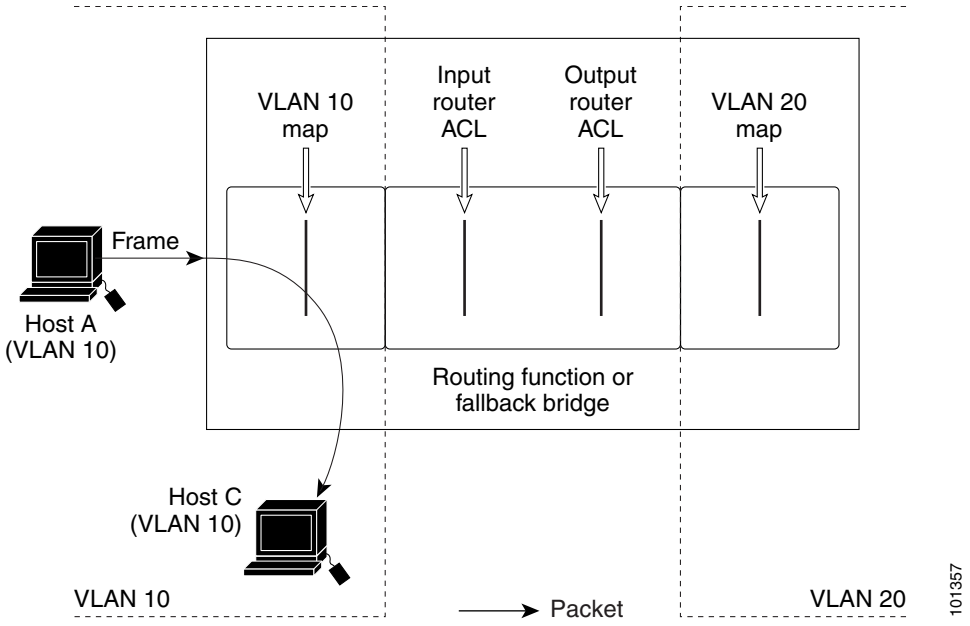
Examples of Router ACLs and VLAN Maps Applied to VLANs

This section gives examples of applying router ACLs and VLAN maps to a VLAN for switched, bridged, routed, and multicast packets. Although the following illustrations show packets being forwarded to their destination, each time the packet's path crosses a line indicating a VLAN map or an ACL, it is also possible that the packet might be dropped, rather than forwarded.

ACLs and Switched Packets

Figure 29-6 shows how an ACL is applied on packets that are switched within a VLAN. Packets switched within the VLAN without being routed or forwarded by fallback bridging are only subject to the VLAN map of the input VLAN.

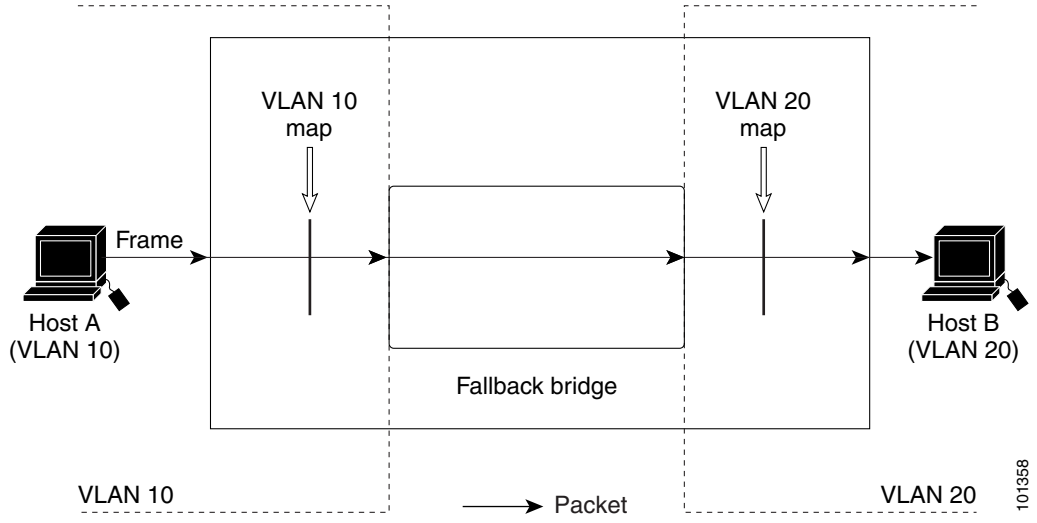
Figure 29-6 Applying ACLs on Switched Packets



ACLs and Bridged Packets

Figure 29-7 shows how an ACL is applied on fallback-bridged packets. For bridged packets, only Layer 2 ACLs are applied to the input VLAN. Only non-IP, non-ARP packets can be fallback-bridged.

Figure 29-7 Applying ACLs on Bridged Packets

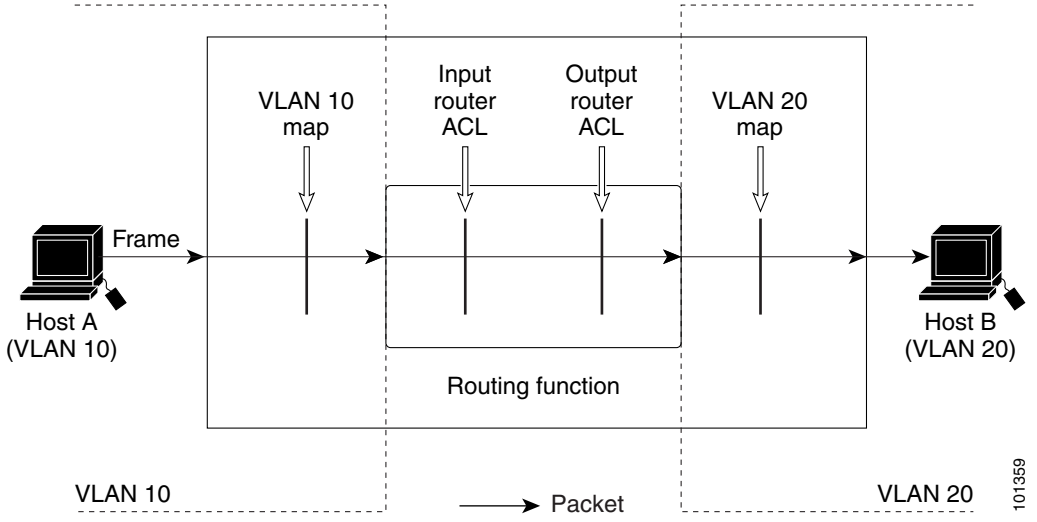


ACLs and Routed Packets

Figure 29-8 shows how ACLs are applied on routed packets. For routed packets, the ACLs are applied in this order:

1. VLAN map for input VLAN
2. Input router ACL
3. Output router ACL
4. VLAN map for output VLAN

Figure 29-8 Applying ACLs on Routed Packets

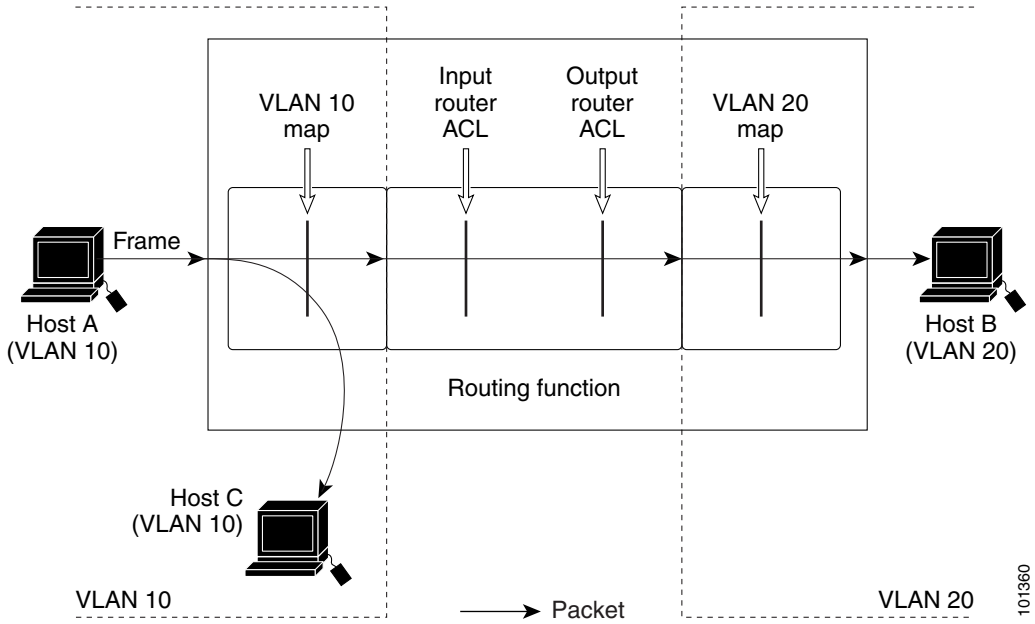


ACLs and Multicast Packets

Figure 29-9 shows how ACLs are applied on packets that are replicated for IP multicasting. A multicast packet being routed has two different kinds of filters applied: one for destinations that are other ports in the input VLAN and another for each of the destinations that are in other VLANs to which the packet has been routed. The packet might be routed to more than one output VLAN, in which case a different router output ACL and VLAN map would apply for each destination VLAN.

The final result is that the packet might be permitted in some of the output VLANs and not in others. A copy of the packet is forwarded to those destinations where it is permitted. However, if the input VLAN map (VLAN 10 map in Figure 29-9) drops the packet, no destination receives a copy of the packet.

Figure 29-9 Applying ACLs on Multicast Packets



Displaying ACL Information

You can display the ACLs that are configured on the switch, and you can display the ACLs that have been applied to interfaces and VLANs. You can also display information about configuration conflicts or resource usage related to ACLs.

This section includes these topics:

- Displaying ACL Configuration, page 29-41
- Displaying ACL Resource Usage and Configuration Problems, page 29-42

Displaying ACL Configuration

You can display existing ACLs and when you use the **ip access-group** interface configuration command to apply ACLs to a Layer 2 or 3 interface, you can display the access groups on the interface. You can also display the MAC ACLs applied to a Layer 2 interface. You can use the privileged EXEC commands as described in Table 29-2 to display this information.

Table 29-2 Commands for Displaying Access Lists and Access Groups

Command	Purpose
show access-lists [*number* | *name*]	Display the contents of one or all current IP and MAC address access lists or a specific access list (numbered or named).
show ip access-lists [*number* | *name*]	Display the contents of all current IP access lists or a specific IP access list (numbered or named).
show ip interface *interface-id*	Display detailed configuration and status of an interface. If IP is enabled on the interface and ACLs have been applied by using the **ip access-group** interface configuration command, the access groups are included in the display.
show running-config [**interface** *interface-id*]	Displays the contents of the configuration file for the switch or the specified interface, including all configured MAC and IP access lists and which access groups are applied to an interface.
show mac access-group [**interface** *interface-id*]	Displays MAC access lists applied to all Layer 2 interfaces or the specified Layer 2 interface.

This is an example of output from the **show access-lists** privileged EXEC command, displaying all standard and extended ACLs:

```
Switch# show access-lists
Standard IP access list 1
    permit 172.20.10.10
Standard IP access list 10
    permit 12.12.12.12
Standard IP access list 12
    deny   1.3.3.2
Standard IP access list 32
    permit 172.20.20.20
Standard IP access list 34
    permit 10.24.35.56
    permit 23.45.56.34
Extended IP access list 120
    permit eigrp host 12.3.6.5 host 25.36.1.24
Extended MAC access list mac1
```

This is an example of output from the **show ip access-lists** privileged EXEC command. It displays only IP standard and extended ACLs. Note that the named MAC extended ACL displayed in the previous example is not included in this display.

```
Switch# show ip access-lists
Standard IP access list 1
    permit 172.20.10.10
Standard IP access list 10
    permit 12.12.12.12
Standard IP access list 12
    deny   1.3.3.2
Standard IP access list 32
    permit 172.20.20.20
Standard IP access list 34
```

```
    permit 10.24.35.56
    permit 23.45.56.34
Extended IP access list 120
    permit eigrp host 12.3.6.5 host 25.36.1.24
```

This is an example of output from the **show mac access-group** privileged EXEC command when only one interface (Gigabit Ethernet interface 2) has a MAC access list (*macl-e1*) applied.

```
Switch# show mac access-group
Interface GigabitEthernet0/1:
   Inbound access-list is not set
Interface GigabitEthernet0/2:
   Inbound access-list is macl_e1
Interface GigabitEthernet0/3:
   Inbound access-list is not set
Interface GigabitEthernet0/4:
   Inbound access-list is not set
Interface GigabitEthernet0/5:
   Inbound access-list is not set

<output truncated>
```

You can also display information about VLAN access maps or VLAN filters. Use the privileged EXEC commands in Table 29-3 to display VLAN map information.

Table 29-3 Commands for Displaying VLAN Map Information

Command	Purpose
show vlan access-map [*mapname*]	Show information about all VLAN access-maps or the specified access map.
show vlan filter [**access-map** *name* | **vlan** *vlan-id*]	Show information about all VLAN filters or about a specified VLAN or VLAN access map.

This is an example of output from the **show vlan access-map** privileged EXEC command:

```
Switch# show vlan access-map
Vlan access-map "map_1" 10
  Match clauses:
     ip address: ip1
  Action:
     drop
Vlan access-map "map_1" 20
  Match clauses:
     mac address: mac1
  Action:
     forward
```

This is an example of output from the **show vlan filter** privileged EXEC command:

```
Switch# show vlan filter
VLAN Map map_1 is filtering VLANs:
  20-22
```

Displaying ACL Resource Usage and Configuration Problems

The switch feature manager allocates resources to configured ACLs. When there are not enough hardware resources for a configuration or when there is a configuration conflict, an error message is generated. If the console is not set to receive error messages, you can use the **show fm** privileged EXEC

commands to display feature-manager messages and to get more information about the resources handling ACLs on an interface. You can also use the **show tcam** privileged EXEC commands to get status information about the switch ternary content addressable memory (TCAM) capacity.

Table 29-4 lists the privileged EXEC commands that display ACL feature-manager information.

Table 29-4 Commands for Displaying VLAN Map Information

Command	Purpose
show fm vlan *vlan-id* or **show fm interface** *interface-id*	Display feature-manager information for the interface or the VLAN, including the hardware port-label or vlan-label number for the interface and feature-manager problems that have occurred.
show fm vlan-label *label-id* or **show fm port-label** *label-id*	Display information about the identified label, including which of the configured ACL features fit into hardware. VLAN labels are used for router ACLs and VLAN maps; port labels are used for port ACLs. The VLAN *label-id* range is from 0 to 255; the port *label-id* range is from 0 to 127.
show tcam {**inacl** | **outacl**} *tcam-id* {{**port-labels** [*label-id*]} | **size** |{**statistics** [**entries** | **hits** | **labels** | **masks**]} | {**vlan-labels** [*label-id*]}}	Display information about the input or output ACL regions of TCAM. The TCAM ID range varies from 1 to 3, depending on the switch model. Other keywords available for the command are used primarily to display output for use by Cisco technical support.

For more detailed information about these commands, see the command reference for this release.

This section describes how to display this information about these ACL issues:

- Configuration Conflicts, page 29-43
- ACL Configuration Fitting in Hardware, page 29-44
- TCAM Usage, page 29-47

Configuration Conflicts

If you attempt to enter an ACL configuration that is not allowed, for example, applying a port ACL to an interface on a switch that has router ACLs already configured, an error message is logged.

In this example, Gigabit port 1 is a Layer 2 interface. When you try to apply access list *ip3*, the error message shows that there are already ACLs applied to Layer 3 interfaces on the switch.

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip access-group ip3 in
Switch(config-if)#
  1d18h:%FM-3-CONFLICT:Port ACL ip3 conflicts with input router ACLs
```

You can enter the **show fm interface** privileged EXEC command for an interface to determine if there are ACL configuration conflicts or to learn the port-label number for the port. You can then enter the **show fm port-label** privileged EXEC command to display more details, as shown in this example:

```
Switch# show fm interface gigabitethernet0/1
Conflicts exist with layer 3 access groups.
Input Port Label:2
Switch# show fm port-label 2
Conflicts exist with layer 3 access groups.
Needed in CAM(s):1
```

```
Loaded into CAM(s):1
Sent to CPU by CAM(s):
Interfaces: Gi0/1
IP Access Group:ip3 0 VMRs
DHCP Broadcast Suppression Disabled.
MAC Access Group:(None) 0 VMRs
```

This example shows the result of trying to apply ACL 121 to an SVI, VLAN 1, when the switch already has ACLs applied to Layer 2 interfaces.

```
Switch(config)# interface vlan 1
Switch(config-if)# ip access-group 121 in
Switch(config-if)#
  1d18h:%FM-3-CONFLICT:Input router ACL 121 conflicts with port ACLs
```

You can enter the **show fm vlan** privileged EXEC command for a VLAN to display the conflict and to determine the VLAN *label-ids*, and then enter the **show fm vlan-label** command for more information.

```
Switch# show fm vlan 1
Conflicts exist with layer 2 access groups.
Input VLAN Label:1
Output VLAN Label:0 (default)
Priority:normal
Switch# show fm vlan-label 1
Conflicts exist with layer 2 access groups.
Input Features:
  Interfaces or VLANs: Vl1
  Priority:normal
  Vlan Map:(none)
  Access Group:121, 0 VMRs
  Multicast Boundary:(none), 0 VMRs
Output Features:
  Interfaces or VLANs:
  Priority:low
  Bridge Group Member:no
  Vlan Map:(none)
  Access Group:(none), 0 VMRs
```

ACL Configuration Fitting in Hardware

As previously stated, ACL processing in the Catalyst 3550 switch is mostly accomplished in hardware. However, if the hardware reaches its capacity to store ACL configurations, the switch software attempts to fit a simpler configuration into the hardware. This simpler configuration does not do all the filtering that has been configured, but instead sends some or all packets to the CPU to be filtered by software. In this way, all configured filtering will be accomplished, but performance is greatly decreased when the filtering is done in software.

For example, if the combination of an input router ACL applied to a VLAN interface and a VLAN map applied to the same VLAN does not fit into the hardware, these results might occur:

- If the VLAN map alone fits in hardware, the software sets up the hardware to send to the CPU all packets that need to be routed for filtering and possible routing (if the packet passes the filter). Packets that only require bridging within the input VLAN are still handled entirely by hardware and not sent to the CPU.
- If the VLAN map does not fit in the hardware, all packets on that VLAN must be both filtered and forwarded by software.

Any problem in fitting the configuration into hardware is logged. You can use the **show fm** privileged EXEC commands to determine if any interface configuration or VLAN configuration did not fit into hardware.

Port ACL Examples

This is an example of a port access list that is too big for the available TCAM space.

```
Switch(config-if)# interface gigabitethernet0/3
Switch(config-if)# ip access-group 100 in
Switch(config-if)#
00:04:58:%FM-3-UNLOADING:Unloading port label 3 feature from TCAM 1
```

To verify the port label or to see if a label was assigned to an interface, you can enter the **show fm interface** command.

```
Switch# show fm interface gigabitethernet0/3
Input Port Label:3
```

Entering the **show fm port-label 3** privileged EXEC command shows that label 3 is needed in CAM 1 but that it is not loaded in CAM 1; instead, it is sent to the CPU.

```
Switch# show fm port-label 3
Needed in CAM(s):1
Loaded into CAM(s):
Sent to CPU by CAM(s):1
Interfaces: Gi0/3
IP Access Group:100 3400 VMRs
DHCP Broadcast Suppression Disabled.
MAC Access Group:(None) 2 VMRs
```

The number of TCAMs on a switch varies (from 1 to 3) with switch model. On switches that have more than one TCAM, if the same port ACL has been applied to several interfaces, it is possible that the configuration fits into some, but not all, of the required TCAMs. In that case, a log message generated when the ACL is applied specifies which TCAM was unable to load the ACL.

```
Switch(config)# interface gigabitethernet0/10
Switch(config-if)# ip access-group 101 in
Switch(config-if)#
01:46:25:%FM-3-UNLOADING:Unloading port label 4 feature from TCAM 1
```

When you enter the **show fm port-label** command for label 4, the display shows which TCAMs have the feature loaded and which do not:

```
Switch# show fm port-label 4
Needed in CAM(s):1 3
Loaded into CAM(s):3
Sent to CPU by CAM(s):1
Interfaces: Gi0/3, Gi0/10
IP Access Group:101 379 VMRs
DHCP Broadcast Suppression Disabled.
MAC Access Group:(None) 2 VMRs
```

The display shows that port label 4 is needed in CAMs 1 and 3, but did not fit into CAM 1, because in this case CAM 1 already contained entries for other port labels and had less available space than CAM 3. The output shows that the label is loaded into CAM 3 and that CAM 1 sends packets on this label to the CPU because the entries for the port ACLs on port label 4 have been unloaded from CAM 1.

VLAN or Router ACL Examples

This example shows how to display the feature manager information for VLAN 1:

```
Switch# show fm vlan 1
Input VLAN Label:1
Output VLAN Label:0 (default)
Priority:normal
```

This output from the **show fm vlan-label** privileged EXEC command shows a merge failure on an input access group:

```
Switch# show fm vlan-label 1
Unloaded due to merge failure or lack of space:
  InputAccessGroup
  Merge Fail:input
Input Features:
  Interfaces or VLANs: Vl1
  Priority:normal
  Vlan Map:(none)
  Access Group:131, 6788 VMRs
  Multicast Boundary:(none), 0 VMRs
Output Features:
  Interfaces or VLANs:
  Priority:low
  Bridge Group Member:no
  Vlan Map:(none)
  Access Group:(none), 0 VMRs
```

This output from the **show fm vlan-label** privileged EXEC command shows insufficient room for an input access group in the hardware:

```
Switch# show fm vlan-label 1
Unloaded due to merge failure or lack of space:
  InputAccessGroup
Input Features:
  Interfaces or VLANs: Vl1
  Priority:normal
  Vlan Map:(none)
  Access Group:bigone, 11 VMRs
  Multicast Boundary:(none), 0 VMRs
Output Features:
  Interfaces or VLANs:
  Priority:low
  Bridge Group Member:no
  Vlan Map:(none)
  Access Group:(none), 0 VMRs
```

This output from the **show fm vlan-label** privileged EXEC command shows not enough room for the input access group or the output access group on the label. (Note that the access groups were configured on two different interfaces. Labels are assigned independently for input and output.)

```
Switch# show fm label 1
Unloaded due to merge failure or lack of space:
  InputAccessGroup OutputAccessGroup
Input Features:
  Interfaces or VLANs: Vl1
  Priority:normal
  Vlan Map:(none)
  Access Group:bigone, 11 VMRs
  Multicast Boundary:(none), 0 VMRs
Output Features:
  Interfaces or VLANs: Vl2
  Priority:normal
  Bridge Group Member:no
  Vlan Map:(none)
  Access Group:bigtwo, 11 VMRs
```

Note When configuring ACLs on the switch, to allocate maximum hardware resources for ACLs, you can use the **sdm prefer access** global configuration command to set the Switch Database Management feature to the access template. For more information on the SDM templates, see the "Optimizing System Resources for User-Selected Features" section on page 6-26.

TCAM Usage

You can display the remaining capacity in a TCAM before or after configuring ACLs, and you can also display how much space is allotted in the TCAM to a particular interface or VLAN by using the **show tcam** privileged EXEC commands.

You can use the **show tcam size** to display the total size of the regions of TCAM in which the ACLs are entered.

```
Switch# show tcam inacl 1 size
Ingress ACL TCAM Size:6592 Entries
```

To change the amount allocated to various TCAM regions, use the **sdm prefer** global configuration command to allocate more resources to ACLs, routing, or Layer 2 switching.

The **show tcam statistics** command for an input or output TCAM region displays how full that region is, including allocated and available masks and entries. This is an example of the output from the command:

```
Switch# show tcam inacl 1 statistics
Ingress ACL TCAM#1:Number of active labels:3
Ingress ACL TCAM#1:Number of masks   allocated:  14, available: 810
Ingress ACL TCAM#1:Number of entries allocated:  17, available:6575
```

To determine how much of the TCAM is being used by ACL configuration on an interface or VLAN, use the **show fm interface** or **show fm vlan** command to determine the port label or vlan label being used for the port or VLAN ACL configuration. Then use the **show tcam port-label** or **show tcam vlan-label** command to display how much TCAM space is allocated to the label. VLAN labels are used for router ACLs and VLAN maps. Port labels are used for port ACLs.

```
Switch# show fm vlan 1
Input VLAN Label:1
Output VLAN Label:0 (default)
Priority:normal
Switch# show tcam inacl 1 vlan-labels 1
Label Value :      8193(vlan label 1)
Number of entries :779
Entry List
----------
Mask Index :4
F7 00 00 00 00 00 00 00 00 80 FF C0 00 C0 FF FF 00 00
Entry Index :32  Timestamp:1
96 00 00 00 00 00 00 00 00 80 01 40 00 80 00 01 00 00 As Data(hex) :00260086
Mask Index :5
F7 00 00 00 00 00 00 00 00 80 FF C0 00 C0 00 00 FF FF
Entry Index :33  Timestamp:4
96 00 00 00 00 00 00 00 00 80 01 40 00 80 00 00 00 B3 As Data(hex) :00260086
Mask Index :6
F5 00 00 00 00 E0 00 00 00 80 FF C0 00 C0 00 00 00 00
Entry Index :48  Timestamp:1
94 00 00 00 00 E0 00 00 00 80 01 40 00 80 00 00 00 00 As Data(hex) :00210086
Mask Index :7
F7 00 00 00 00 00 00 00 00 80 FF C0 00 C0 00 00 00 00
Entry Index :49  Timestamp:4
```

```
96 00 00 00 00 00 00 00 00 80 01 40 00 80 00 00 00 00 As Data(hex) :00210086
Mask Index :8
F5 00 00 00 00 00 00 00 00 80 FF C0 00 C0 00 00 00 00
Entry Index :64  Timestamp:1

<output truncated>
```

Note In the **show tcam vlan-label** output, the *Number of entries* field does not account for the two default entries and therefore omits two entries from the count. Default entries are not used for port labels, so the field is accurate for that output.

CHAPTER 30

Configuring EtherChannels

This chapter describes how to configure EtherChannel on the Layer 2 and Layer 3 interfaces of a Catalyst 3550 switch.

This chapter consists of these sections:

- Understanding EtherChannels, page 30-1
- EtherChannel On Mode, page 30-6
- Configuring EtherChannels, page 30-7
- Displaying EtherChannel, PAgP, and LACP Status, page 30-18

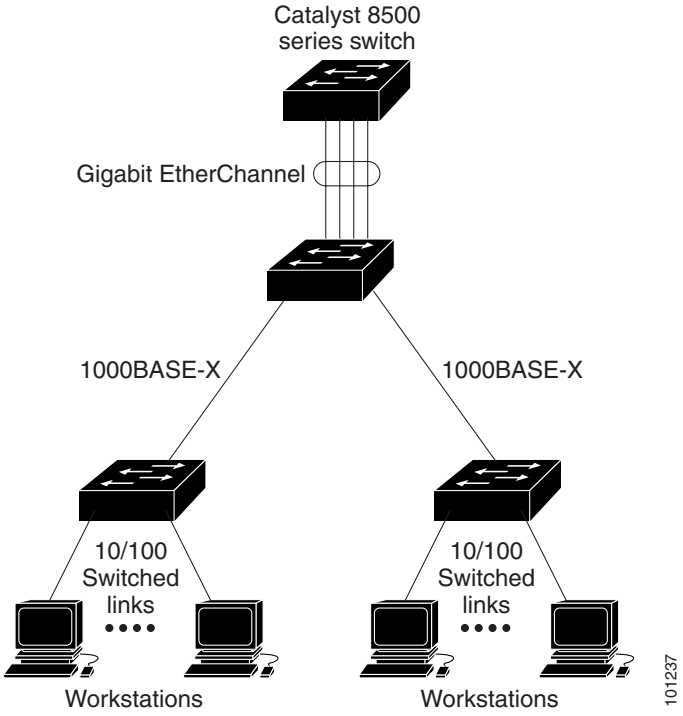
Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

Understanding EtherChannels

EtherChannel provides fault-tolerant high-speed links between switches, routers, and servers. You can use it to increase the bandwidth among the wiring closets and the data center, and you can deploy it anywhere in the network where bottlenecks are likely to occur. EtherChannel provides automatic recovery for the loss of a link by redistributing the load across the remaining links. If a link fails, EtherChannel redirects traffic from the failed link to the remaining links in the channel without intervention.

An EtherChannel consists of individual Fast Ethernet or Gigabit Ethernet links bundled into a single logical link as shown in Figure 30-1. The EtherChannel provides full-duplex bandwidth up to 800 Mbps (Fast EtherChannel) or 8 Gbps (Gigabit EtherChannel) between your switch and another switch or host.

Figure 30-1 Typical EtherChannel Configuration



Each EtherChannel can consist of up to eight compatibly configured Ethernet interfaces. All interfaces in each EtherChannel must be the same speed, and all must be configured as either Layer 2 or Layer 3 interfaces.

Note The network device to which your switch is connected can impose its own limits on the number of interfaces in the EtherChannel. For Catalyst 3550 switches, the number of EtherChannels is limited to the number of ports of the same type.

You can configure an EtherChannel in one of these modes: Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), or On mode. Configure both ends of the EtherChannel in the same mode:

- When you configure one end of an EtherChannel in either PAgP or LACP mode, the system negotiates with the other end of the channel to determine which ports should become active. Incompatible ports are suspended.
- When you configure an EtherChannel in the **on** mode, no negotiations take place. The switch forces all compatible ports to become active in the EtherChannel. The other end of the channel (on the other switch) must also be configured in the **on** mode; otherwise, packet loss can occur.

If a link within an EtherChannel fails, traffic previously carried over that failed link moves to the remaining links within the EtherChannel. If traps are enabled on the switch, a trap is sent for a failure that identifies the switch, the EtherChannel, and the failed link. Inbound broadcast and multicast packets on one link in an EtherChannel are blocked from returning on any other link of the EtherChannel.

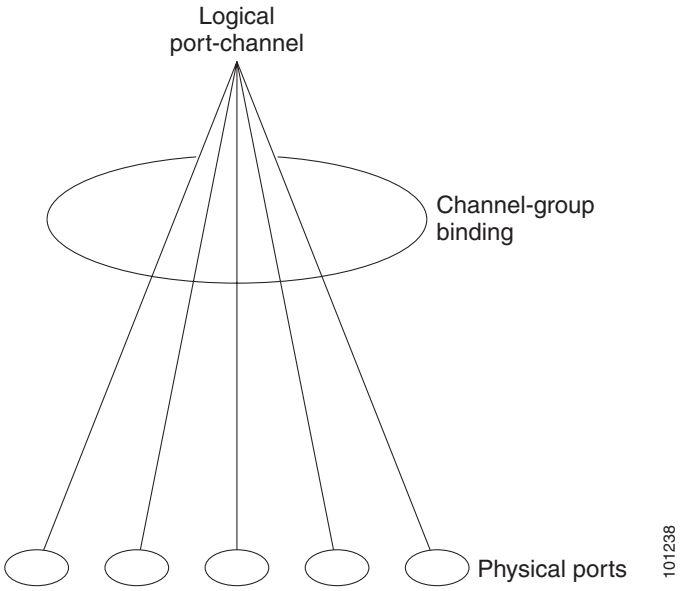
Understanding Port-Channel Interfaces

You create an EtherChannel for Layer 2 interfaces differently from Layer 3 interfaces. Both configurations involve logical interfaces.

- With Layer 3 interfaces, you manually create the logical interface by using the **interface port-channel** global configuration command.
- With Layer 2 interfaces, the logical interface is dynamically created.
- With both Layer 3 and 2 interfaces, you manually assign an interface to the EtherChannel by using the **channel-group** interface configuration command. This command binds the physical and logical ports together as shown in Figure 30-2.

Each EtherChannel has a logical port-channel interface numbered from 1 to 64. The channel groups are also numbered from 1 to 64.

Figure 30-2 Relationship of Physical Ports, Logical Port Channels, and Channel Groups



When a port joins an EtherChannel, the physical interface for that port is shut down. When the port leaves the port-channel, its physical interface is brought up, and it has the same configuration as it had before joining the EtherChannel.

Note Configuration changes made to the logical interface of an EtherChannel may not propagate to all the member ports of the channel.

Understanding the Port Aggregation Protocol and Link Aggregation Protocol

The Port Aggregation Protocol (PAgP) and Link Aggregation Control Protocol (LACP) facilitate the automatic creation of EtherChannels by exchanging packets between Ethernet interfaces. PAgP is a Cisco-proprietary protocol that can be run only on Cisco switches and on those switches licensed by licensed vendors to support PAgP. LACP is defined in IEEE 802.3ad and allows Cisco switches to manage Ethernet channels between switches that conform to the IEEE 802.3ad protocol.

By using one of these protocols, a switch learns the identity of partners capable of supporting either PAgP or LACP and learns the capabilities of each interface. It then dynamically groups similarly configured interfaces into a single logical link (channel or aggregate port); these interfaces are grouped based on hardware, administrative, and port parameter constraints. For example, PAgP groups the interfaces with the same speed, duplex mode, native VLAN, VLAN range, and trunking status and type. After grouping the links into an EtherChannel, PAgP adds the group to the spanning tree as a single switch port.

PAgP and LACP Modes

Table 30-1 shows the user-configurable EtherChannel modes for the **channel-group** interface configuration command. Switch interfaces exchange PAgP packets only with partner interfaces configured in the **auto** or **desirable** modes. Switch interfaces exchange LACP packets only with partner interfaces configured in the **active** or **passive** modes. Interfaces configured in the **on** mode do not exchange PAgP or LACP packets.

Table 30-1 EtherChannel Modes

Mode	Description
active	Places an interface into an active negotiating state, in which the interface starts negotiations with other interfaces by sending LACP packets.
auto	Places an interface into a passive negotiating state, in which the interface responds to PAgP packets it receives but does not start PAgP packet negotiation. This setting minimizes the transmission of PAgP packets.
desirable	Places an interface into an active negotiating state, in which the interface starts negotiations with other interfaces by sending PAgP packets.
passive	Places an interface into a passive negotiating state, in which the interface responds to LACP packets that it receives, but does not start LACP packet negotiation. This setting minimizes the transmission of LACP packets.

Exchanging PAgP Packets

Both the **auto** and **desirable** PAgP modes allow interfaces to negotiate with partner interfaces to determine if they can form an EtherChannel based on criteria such as interface speed and, for Layer 2 EtherChannels, trunking state and VLAN numbers.

Interfaces can form an EtherChannel when they are in different PAgP modes as long as the modes are compatible. For example:

- An interface in the **desirable** mode can form an EtherChannel with another interface that is in the **desirable** or **auto** mode.
- An interface in the **auto** mode can form an EtherChannel with another interface in the **desirable** mode.

An interface in the **auto** mode cannot form an EtherChannel with another interface that is also in the **auto** mode because neither interface starts PAgP negotiation.

If your switch is connected to a partner that is PAgP-capable, you can configure the switch interface for nonsilent operation by using the **non-silent** keyword. If you do not specify **non-silent** with the **auto** or **desirable** mode, silent mode is assumed.

The silent mode is used when the switch is connected to a device that is not PAgP-capable and seldom, if ever, sends packets. An example of a silent partner is a file server or a packet analyzer that is not generating traffic. In this case, running PAgP on a physical port connected to a silent partner prevents that switch port from ever becoming operational; however, the silent setting allows PAgP to operate, to attach the interface to a channel group, and to use the interface for transmission.

Note An EtherChannel cannot be configured in both the PAgP and LACP modes.

Exchanging LACP Packets

Both the **active** and **passive LACP** modes allow interfaces to negotiate with partner interfaces to determine if they can form an EtherChannel based on criteria such as interface speed and, for Layer 2 EtherChannels, trunking state, and VLAN numbers.

Interfaces can form an EtherChannel when they are in different LACP modes as long as the modes are compatible. For example:

- An interface in the **active** mode can form an EtherChannel with another interface that is in the **active** mode.
- An interface in the **active** mode can form an EtherChannel with another interface in the **passive** mode.

An interface in the **passive** mode cannot form an EtherChannel with another interface that is also in the **passive** mode because neither interface starts LACP negotiation.

Note An EtherChannel cannot be configured in both the PAgP and LACP modes.

Physical Learners and Aggregate-Port Learners

Network devices are classified as PAgP physical learners or aggregate-port learners. A device is a physical learner if it learns addresses by physical ports and directs transmissions based on that knowledge. A device is an aggregate-port learner if it learns addresses by aggregate (logical) ports.

When a device and its partner are both aggregate-port learners, they learn the address on the logical port-channel. The device sends packets to the source by using any of the interfaces in the EtherChannel. With aggregate-port learning, it is not important on which physical port the packet arrives.

PAgP cannot automatically detect when the partner device is a physical learner and the local device is an aggregate-port learner. Therefore, you must manually set the learning method on the local device or source-based distribution by using the **pagp learn-method** interface configuration command. With source-based distribution, any given source-MAC address is always sent on the same physical port.

You can also configure a single interface within the group for all transmissions and use other interfaces for hot standby. The unused interfaces in the group can be swapped into operation in just a few seconds if the selected single interface loses hardware-signal detection. You can configure which interface is always selected for packet transmission by changing its priority by using the **pagp port-priority** interface configuration command. The higher the priority, the more likely that the port will be selected.

PAgP and LACP Interaction with Other Features

The Dynamic Trunking Protocol (DTP) and Cisco Discovery Protocol (CDP) send and receive packets over the physical interfaces in the EtherChannel. Trunk ports send and receive PAgP and LACP protocol data units (PDUs) on the lowest numbered VLAN.

Spanning tree sends packets over the first interface in the EtherChannel.

The MAC address of a Layer 3 EtherChannel is the MAC address of the first interface in the port-channel.

PAgP sends and receives PAgP PDUs only from interfaces that have PAgP enabled for the auto or desirable mode. LACP sends and receives LACP PDUs only from interfaces that have LACP enabled for the active or passive mode.

EtherChannel On Mode

EtherChannel **on** mode can be used to manually configure an EtherChannel. The **on** mode forces a port to join an EtherChannel without negotiations. It can be useful if the remote device does not support PAgP or LACP. With the **on** mode, a usable EtherChannel exists only when both ends of the link are configured in the **on** mode.

Ports that are configured with **on** mode in the same channel group must have compatible port characteristics, such as speed and duplex. Ports that are not compatible are suspended, even though they are configured with **on** mode.

Caution You should exercise care when using the **on** mode. This is a manual configuration, and ports on both ends of the EtherChannel must have a similar configuration. If the group is misconfigured, packet loss or spanning-tree loops can occur.

Understanding Load Balancing and Forwarding Methods

EtherChannel balances the traffic load across the links in a channel by randomly associating a newly learned MAC address with one of the links in the channel.

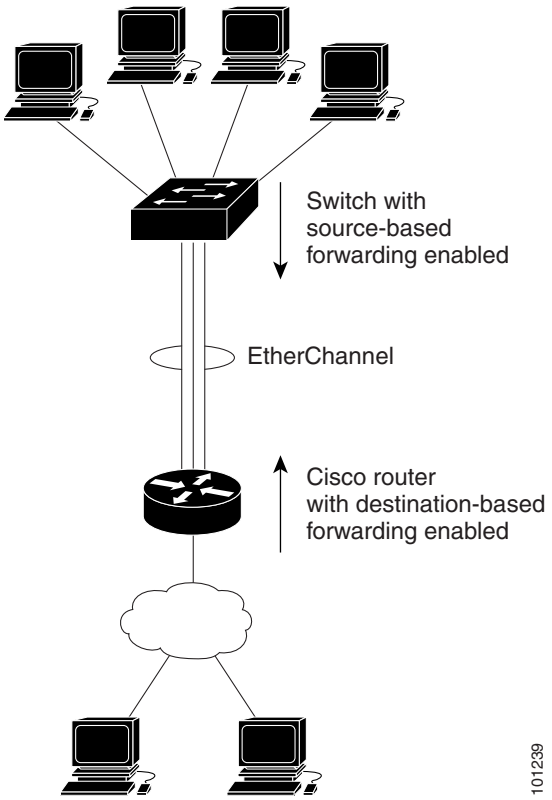
With source-MAC address forwarding, packets forwarded to an EtherChannel are distributed across the ports in the channel based on the source-MAC address of the incoming packet. Therefore, to provide load balancing, packets from different hosts use different ports in the channel, but packets from the same host use the same port in the channel. The MAC address learned by the switch does not change).

When source-MAC address forwarding is used, load distribution based on the source and destination IP address is also enabled for routed IP traffic. All routed IP traffic chooses a port based on the source and destination IP address. Packets between two IP hosts always use the same port in the channel, and traffic between any other pair of hosts can use a different port in the channel.

With destination-MAC address forwarding, packets forwarded to an EtherChannel are distributed across the ports in the channel based on the destination host MAC address of the incoming packet. Therefore, packets to the same destination are forwarded over the same port, and packets to a different destination might be sent on a different port in the channel. You configure the load balancing and forwarding method by using the **port-channel load-balance** global configuration command.

In Figure 30-3, multiple workstations are connected to a switch, and an EtherChannel connects the switch to the router. Source-based load balancing is used on the switch end of the EtherChannel to ensure that the switch efficiently uses the bandwidth of the router by distributing traffic from the workstation across the physical links. Since the router is a single MAC address device, it uses destination-based load balancing to efficiently spread the traffic to the workstations across the physical links in the EtherChannel.

Figure 30-3 Load Distribution and Forwarding Methods



Use the option that provides the greatest variety in your configuration. For example, if the traffic on a channel is going only to a single MAC address, using the destination-MAC address always chooses the same link in the channel; using source addresses or IP addresses might result in better load balancing.

Configuring EtherChannels

These sections describe how to configure EtherChannel on Layer 2 and Layer 3 interfaces:

- Default EtherChannel Configuration, page 30-8
- EtherChannel Configuration Guidelines, page 30-8
- Configuring Layer 2 EtherChannels, page 30-9
- Configuring Layer 3 EtherChannels, page 30-12
- Configuring EtherChannel Load Balancing, page 30-15
- Configuring the PAgP Learn Method and Priority, page 30-15

Note Make sure that the interfaces are correctly configured (see the "EtherChannel Configuration Guidelines" section on page 30-8).

Note After you configure an EtherChannel, configuration changes applied to the port-channel interface apply to all the physical interfaces assigned to the port-channel interface, and configuration changes applied to the physical interface affect only the interface where you apply the configuration.

Default EtherChannel Configuration

Table 30-2 shows the default EtherChannel configuration.

Table 30-2 Default EtherChannel Configuration

Feature	Default Setting
Channel groups	None assigned.
Layer 3 port-channel logical interface	None defined.
PAgP mode	No default.
PAgP learn method	Aggregate-port learning on all interfaces.
PAgP priority	128 on all interfaces.
LACP learn method	Aggregate-port learning on all interfaces.
LACP priority	32768 on all interfaces.
Load balancing	Load distribution on the switch is based on the source-MAC address of the incoming packet. Load distribution based on the source and destination IP address is also enabled for routed IP traffic.

EtherChannel Configuration Guidelines

If improperly configured, some EtherChannel ports are automatically disabled to avoid network loops and other problems. Follow these guidelines to avoid configuration problems:

- Configure an EtherChannel with up to eight Ethernet ports of the same type.

  Note Do not configure a GigaStack GBIC port as part of an EtherChannel.

- Configure all ports in an EtherChannel to operate at the same speeds and duplex modes.
- Enable all interfaces in an EtherChannel. A port in an EtherChannel that is disabled by using the **shutdown** interface configuration command is treated as a link failure, and its traffic is transferred to one of the remaining interfaces in the EtherChannel.
- When a group is first created, all ports follow the parameters set for the first port to be added to the group. If you change the configuration of one of these parameters, you must also make the changes to all ports in the group:
  - Allowed-VLAN list
  - Spanning-tree path cost for each VLAN
  - Spanning-tree port priority for each VLAN
  - Spanning-tree Port Fast setting

- An EtherChannel interface that is configured as a Switched Port Analyzer (SPAN) destination port does not join the group until it is deconfigured as a SPAN destination port.
- Do not configure a secure port as part of an EtherChannel or the reverse.
- Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an IEEE 802.1x port. If you try to enable IEEE 802.1x on an EtherChannel port, an error message appears, and IEEE 802.1x is not enabled.

Note In software releases earlier than Cisco IOS Release 12.2(25)SE, if 802.1x is enabled on a not-yet-active port of an EtherChannel, the port does not join the EtherChannel.

- If EtherChannels are configured on switch interfaces, remove the EtherChannel configuration from the interfaces before globally enabling IEEE 802.1x on a switch by using the **dot1x system-auth-control** global configuration command.
- For Layer 2 EtherChannels:
  - Assign all interfaces in the EtherChannel to the same VLAN, or configure them as trunks. Interfaces with different native VLANs cannot form an EtherChannel.
  - If you configure an EtherChannel from trunk interfaces, verify that the trunking mode (ISL or 802.1Q) is the same on all the trunks. Inconsistent trunk modes on EtherChannel interfaces can have unexpected results.
  - An EtherChannel supports the same allowed range of VLANs on all the interfaces in a trunking Layer 2 EtherChannel. If the allowed range of VLANs is not the same, the interfaces do not form an EtherChannel even when PAgP is set to the **auto** or **desirable** mode.
  - Interfaces with different spanning-tree path costs can form an EtherChannel if they are otherwise compatibly configured. Setting different spanning-tree path costs does not, by itself, make interfaces incompatible for the formation of an EtherChannel.
- For Layer 3 EtherChannels, assign the Layer 3 address to the port-channel logical interface, not to the physical interfaces in the channel.

Configuring Layer 2 EtherChannels

You configure Layer 2 EtherChannels by configuring the Ethernet interfaces with the **channel-group** interface configuration command, which creates the port-channel logical interface. You cannot put a Layer 2 interface into a manually created port-channel interface.

Note Layer 2 interfaces must be connected and functioning for the software to create port-channel interfaces.

Beginning in privileged EXEC mode, follow these steps to assign a Layer 2 Ethernet interface to a Layer 2 EtherChannel:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify a physical interface to configure, and enter interface configuration mode. Valid interfaces include physical interfaces. Up to eight interfaces of the same type and speed can be configured for the same group.
Step 3	**switchport mode** {**access** | **trunk**} **switchport access vlan** *vlan-id*	Assign all interfaces as static-access ports in the same VLAN, or configure them as trunks. If you configure the interface as a static-access port, assign it to only one VLAN. The range is 1 to 4094.

	Command	Purpose
Step 4	**channel-group** *channel-group-number* **mode** {{**auto** [**non-silent**] | **desirable** [**non-silent**] | **on**} | {**active** | **passive**}}	Assign the port to a channel group, and specify the PAgP or LACP mode. For *channel-group-number*, the range is 1 to 64. Each EtherChannel can have up to eight compatibly configured Ethernet interfaces. For **mode**, select one of these keywords: • **auto**—Enables PAgP only if a PAgP device is detected. It places an interface into a passive negotiating state, in which the interface responds to PAgP packets it receives but does not start PAgP packet negotiation. • **desirable**—Unconditionally enables PAgP. It places an interface into an active negotiating state, in which the interface starts negotiations with other interfaces by sending PAgP packets. • **on**—Forces the interface to channel without PAgP. With the **on** mode, a usable EtherChannel exists only when an interface group in the **on** mode is connected to another interface group in the **on** mode. • **non-silent**—If your switch is connected to a partner that is PAgP-capable, you can configure the switch interface for nonsilent operation. You can configure an interface with the **non-silent** keyword for use with the **auto** or **desirable** mode. If you do not specify **non-silent** with the **auto** or **desirable** mode, silent is assumed. The silent setting is for connections to file servers or packet analyzers. This setting allows PAgP to operate, to attach the interface to a channel group, and to use the interface for transmission. • **active—Enables LACP only if an LACP device is detected. It places an interface into an active negotiating state, in which the interface starts negotiations with other interfaces by sending LACP packets.** • **passive**—Enables LACP on an interface and places it into a passive negotiating state, in which the interface responds to LACP packets that it receives, but does not start LACP packet negotiation. For information on compatible PAgP and LACP modes for the switch and its partner, see the "PAgP and LACP Modes" section on page 30-4.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a port from the EtherChannel group, use the **no channel-group** interface configuration command.

This example shows how to assign a range of interfaces as static-access ports in VLAN 10 to channel 5 with the PAgP mode **desirable**:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet0/1 -2
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
Switch(config-if-range)# channel-group 5 mode desirable
Switch(config-if-range)# end
```

Configuring Layer 3 EtherChannels

To configure Layer 3 EtherChannels, you create the port-channel logical interface and then put the Ethernet interfaces into the port-channel as described in the next two sections.

Creating Port-Channel Logical Interfaces

When configuring Layer 3 EtherChannels, you must manually create the port-channel logical interface first by using the **interface port-channel** global configuration command. Then, you put the logical interface into the channel group by using the **channel-group** interface configuration command.

Note To move an IP address from a physical interface to an EtherChannel, you must delete the IP address from the physical interface before configuring it on the port-channel interface.

Beginning in privileged EXEC mode, follow these steps to create a port-channel interface for a Layer 3 EtherChannel:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface port-channel** *port-channel-number*	Create the port-channel logical interface, and enter interface configuration mode. For *port-channel-number*, the range is 1 to 64.
Step 3	**no switchport**	Put the interface into Layer 3 mode.
Step 4	**ip address** *ip-address mask*	Assign an IP address and subnet mask to the EtherChannel.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show etherchannel** *channel-group-number* **detail**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.
Step 8		Assign an Ethernet interface to the Layer 3 EtherChannel. For more information, see the "Configuring the Physical Interfaces" section on page 30-13.

To remove the port-channel, use the **no interface port-channel** *port-channel-number* global configuration command.

This example shows how to create the logical port channel (5) and assign 172.10.20.10 as its IP address:

```
Switch# configure terminal
Switch(config)# interface port-channel 5
Switch(config-if)# no switchport
Switch(config-if)# ip address 172.10.20.10 255.255.255.0
Switch(config-if)# end
```

Configuring the Physical Interfaces

Beginning in privileged EXEC mode, follow these steps to assign an Ethernet interface to a Layer 3 EtherChannel:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify a physical interface to configure, and enter interface configuration mode. Valid interfaces include physical interfaces. Up to eight interfaces of the same type and speed can be configured for the same group.
Step 3	**no ip address**	Ensure that there is no IP address assigned to the physical interface.

	Command	Purpose
Step 4	**channel-group** *channel-group-number* **mode** {**auto** [**non-silent**] | **desirable** [**non-silent**] | **on** | **active** | **passive**}	Assign the interface to a channel group, and specify the PAgP or LACP mode. For *channel-group-number*, the range is 1 to 64. This number must be the same as the *port-channel-number* (logical port) configured in the "Creating Port-Channel Logical Interfaces" section on page 30-12. Each EtherChannel can consist of up to eight compatibly configured Ethernet interfaces. For **mode**, select one of these keywords: • **active—Enables LACP only if an LACP device is detected. It places an interface into an active negotiating state, in which the interface starts negotiations with other interfaces by sending LACP packets.** • **auto**—Enables PAgP only if a PAgP device is detected. It places an interface into a passive negotiating state, in which the interface responds to PAgP packets it receives but does not start PAgP packet negotiation. • **desirable**—Unconditionally enables PAgP. It places an interface into an active negotiating state, in which the interface starts negotiations with other interfaces by sending PAgP packets. • **non-silent**—If your switch is connected to a partner that is PAgP capable, you can configure the switch interface for nonsilent operation. You can configure an interface with the **non-silent** keyword for use with the **auto** or **desirable** mode. If you do not specify **non-silent** with the **auto** or **desirable** mode, silent is assumed. The silent setting is for connections to file servers or packet analyzers; this setting allows PAgP to operate, to attach the interface to a channel group, and to use the interface for transmission. • **on**—Forces the interface to channel without PAgP or LACP. With the **on** mode, a usable EtherChannel exists only when an interface group in the **on** mode is connected to another interface group in the **on** mode. • **passive**—Enables LACP on an interface and places it into a passive negotiating state, in which the interface responds to LACP packets that it receives, but does not start LACP packet negotiation. For information on compatible PAgP modes for the switch and its partner, see the "PAgP and LACP Modes" section on page 30-4.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an interface from the EtherChannel group, use the **no channel-group** interface configuration command.

This example shows how to assign interfaces 1 and 2 to channel 5 with the PAgP mode **desirable**:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet0/1 -2
Switch(config-if-range)# no ip address
Switch(config-if-range)# channel-group 5 mode desirable
Switch(config-if-range)# end
```

Configuring EtherChannel Load Balancing

This section describes how to configure EtherChannel load balancing by using source-based or destination-based forwarding methods. For more information, see the "Understanding Load Balancing and Forwarding Methods" section on page 30-6.

Beginning in privileged EXEC mode, follow these steps to configure EtherChannel load balancing:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**port-channel load-balance** *method*	Configure an EtherChannel load-balancing *method* value: • **src-mac**—Load distribution using the source-MAC address. • **dst-mac**—Load distribution using the destination-MAC address. The default is **src-mac**. When **src-mac** is used, load distribution based on the source and destination IP address is also enabled. For all IP traffic being routed, the switch chooses a port for transmission based on the source and destination IP address. Packets between two IP hosts always use the same port for packet transmission, but packets between any other pair of hosts might use a different transmission port.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show etherchannel load-balance**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return EtherChannel load balancing to the default configuration, use the **no port-channel load-balance** global configuration command.

Configuring the PAgP Learn Method and Priority

Network devices are classified as PAgP physical learners or aggregate-port learners. A device is a physical learner if it learns addresses by physical ports and directs transmissions based on that knowledge. A device is an aggregate-port learner if it learns addresses by aggregate ports.

For compatibility with Catalyst 1900 series switches, configure the PAgP learning method on the Catalyst 3550 switches to learn source-MAC addresses on the physical port. The switch then sends packets to the Catalyst 1900 switch using the same interface in the EtherChannel from which it learned the source address.

Note The switch supports address learning only on aggregate ports even though the **physical-port** keyword is provided in the CLI. The **pagp learn-method** command and the **pagp port-priority** command have no effect on the switch hardware, but they are required for PAgP interoperability with devices that only support address learning by physical ports, such as the Catalyst 1900 switch.

When the link partner to the Catalyst 3550 switch is a physical learner, we recommend that you configure the switch as a physical-port learner by using the **pagp learn-method physical-port** interface configuration command and to set the load-distribution method based on the source-MAC address by using the **port-channel load-balance src-mac** global configuration command. Use the **pagp learn-method** command only in this situation.

Beginning in privileged EXEC mode, follow these steps to configure your switch as a PAgP physical-port learner and to adjust the priority so that the same port in the bundle is selected for sending packets:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface for transmission, and enter interface configuration mode.
Step 3	**pagp learn-method** physical-port	Select the PAgP learning method. By default, **aggregation-port learning** is selected, which means the switch sends packets to the source by using any of the interfaces in the EtherChannel. With aggregate-port learning, it is not important on which physical port the packet arrives. Select **physical-port** to connect with another switch that is a physical learner. Make sure to configure the **port-channel load-balance** global configuration command to **src-mac** as described in the "Configuring EtherChannel Load Balancing" section on page 30-15. The learning method must be configured the same at both ends of the link.
Step 4	**pagp port-priority** *priority*	Assign a priority so that the selected interface is chosen for packet transmission. For *priority*, the range is 0 to 255. The default is 128. The higher the priority, the more likely that the interface will be used for PAgP transmission.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config** or **show pagp** *channel-group-number* **internal**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the priority to its default setting, use the **no pagp port-priority** interface configuration command. To return the learning method to its default setting, use the **no pagp learn-method** interface configuration command.

Configuring the LACP Port Priority

You can set the priority for each port in an EtherChannel that is configured for LACP by using the **lacp port-priority** privileged EXEC command. The range is from 1 to 65535. Beginning in privileged EXEC mode, follow these steps to configure the LACP port priority:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface for transmission, and enter interface configuration mode.
Step 3	**lacp port-priority** ***priority-value***	Select the LACP port priority value. For priority-value, the range is 1 to 65535. By default, **the priority value is 32768. The lower the range, the more likely that the interface will be used for LACP transmission.**
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config** or **show lacp** *channel-group-number* **internal**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring Hot Standby Ports

When enabled, LACP tries to configure the maximum number of LACP-compatible ports in a channel, up to a maximum of 16 ports. Only eight LACP links can be active at one time. Any additional links are put in a hot standby state. If one of the active links becomes inactive, a link that is in hot standby mode becomes active in its place.

If more than eight links are configured for an EtherChannel group, the software determines which of the hot standby ports to make active based on:

- LACP port-priority
- Port ID

All ports default to the same port priority. You can change the port priority of LACP EtherChannel ports to specify which hot standby links become active first by using the **lacp port-priority interface configuration command to set the port priority to a value lower than the default of 32768.**

The hot standby ports that have lower port numbers become active in the channel first unless the port priority is configured to be a lower number than the default value of **32768.**

Note If LACP is not able to aggregate all the ports that are compatible (for example, the remote system might have more restrictive hardware limitations), all the ports that cannot be actively included in the EtherChannel are put in hot standby state and are used only if one of the channeled ports fails.

Configuring the LACP System Priority

You can set the system priority for all of the EtherChannels that are configured for LACP by using the **lacp system-priority** privileged EXEC command. The range is from 1 to 65535.

Note The **lacp system-priority** command is global. You cannot set a system priority for each LACP-configured channel separately.

We recommend using this command only when there are a combination of LACP-configured EtherChannels that are in both **active** and **standby** modes.

Beginning in privileged EXEC mode, follow these steps to configure the LACP system priority:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**lacp system-priority** ***priority-value***	Select the LACP system priority value. For priority-value, the range is 1 to 65535. By default, **the priority value is 32768. The lower the range, the higher the system priority. The switch with the lower system priority value determines which links between LACP partner switches are active and which are in standby for each LACP EtherChannel.**
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config** or **show lacp** *channel-group-number* **internal**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Displaying EtherChannel, PAgP, and LACP Status

You can use the privileged EXEC commands described in Table 30-3 to display EtherChannel, PAgP, and LACP status information:

Table 30-3 Commands for Displaying EtherChannel, PAgP, and LACP Status

Command	Description
show etherchannel [*channel-group-number*] {**detail** | **load-balance** | **port** | **port-channel** | **summary**}	Displays EtherChannel information in a detailed and one-line summary form. Also displays the load-balance or frame-distribution scheme, port, and port-channel information.
show pagp [*channel-group-number*] {**counters** | **internal** | **neighbor**}[1]	Displays PAgP information such as traffic information, the internal PAgP configuration, and neighbor information.

Table 30-3 Commands for Displaying EtherChannel, PAgP, and LACP Status (continued)

Command	Description
show lacp [*channel-group-number*] {**counters** | **internal** | **neighbor**}[2]	Displays LACP information such as traffic information, the internal PAgP configuration, and neighbor information.

1. You can clear PAgP channel-group information and traffic filters by using the **clear pagp** {*channel-group-number* [**counters**] \| **counters**} privileged EXEC command.
2. You can clear LACP channel-group information and traffic filters by using the **clear lacp** {*channel-group-number* [**counters**] \| **counters**} privileged EXEC command.

For detailed information about the fields in the command outputs, see the command reference for this release.

CHAPTER 31

Configuring IP Unicast Routing

This chapter describes how to configure IP unicast routing on your Catalyst 3550 multilayer switch. Beginning with Cisco IOS Release 12.1(11)EA1, basic routing functions, including static unicast routing and the Routing Information Protocol (RIP), are available with both the IP base image (formerly known as the standard multilayer software image [SMI]) and the IP services image (formerly known as the enhanced multilayer software image [EMI]). To use advanced routing features and other routing protocols, or for all routing support prior to Cisco IOS Release 12.1(11)EA1, you must have the IP services image installed on your switch.

For more detailed IP unicast configuration information, see the *Cisco IOS IP Configuration Guide, Release 12.2* For complete syntax and usage information for the commands used in this chapter, see these command references:

- *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2*
- *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*
- *Cisco IOS IP Command Reference, Volume 3 of 3: Multicast, Release 12.2*

This chapter consists of these sections:

- Understanding IP Routing, page 31-2
- Steps for Configuring Routing, page 31-3
- Configuring IP Addressing on Layer 3 Interfaces, page 31-4
- Enabling IP Unicast Routing, page 31-18
- Configuring RIP, page 31-19
- Configuring OSPF, page 31-24
- Configuring EIGRP, page 31-34
- Configuring BGP, page 31-40
- Configuring Multi-VRF CE, page 31-61
- Configuring Protocol-Independent Features, page 31-72
- Monitoring and Maintaining the IP Network, page 31-84

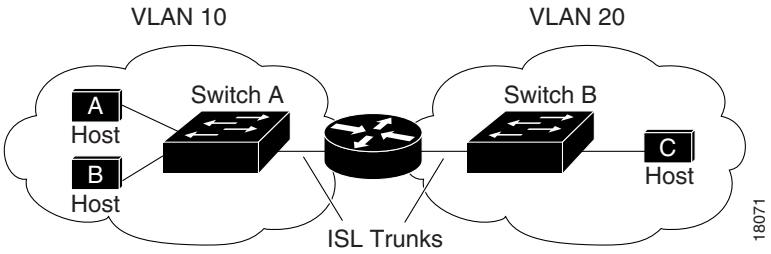
Note When configuring routing parameters on the switch, to allocate system resources to maximize the number of unicast routes allowed, you can use the **sdm prefer routing** global configuration command to set the Switch Database Management (sdm) feature to the routing template. For more information on the SDM templates, see the "Optimizing System Resources for User-Selected Features" section on page 6-26.

Understanding IP Routing

In some network environments, VLANs are associated with individual networks or subnetworks. In an IP network, each subnetwork is mapped to an individual VLAN. Configuring VLANs helps control the size of the broadcast domain and keeps local traffic local. However, network devices in different VLANs cannot communicate with one another without a Layer 3 device (router) to route traffic between the VLAN, referred to as inter-VLAN routing. You configure one or more routers to route traffic to the appropriate destination VLAN.

Figure 31-1 shows a basic routing topology. Switch A is in VLAN 10, and Switch B is in VLAN 20. The router has an interface in each VLAN.

Figure 31-1 Routing Topology Example



When Host A in VLAN 10 needs to communicate with Host B in VLAN 10, it sends a packet addressed to that host. Switch A forwards the packet directly to Host B, without sending it to the router.

When Host A sends a packet to Host C in VLAN 20, Switch A forwards the packet to the router, which receives the traffic on the VLAN 10 interface. The router checks the routing table, determines the correct outgoing interface, and forwards the packet on the VLAN 20 interface to Switch B. Switch B receives the packet and forwards it to Host C.

Routers can perform unicast routing in three different ways:

- By using default routing
- By using preprogrammed static routes for the traffic
- By dynamically calculating routes by using a routing protocol

Default routing refers to sending traffic with a destination unknown to the router to a default outlet or destination.

Static unicast routing forwards packets from predetermined ports through a single path into and out of a network. Static routing is secure and uses little bandwidth, but does not automatically respond to changes in the network, such as link failures, and therefore, might result in unreachable destinations. As networks grow, static routing becomes a labor-intensive liability.

Dynamic routing protocols are used by routers to dynamically calculate the best route for forwarding traffic. There are two types of dynamic routing protocols:

- Routers using distance-vector protocols maintain routing tables with distance values of networked resources, and periodically pass these tables to their neighbors. Distance-vector protocols use one or a series of metrics for calculating the best routes. These protocols are easy to configure and use.
- Routers using link-state protocols maintain a complex database of network topology, based on the exchange of link-state advertisements (LSAs) between routers. LSAs are triggered by an event in the network, which speeds up the convergence time or time required to respond to these changes. Link-state protocols respond quickly to topology changes, but require greater bandwidth and more resources than distance-vector protocols.

Distance-vector protocols supported by the switch are Routing Information Protocol (RIP), which uses a single distance metric (cost) to determine the best path and Border Gateway Protocol (BGP), which adds a path vector mechanism. The switch also supports the Open Shortest Path First (OSPF) link-state protocol and Enhanced Interior Gateway Routing Protocol (EIGRP), which adds some link-state routing features to traditional IGRP to improve efficiency.

Note The IP base image supports only default routing, static routing, and RIP. All other routing protocols require the IP services image on your switch.

Steps for Configuring Routing

By default, IP routing is disabled on the switch, and you must enable it before routing can take place. For detailed IP routing configuration information, see the *Cisco IOS IP Configuration Guide, Release 12.2*.

In the following procedures, the specified interface must be one of these Layer 3 interfaces:

- A routed port: a physical port configured as a Layer 3 port by using the **no switchport** interface configuration command.
- A switch virtual interface (SVI): a VLAN interface created by using the **interface vlan** *vlan_id* global configuration command and by default a Layer 3 interface.
- An EtherChannel port channel in Layer 3 mode: a port-channel logical interface created by using the **interface port-channel** *port-channel-number* global configuration command and binding the Ethernet interface into the channel group. For more information, see the "Configuring Layer 3 EtherChannels" section on page 30-12.

Note The switch does not support tunnel interfaces for unicast routed traffic.

All Layer 3 interfaces must have IP addresses assigned to them. See the "Assigning IP Addresses to Network Interfaces" section on page 31-5.

Note A Layer 3 switch can have an IP address assigned to each routed port and SVI. The number of routed ports and SVIs that you can configure is not limited by software. However, the interrelationship between this number and the number and volume of features being implemented might have an impact on CPU utilization because of hardware limitations. For more information about feature combinations, see the "Optimizing System Resources for User-Selected Features" section on page 6-26.

Configuring routing consists of several main procedures:

- To support VLAN interfaces, create and configure VLANs on the switch, and assign VLAN membership to Layer 2 interfaces. For more information, see Chapter 11, "Configuring VLANs."
- Configure Layer 3 interfaces.
- Enable IP routing on the switch.
- Assign IP addresses to the Layer 3 interfaces.
- Enable selected routing protocols on the switch.
- Configure routing protocol parameters (optional).

Configuring IP Addressing on Layer 3 Interfaces

A required task for configuring IP routing is to assign IP addresses to Layer 3 network interfaces to enable the interfaces and allow communication with the hosts on those interfaces that use IP. These sections describe how to configure various IP addressing features. Assigning IP addresses to the interface is required; the other procedures are optional.

- Default Addressing Configuration, page 31-4
- Assigning IP Addresses to Network Interfaces, page 31-5
- Configuring Address Resolution Methods, page 31-8
- Routing Assistance When IP Routing is Disabled, page 31-11
- Configuring Broadcast Packet Handling, page 31-13
- Monitoring and Maintaining IP Addressing, page 31-17

Default Addressing Configuration

Table 31-1 shows the default addressing configuration.

Table 31-1 Default Addressing Configuration

Feature	Default Setting
IP address	None defined.
ARP	No permanent entries in the Address Resolution Protocol (ARP) cache. Encapsulation: Standard Ethernet-style ARP. Timeout: 14400 seconds (4 hours).
IP broadcast address	255.255.255.255 (all ones).
IP classless routing	Enabled.
IP default gateway	Disabled.
IP directed broadcast	Disabled (all IP directed broadcasts are dropped).
IP domain	Domain list: No domain names defined. Domain lookup: Enabled. Domain name: Enabled.
IP forward-protocol	If a helper address is defined or User Datagram Protocol (UDP) flooding is configured, UDP forwarding is enabled on default ports. Any-local-broadcast: Disabled. Spanning Tree Protocol (STP): Disabled. Turbo-flood: Disabled.
IP helper address	Disabled.
IP host	Disabled.

Table 31-1 Default Addressing Configuration (continued)

Feature	Default Setting
IRDP	Disabled. Defaults when enabled: • Broadcast IRDP advertisements. • Maximum interval between advertisements: 600 seconds. • Minimum interval between advertisements: 0.75 times max interval • Preference: 0.
IP proxy ARP	Enabled.
IP routing	Disabled.
IP subnet-zero	Disabled.

Assigning IP Addresses to Network Interfaces

An IP address identifies a location to which IP packets can be sent. Some IP addresses are reserved for special uses and cannot be used for host, subnet, or network addresses. Table 31-2 lists ranges of IP addresses and shows which are reserved and which are available for use. RFC 1166, “Internet Numbers,” contains the official description of IP addresses.

Table 31-2 Reserved and Available IP Addresses

Class	Address or Range	Status
A	0.0.0.0 1.0.0.0 to 126.0.0.0 127.0.0.0	Reserved Available Reserved
B	128.0.0.0 to 191.254.0.0 191.255.0.0	Available Reserved
C	192.0.0.0 192.0.1.0 to 223.255.254 223.255.255.0	Reserved Available Reserved
D	224.0.0.0 to 239.255.255.255	Multicast group addresses
E	240.0.0.0 to 255.255.255.254 255.255.255.255	Reserved Broadcast

An interface can have one primary IP address. A mask identifies the bits that denote the network number in an IP address. When you use the mask to subnet a network, the mask is referred to as a subnet mask. To receive an assigned network number, contact your Internet service provider.

Beginning in privileged EXEC mode, follow these steps to assign an IP address and a network mask to a Layer 3 interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**no switchport**	Remove the interface from Layer 2 configuration mode (if it is a physical interface).
Step 4	**ip address** *ip-address subnet-mask*	Configure the IP address and IP subnet mask.
Step 5	**no shutdown**	Enable the interface.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces** [*interface-id*] **show ip interface** [*interface-id*] **show running-config interface** [*interface-id*]	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip address** interface configuration command to remove an IP address or to disable IP processing.

This example shows how to configure an IP address on and enable Gigabit Ethernet interface 0/10:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface gigabitethernet0/10
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.1.2.3 255.255.0.0
Switch(config-if)# no shutdown
```

Use of Subnet Zero

Subnetting with a subnet address of zero is strongly discouraged because of the problems that can arise if a network and a subnet have the same addresses. For example, if network 131.108.0.0 is subnetted as 255.255.255.0, subnet zero would be written as 131.108.0.0, which is the same as the network address.

You can use the all ones subnet (131.108.255.0) and even though it is discouraged, you can enable the use of subnet zero if you need the entire subnet space for your IP address.

Beginning in privileged EXEC mode, follow these steps to enable subnet zero:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip subnet-zero**	Enable the use of subnet zero for interface addresses and routing updates.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entry.
Step 5	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

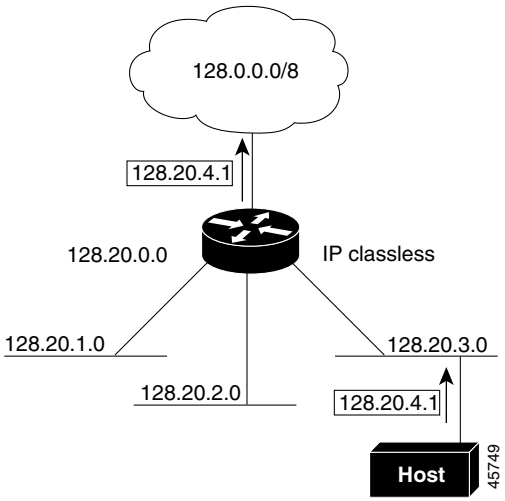
Use the **no ip subnet-zero** global configuration command to restore the default and disable the use of subnet zero.

Classless Routing

By default, classless routing behavior is enabled on the switch when it is configured to route. With classless routing, if a router receives packets for a subnet of a network with no default route, the router forwards the packet to the best supernet route. A *supernet* consists of contiguous blocks of Class C address spaces used to simulate a single, larger address space and is designed to relieve the pressure on the rapidly depleting Class B address space.

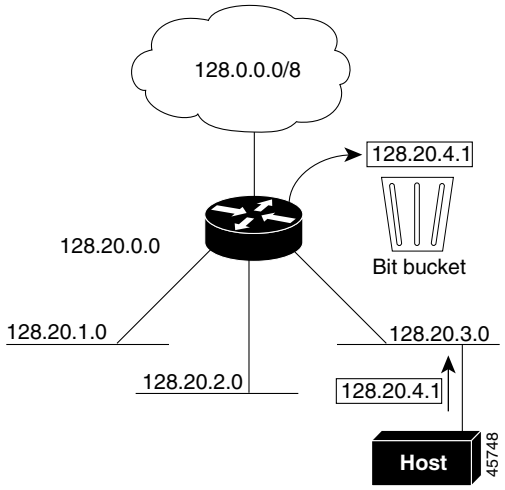
In Figure 31-2, classless routing is enabled. When the host sends a packet to 120.20.4.1, instead of discarding the packet, the router forwards it to the best supernet route. If you disable classless routing and a router receives packets destined for a subnet of a network with no network default route, the router discards the packet.

Figure 31-2 IP Classless Routing



In Figure 31-3, the router in network 128.20.0.0 is connected to subnets 128.20.1.0, 128.20.2.0, and 128.20.3.0. If the host sends a packet to 120.20.4.1, because there is no network default route, the router discards the packet.

Figure 31-3 No IP Classless Routing



To prevent the switch from forwarding packets destined for unrecognized subnets to the best supernet route possible, you can disable classless routing behavior.

Beginning in privileged EXEC mode, follow these steps to disable classless routing:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no ip classless**	Disable classless routing behavior.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entry.
Step 5	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

To restore the default and have the switch forward packets destined for a subnet of a network with no network default route to the best supernet route possible, use the **ip classless** global configuration command.

Configuring Address Resolution Methods

You can control interface-specific handling of IP by using address resolution. A device using IP can have both a local address or MAC address, which uniquely defines the device on its local segment or LAN, and a network address, which identifies the network to which the device belongs. The local address or MAC address is known as a data link address because it is contained in the data link layer (Layer 2) section of the packet header and is read by data link (Layer 2) devices. To communicate with a device on Ethernet, the software must determine the MAC address of the device. The process of determining the MAC address from an IP address is called *address resolution*. The process of determining the IP address from the MAC address is called *reverse address resolution*.

The switch can use these forms of address resolution:

- Address Resolution Protocol (ARP) is used to associate IP address with MAC addresses. Taking an IP address as input, ARP determines the associated MAC address and then stores the IP address/MAC address association in an ARP cache for rapid retrieval. Then the IP datagram is encapsulated in a link-layer frame and sent over the network. Encapsulation of IP datagrams and ARP requests or replies on IEEE 802 networks other than Ethernet is specified by the Subnetwork Access Protocol (SNAP).
- Proxy ARP helps hosts with no routing tables determine the MAC addresses of hosts on other networks or subnets. If the switch (router) receives an ARP request for a host that is not on the same interface as the ARP request sender, and if the router has all of its routes to the host through other interfaces, it generates a proxy ARP packet giving its own local data link address. The host that sent the ARP request then sends its packets to the router, which forwards them to the intended host.

The switch also uses the Reverse Address Resolution Protocol (RARP), which functions the same as ARP does, except that the RARP packets request an IP address instead of a local MAC address. Using RARP requires a RARP server on the same network segment as the router interface. Use the **ip rarp-server** *address* interface configuration command to identify the server.

For more information on RARP, see the *Cisco IOS Configuration Fundamentals Configuration Guide, Release 12.2*.

You can perform these tasks to configure address resolution:

- Define a Static ARP Cache, page 31-9
- Set ARP Encapsulation, page 31-10
- Enable Proxy ARP, page 31-10

Define a Static ARP Cache

ARP and other address resolution protocols provide dynamic mapping between IP addresses and MAC addresses. Because most hosts support dynamic address resolution, you usually do not need to specify static ARP cache entries. If you must define a static ARP cache entry, you can do so globally, which installs a permanent entry in the ARP cache that the switch uses to translate IP addresses into MAC addresses. Optionally, you can also specify that the switch respond to ARP requests as if it were the owner of the specified IP address. If you do not want the ARP entry to be permanent, you can specify a timeout period for the ARP entry.

Beginning in privileged EXEC mode, follow these steps to provide static mapping between IP addresses and MAC addresses:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**arp** *ip-address hardware-address type*	Globally associate an IP address with a MAC (hardware) address in the ARP cache, and specify encapsulation type as one of these: • **arpa**—ARP encapsulation for Ethernet interfaces • **snap**—Subnetwork Address Protocol encapsulation for Token Ring and FDDI interfaces • **sap**—HP's ARP type
Step 3	**arp** *ip-address hardware-address type* [**alias**]	(Optional) Specify that the switch respond to ARP requests as if it were the owner of the specified IP address.
Step 4	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
Step 5	**arp timeout** *seconds*	(Optional) Set the length of time an ARP cache entry will stay in the cache. The default is 14400 seconds (4 hours). The range is 0 to 2147483 seconds. **Note** We recommend that you do not set an ARP timeout value lower than 120 seconds.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show interfaces** [*interface-id*]	Verify the type of ARP and the timeout value used on all interfaces or a specific interface.
Step 8	**show arp** **show ip arp**	View the contents of the ARP cache.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an entry from the ARP cache, use the **no arp** *ip-address hardware-address type* global configuration command. To remove all nonstatic entries from the ARP cache, use the **clear arp-cache** privileged EXEC command.

Set ARP Encapsulation

By default, Ethernet ARP encapsulation (represented by the **arpa** keyword) is enabled on an IP interface. You can change the encapsulation methods to SNAP if required by your network.

Beginning in privileged EXEC mode, follow these steps to specify the ARP encapsulation type:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**arp** {**arpa** | **snap**}	Specify the ARP encapsulation method: • **arpa**—Address Resolution Protocol • **snap**—Subnetwork Address Protocol
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show interfaces** [*interface-id*]	Verify ARP encapsulation configuration on all interfaces or the specified interface.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable an encapsulation type, use the **no arp arpa** or **no arp snap** interface configuration command.

Enable Proxy ARP

By default, the switch uses proxy ARP to help hosts determine MAC addresses of hosts on other networks or subnets.

Beginning in privileged EXEC mode, follow these steps to enable proxy ARP if it has been disabled:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**ip proxy-arp**	Enable proxy ARP on the interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip interface** [*interface-id*]	Verify the configuration on the interface or all interfaces.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable proxy ARP on the interface, use the **no ip proxy-arp** interface configuration command.

Routing Assistance When IP Routing is Disabled

These mechanisms allow the switch to learn about routes to other networks when it does not have IP routing enabled:

- Proxy ARP, page 31-11
- Default Gateway, page 31-11
- ICMP Router Discovery Protocol (IRDP), page 31-12

Proxy ARP

Proxy ARP, the most common method for learning about other routes, enables an Ethernet host with no routing information to communicate with hosts on other networks or subnets. The host assumes that all hosts are on the same local Ethernet and that they can use ARP to determine their MAC addresses. If a switch receives an ARP request for a host that is not on the same network as the sender, the switch evaluates whether it has the best route to that host. If it does, it sends an ARP reply packet with its own Ethernet MAC address, and the host that sent the request sends the packet to the switch, which forwards it to the intended host. Proxy ARP treats all networks as if they are local and performs ARP requests for every IP address.

Proxy ARP is enabled by default. To enable it after it has been disabled, see the "Enable Proxy ARP" section on page 31-10. Proxy ARP works as long as other routers support it.

Default Gateway

Another method for locating routes is to define a default router or default gateway. All nonlocal packets are sent to this router, which either routes them appropriately or sends an IP Control Message Protocol (ICMP) redirect message back, defining which local router the host should use. The switch caches the redirect messages and forwards each packet as efficiently as possible. A limitation of this method is that there is no means of detecting when the default router has gone down or is unavailable.

Beginning in privileged EXEC mode, follow these steps to define a default gateway (router) when IP routing is disabled:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip default-gateway** *ip-address*	Set up a default gateway (router).
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip redirects**	Display the address of the default gateway router to verify the setting.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip default-gateway** global configuration command to disable this function.

This example shows how to set and verify a default gateway:

```
Switch(config)# ip default-gateway 10.1.5.59
Switch(config)# end
Switch# show ip redirect
Default gateway is 10.1.5.59
Host               Gateway           Last Use    Total Uses  Interface
ICMP redirect cache is empty
```

ICMP Router Discovery Protocol (IRDP)

Router discovery allows the switch to dynamically learn about routes to other networks using IRDP. IRDP allows hosts to locate routers. When operating as a client, the switch generates router discovery packets. When operating as a host, the switch receives router discovery packets. The switch can also listen to Routing Information Protocol (RIP) routing updates and use this information to infer locations of routers. The switch does not actually store the routing tables sent by routing devices; it merely keeps track of which systems are sending the data. The advantage of using IRDP is that it allows each router to specify both a priority and the time after which a device is assumed to be down if no further packets are received.

Each device discovered becomes a candidate for the default router, and a new highest-priority router is selected when a higher priority router is discovered, when the current default router is declared down, or when a TCP connection is about to time out because of excessive retransmissions.

The only required task for IRDP routing on an interface is to enable IRDP processing on that interface. When enabled, the default parameters apply. You can optionally change any of these parameters.

Beginning in privileged EXEC mode, follow these steps to enable and configure IRDP on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**ip irdp**	Enable IRDP processing on the interface.
Step 4	**ip irdp multicast**	(Optional) Send IRDP advertisements to the multicast address (224.0.0.1) instead of IP broadcasts. **Note** This command allows for compatibility with Sun Microsystems Solaris, which requires IRDP packets to be sent out as multicasts. Many implementations cannot receive these multicasts; ensure end-host ability before using this command.
Step 5	**ip irdp holdtime** *seconds*	(Optional) Set the IRDP period for which advertisements are valid. The default is three times the **maxadvertinterval** value. It must be greater than **maxadvertinterval** and cannot be greater than 9000 seconds. If you change the **maxadvertinterval** value, this value also changes.
Step 6	**ip irdp maxadvertinterval** *seconds*	(Optional) Set the IRDP maximum interval between advertisements. The default is 600 seconds.
Step 7	**ip irdp minadvertinterval** *seconds*	(Optional) Set the IRDP minimum interval between advertisements. The default is 0.75 times the **maxadvertinterval**. If you change the **maxadvertinterval**, this value changes to the new default (0.75 of **maxadvertinterval**).
Step 8	**ip irdp preference** *number*	(Optional) Set a device IRDP preference level. The allowed range is -2^{31} to 2^{31}. The default is 0. A higher value increases the router preference level.
Step 9	**ip irdp address** *address* [*number*]	(Optional) Specify an IRDP address and preference to proxy-advertise.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**show ip irdp**	Verify settings by displaying IRDP values.
Step 12	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

If you change the **maxadvertinterval** value, the **holdtime** and **minadvertinterval** values also change, so it is important to first change the **maxadvertinterval** value, before manually changing either the **holdtime** or **minadvertinterval** values.

Use the **no ip irdp** interface configuration command to disable IRDP routing.

Configuring Broadcast Packet Handling

After configuring an IP interface address, you can enable routing and configure one or more routing protocols, or you can configure the way the switch responds to network broadcasts. A broadcast is a data packet destined for all hosts on a physical network. The switch supports two kinds of broadcasting:

- A directed broadcast packet is sent to a specific network or series of networks. A directed broadcast address includes the network or subnet fields.
- A flooded broadcast packet is sent to every network.

Note You can also limit broadcast, unicast, and multicast traffic on Layer 2 interfaces by using the **storm-control** interface configuration commands. For more information, see Chapter 21, "Configuring Port-Based Traffic Control."

Routers provide some protection from broadcast storms by limiting their extent to the local cable. Bridges (including intelligent bridges), because they are Layer 2 devices, forward broadcasts to all network segments, thus propagating broadcast storms. The best solution to the broadcast storm problem is to use a single broadcast address scheme on a network. In most modern IP implementations, you can set the address to be used as the broadcast address. Many implementations, including the one in the switch, support several addressing schemes for forwarding broadcast messages.

Perform the tasks in these sections to enable these schemes:

- Enabling Directed Broadcast-to-Physical Broadcast Translation, page 31-13
- Forwarding UDP Broadcast Packets and Protocols, page 31-14
- Establishing an IP Broadcast Address, page 31-15
- Flooding IP Broadcasts, page 31-16

Enabling Directed Broadcast-to-Physical Broadcast Translation

By default, IP directed broadcasts are dropped; they are not forwarded. Dropping IP-directed broadcasts makes routers less susceptible to denial-of-service attacks.

You can enable forwarding of IP-directed broadcasts on an interface where the broadcast becomes a physical (MAC-layer) broadcast. Only those protocols configured by using the **ip forward-protocol** global configuration command are forwarded.

You can specify an access list to control which broadcasts are forwarded. When an access list is specified, only those IP packets permitted by the access list are eligible to be translated from directed broadcasts to physical broadcasts. For more information on access lists, see Chapter 29, "Configuring Network Security with ACLs."

Beginning in privileged EXEC mode, follow these steps to enable forwarding of IP-directed broadcasts on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
Step 3	**ip directed-broadcast** [*access-list-number*]	Enable directed broadcast-to-physical broadcast translation on the interface. You can include an access list to control which broadcasts are forwarded. When you specify an access list, only IP packets permitted by the access list can be translated. **Note** The **ip directed-broadcast** interface configuration command can be configured on a VPN routing/forwarding (VRF) interface and is VRF aware. Directed broadcast traffic is routed only within the VRF.
Step 4	**exit**	Return to global configuration mode.
Step 5	**ip forward-protocol** {**udp** [*port*] | **nd** | **sdns**}	Specify which protocols and ports the router forwards when forwarding broadcast packets. • **udp**—Forward UPD datagrams. *port*: (Optional) Destination port that controls which UDP services are forwarded. • **nd**—Forward ND datagrams. • **sdns**—Forward SDNS datagrams
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show ip interface** [*interface-id*] **show running-config**	Verify the configuration on the interface or all interfaces.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip directed-broadcast** interface configuration command to disable translation of directed broadcast to physical broadcasts. Use the **no ip forward-protocol** global configuration command to remove a protocol or port.

Forwarding UDP Broadcast Packets and Protocols

User Datagram Protocol (UDP) is an IP host-to-host layer protocol, as is TCP. UDP provides a low-overhead, connectionless session between two end systems and does not provide for acknowledgment of received datagrams. Network hosts occasionally use UDP broadcasts to determine address, configuration, and name information. If such a host is on a network segment that does not include a server, UDP broadcasts are normally not forwarded. You can remedy this situation by configuring an interface on a router to forward certain classes of broadcasts to a helper address. You can use more than one helper address per interface.

You can specify a UDP destination port to control which UDP services are forwarded. You can specify multiple UDP protocols. You can also specify the Network Disk (ND) protocol, which is used by older diskless Sun workstations and the network security protocol SDNS.

By default, both UDP and ND forwarding are enabled if a helper address has been defined for an interface. The description for the **ip forward-protocol** interface configuration command in the *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2* lists the ports that are forwarded by default if you do not specify any UDP ports.

If you do not specify any UDP ports when you configure the forwarding of UDP broadcasts, you are configuring the router to act as a BOOTP forwarding agent. BOOTP packets carry DHCP information.

Beginning in privileged EXEC mode, follow these steps to enable forwarding UDP broadcast packets on an interface and specify the destination address:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**ip helper-address** *address*	Enable forwarding and specify the destination address for forwarding UDP broadcast packets, including BOOTP.
Step 4	**exit**	Return to global configuration mode.
Step 5	**ip forward-protocol** {**udp** [*port*] | **nd** | **sdns**}	Specify which protocols the router forwards when forwarding broadcast packets.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show ip interface** [*interface-id*] **show running-config**	Verify the configuration on the interface or all interfaces.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip helper-address** interface configuration command to disable the forwarding of broadcast packets to specific addresses. Use the **no ip forward-protocol** global configuration command to remove a protocol or port.

Establishing an IP Broadcast Address

The most popular IP broadcast address (and the default) is an address consisting of all ones (255.255.255.255). However, the switch can be configured to generate any form of IP broadcast address.

Beginning in privileged EXEC mode, follow these steps to set the IP broadcast address on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
Step 3	**ip broadcast-address** *ip-address*	Enter a broadcast address different from the default, for example 128.1.255.255.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip interface** [*interface-id*]	Verify the broadcast address on the interface or all interfaces.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To restore the default IP broadcast address, use the **no ip broadcast-address** interface configuration command.

Flooding IP Broadcasts

You can allow IP broadcasts to be flooded throughout your internetwork in a controlled fashion by using the database created by the bridging STP. Using this feature also prevents loops. To support this capability, bridging must be configured on each interface that is to participate in the flooding. If bridging is not configured on an interface, it still can receive broadcasts. However, the interface never forwards broadcasts it receives, and the router never uses that interface to send broadcasts received on a different interface.

Packets that are forwarded to a single network address using the IP helper-address mechanism can be flooded. Only one copy of the packet is sent on each network segment.

To be considered for flooding, packets must meet these criteria. (Note that these are the same conditions used to consider packet forwarding using IP helper addresses.)

- The packet must be a MAC-level broadcast.
- The packet must be an IP-level broadcast.
- The packet must be a TFTP, DNS, Time, NetBIOS, ND, or BOOTP packet, or a UDP specified by the **ip forward-protocol udp** global configuration command.
- The time-to-live (TTL) value of the packet must be at least two.

A flooded UDP datagram is given the destination address specified with the **ip broadcast-address** interface configuration command on the output interface. The destination address can be set to any address. Thus, the destination address might change as the datagram propagates through the network. The source address is never changed. The TTL value is decremented.

When a flooded UDP datagram is sent out an interface (and the destination address possibly changed), the datagram is handed to the normal IP output routines and is, therefore, subject to access lists, if they are present on the output interface.

Beginning in privileged EXEC mode, follow these steps to use the bridging spanning-tree database to flood UDP datagrams:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip forward-protocol spanning-tree**	Use the bridging spanning-tree database to flood UDP datagrams.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entry.
Step 5	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

Use the **no ip forward-protocol spanning-tree** global configuration command to disable the flooding of IP broadcasts.

In the switch, the majority of packets are forwarded in hardware; most packets do not go through the switch CPU. For those packets that do go to the CPU, you can speed up spanning tree-based UDP flooding by a factor of about four to five times by using turbo-flooding. This feature is supported over Ethernet interfaces configured for ARP encapsulation.

Beginning in privileged EXEC mode, follow these steps to increase spanning-tree-based flooding:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode
Step 2	**ip forward-protocol turbo-flood**	Use the spanning-tree database to speed up flooding of UDP datagrams.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entry.
Step 5	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

To disable this feature, use the **no ip forward-protocol turbo-flood** global configuration command.

Monitoring and Maintaining IP Addressing

When the contents of a particular cache, table, or database have become or are suspected to be invalid, you can remove all its contents by using the **clear** privileged EXEC commands. Table 31-3 lists the commands for clearing contents.

Table 31-3 Commands to Clear Caches, Tables, and Databases

Command	Purpose
clear arp-cache	Clear the IP ARP cache and the fast-switching cache.
clear host {*name* | *}}	Remove one or all entries from the host name and the address cache.
clear ip route {*network* [*mask*] |*}	Remove one or more routes from the IP routing table.

You can display specific statistics, such as the contents of IP routing tables, caches, and databases; the reachability of nodes; and the routing path that packets are taking through the network. Table 31-4 lists the privileged EXEC commands for displaying IP statistics.

Table 31-4 Commands to Display Caches, Tables, and Databases

Command	Purpose
show arp	Display the entries in the ARP table.
show hosts	Display the default domain name, style of lookup service, name server hosts, and the cached list of host names and addresses.
show ip aliases	Display IP addresses mapped to TCP ports (aliases).
show ip arp	Display the IP ARP cache.
show ip interface [*interface-id*]	Display the IP status of interfaces.
show ip irdp	Display IRDP values.
show ip masks *address*	Display the masks used for network addresses and the number of subnets using each mask.
show ip redirects	Display the address of a default gateway.
show ip route [*address* [*mask*]] | [*protocol*]	Display the current state of the routing table.
show ip route summary	Display the current state of the routing table in summary form.

Enabling IP Unicast Routing

By default, the switch is in Layer 2 switching mode and IP routing is disabled. To use the Layer 3 capabilities of the switch, you must enable IP routing.

Beginning in privileged EXEC mode, follow these steps to enable IP routing:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip routing**	Enable IP routing.
Step 3	**router** *ip_routing_protocol*	Specify an IP routing protocol. This step might include other commands, such as specifying the networks to route with the **network** (RIP) router configuration command. For information on specific protocols, see sections later in this chapter and in the *Cisco IOS IP Configuration Guide, Release 12.2*. **Note** The IP base image supports only RIP as a routing protocol.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip routing** global configuration command to disable routing.

This example shows how to enable IP routing using RIP as the routing protocol:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip routing
Switch(config)# router rip
Switch(config-router)# network 10.0.0.0
Switch(config-router)# end
```

You can now set up parameters for the selected routing protocols as described in these sections:

- Configuring RIP, page 31-19
- Configuring OSPF, page 31-24
- Configuring EIGRP, page 31-34
- Configuring BGP, page 31-40

Configuring RIP

The Routing Information Protocol (RIP) is an interior gateway protocol (IGP) created for use in small, homogeneous networks. It is a distance-vector routing protocol that uses broadcast User Datagram Protocol (UDP) data packets to exchange routing information. The protocol is documented in RFC 1058. You can find detailed information about RIP in *IP Routing Fundamentals,* published by Cisco Press.

Note RIP is the only routing protocol supported by the IP base image; other routing protocols require the IP services image on the switch.

Using RIP, the switch sends routing information updates (advertisements) every 30 seconds. If a router does not receive an update from another router for 180 seconds or more, it marks the routes served by that router as unusable. If there is still no update after 240 seconds, the router removes all routing table entries for the non-updating router.

RIP uses hop counts to rate the value of different routes. The hop count is the number of routers that can be traversed in a route. A directly connected network has a hop count of zero; a network with a hop count of 16 is unreachable. This small range (0 to 15) makes RIP unsuitable for large networks.

If the router has a default network path, RIP advertises a route that links the router to the pseudonetwork 0.0.0.0. The 0.0.0.0 network does not exist; it is treated by RIP as a network to implement the default routing feature. The switch advertises the default network if a default was learned by RIP or if the router has a gateway of last resort and RIP is configured with a default metric. RIP sends updates to the interfaces in specified networks. If an interface's network is not specified, it is not advertised in any RIP update.

This section briefly describes how to configure RIP. It includes this information:

- Default RIP Configuration, page 31-19
- Configuring Basic RIP Parameters, page 31-20
- Configuring RIP Authentication, page 31-22
- Configuring Summary Addresses and Split Horizon, page 31-22

Default RIP Configuration

Table 31-5 shows the default RIP configuration.

Table 31-5 Default RIP Configuration

Feature	Default Setting
Auto summary	Enabled.
Default-information originate	Disabled.
Default metric	Built-in; automatic metric translations.
IP RIP authentication key-chain	No authentication. Authentication mode: clear text.
IP RIP receive version	According to the **version** router configuration command.
IP RIP send version	According to the **version** router configuration command.
IP RIP triggered	According to the **version** router configuration command.

Table 31-5 Default RIP Configuration (continued)

Feature	Default Setting
IP split horizon	Varies with media.
Neighbor	None defined.
Network	None specified.
Offset list	Disabled.
Output delay	0 milliseconds.
Timers basic	• Update: 30 seconds. • Invalid: 180 seconds. • Hold-down: 180 seconds. • Flush: 240 seconds.
Validate-update-source	Enabled.
Version	Receives RIP version 1 and 2 packets; sends version 1 packets.

Configuring Basic RIP Parameters

To configure RIP, you enable RIP routing for a network and optionally configure other parameters.

Beginning in privileged EXEC mode, follow these steps to enable and configure RIP:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip routing**	Enable IP routing (required only if IP routing is disabled).
Step 3	**router rip**	Enable a RIP routing process, and enter router configuration mode.
Step 4	**network** *network number*	Associate a network with a RIP routing process. You can specify multiple **network** commands. RIP routing updates are sent and received through interfaces only on these networks.
Step 5	**neighbor** *ip-address*	(Optional) Define a neighboring router with which to exchange routing information. This step allows routing updates from RIP (normally a broadcast protocol) to reach nonbroadcast networks.
Step 6	**offset list** [*access-list number* | *name*] {**in** | **out**} *offset* [*type number*]	(Optional) Apply an offset list to routing metrics to increase incoming and outgoing metrics to routes learned through RIP. You can limit the offset list with an access list or an interface.

	Command	Purpose
Step 7	**timers basic** *update invalid holddown flush*	(Optional) Adjust routing protocol timers. Valid ranges for all timers are 0 to 4294967295 seconds. • *update*—Time between sending routing updates. The default is 30 seconds. • *invalid*—Time after which a route is declared invalid. The default is 180 seconds. • *holddown*—Time before a route is removed from the routing table. The default is 180 seconds. • *flush*—Amount of time for which routing updates are postponed. The default is 240 seconds.
Step 8	**version** {**1** | **2**}	(Optional) Configure the switch to receive and send only RIP Version 1 or RIP version 2 packets. By default, the switch receives Version 1 and 2 but sends only Version 1. You can also use the interface commands **ip rip** {**send** | **receive**} **version 1** | **2** | **1 2**} to control what versions are used for sending and receiving on interfaces.
Step 9	**no auto summary**	(Optional) Disable automatic summarization. By default, the switch summarizes subprefixes when crossing classful network boundaries. Disable summarization (RIP version 2 only) to advertise subnet and host routing information to classful network boundaries.
Step 10	**no validate-update-source**	(Optional) Disable validation of the source IP address of incoming RIP routing updates. By default, the switch validates the source IP address of incoming RIP routing updates and discards the update if the source address is not valid. Under normal circumstances, disabling this feature is not recommended. However, if you have a router that is off-network and you want to receive its updates, you can use this command.
Step 11	**output-delay** *delay*	(Optional) Add interpacket delay for RIP updates sent. By default, packets in a multiple-packet RIP update have no delay added between packets. If you are sending packets to a lower-speed device, you can add an interpacket delay in the range of 8 to 50 milliseconds.
Step 12	**end**	Return to privileged EXEC mode.
Step 13	**show ip protocols**	Verify your entries.
Step 14	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To turn off the RIP routing process, use the **no router rip** global configuration command.

To display the parameters and current state of the active routing protocol process, use the **show ip protocols** privileged EXEC command. Use the **show ip rip database** privileged EXEC command to display summary address entries in the RIP database.

Configuring RIP Authentication

RIP version 1 does not support authentication. If you are sending and receiving RIP Version 2 packets, you can enable RIP authentication on an interface. The key chain determines the set of keys that can be used on the interface. If a key chain is not configured, no authentication is performed, not even the default. Therefore, you must also perform the tasks in the "Managing Authentication Keys" section on page 31-83.

The switch supports two modes of authentication on interfaces for which RIP authentication is enabled: plain text and MD5. The default is plain text.

Beginning in privileged EXEC mode, follow these steps to configure RIP authentication on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
Step 3	**ip rip authentication key-chain** *name-of-chain*	Enable RIP authentication.
Step 4	**ip rip authentication mode** [**text** | **md5**]	Configure the interface to use plain text authentication (the default) or MD5 digest authentication.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config interface** [*interface-id*]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To restore clear text authentication, use the **no ip rip authentication mode** interface configuration command. To prevent authentication, use the **no ip rip authentication key-chain** interface configuration command.

Configuring Summary Addresses and Split Horizon

Routers connected to broadcast-type IP networks and using distance-vector routing protocols normally use the split-horizon mechanism to reduce the possibility of routing loops. Split horizon blocks information about routes from being advertised by a router on any interface from which that information originated. This feature usually optimizes communication among multiple routers, especially when links are broken.

Note In general, disabling split horizon is not recommended unless you are certain that your application requires it to properly advertise routes.

If you want to configure an interface running RIP to advertise a summarized local IP address pool on a network access server for dial-up clients, use the **ip summary-address rip** interface configuration command.

Note If split horizon is enabled, neither autosummary nor interface IP summary addresses are advertised.

Beginning in privileged EXEC mode, follow these steps to set an interface to advertise a summarized local IP address and to disable split horizon on the interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**ip address** *ip-address subnet-mask*	Configure the IP address and IP subnet.
Step 4	**ip summary-address rip** *ip address ip-network mask*	Configure the IP address to be summarized and the IP network mask.
Step 5	**no ip split horizon**	Disable split horizon on the interface.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show ip interface** *interface-id*	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IP summarization, use the **no ip summary-address rip** router configuration command.

In this example, the major net is 10.0.0.0. The summary address 10.2.0.0 overrides the autosummary address of 10.0.0.0 so that 10.2.0.0 is advertised out interface Gigabit Ethernet 0.2, and 10.0.0.0 is not advertised. In the example, if the interface is still in Layer 2 mode (the default), you must enter a **no switchport** interface configuration command before entering the **ip address** interface configuration command.

Note If split horizon is enabled, neither autosummary nor interface summary addresses (those configured with the **ip summary-address rip** router configuration command) are advertised.

```
Switch(config)# router rip
Switch(config-router)# interface gi0/2
Switch(config-if)# ip address 10.1.5.1 255.255.255.0
Switch(config-if)# ip summary-address rip 10.2.0.0 255.255.0.0
Switch(config-if)# no ip split-horizon
Switch(config-if)# exit
Switch(config)# router rip
Switch(config-router)# network 10.0.0.0
Switch(config-router)# neighbor 2.2.2.2 peer-group mygroup
Switch(config-router)# end
```

Configuring OSPF

This section briefly describes how to configure Open Shortest Path First (OSPF). For a complete description of the OSPF commands, see the "OSPF Commands" chapter of the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*.

Note OSPF classifies different media into broadcast, nonbroadcast, and point-to-point networks. The switch supports broadcast (Ethernet, Token Ring, and FDDI) and point-to-point networks (Ethernet interfaces configured as point-to-point links).

OSPF is an Interior Gateway Protocol (IGP) designed expressly for IP networks, supporting IP subnetting and tagging of externally derived routing information. OSPF also allows packet authentication and uses IP multicast when sending and receiving packets. The Cisco implementation supports RFC 1253, OSPF management information base (MIB).

The Cisco implementation conforms to the OSPF Version 2 specifications with these key features:

- Definition of stub areas is supported.
- Routes learned through any IP routing protocol can be redistributed into another IP routing protocol. At the intradomain level, this means that OSPF can import routes learned through RIP. OSPF routes can also be exported into RIP.
- Plain text and MD5 authentication among neighboring routers within an area is supported.
- Configurable routing interface parameters include interface output cost, retransmission interval, interface transmit delay, router priority, router dead and hello intervals, and authentication key.
- Virtual links are supported.
- Not-so-stubby-areas (NSSAs) per RFC 1587 are supported.

OSPF typically requires coordination among many internal routers, *area border routers* (ABRs) connected to multiple areas, and *autonomous system boundary routers* (ASBRs). The minimum configuration would use all default parameter values, no authentication, and interfaces assigned to areas. If you customize your environment, you must ensure coordinated configuration of all routers.

This section briefly describes how to configure OSPF. It includes this information:

- Default OSPF Configuration, page 31-25
- Configuring Basic OSPF Parameters, page 31-26
- Configuring OSPF Interfaces, page 31-27
- Configuring OSPF Area Parameters, page 31-28
- Configuring Other OSPF Parameters, page 31-30
- Changing LSA Group Pacing, page 31-32
- Configuring a Loopback Interface, page 31-32
- Monitoring OSPF, page 31-33

Default OSPF Configuration

Table 31-6 shows the default OSPF configuration.

Table 31-6 Default OSPF Configuration

Feature	Default Setting
Interface parameters	Cost: No default cost predefined. Retransmit interval: 5 seconds. Transmit delay: 1 second. Priority: 1. Hello interval: 10 seconds. Dead interval: 4 times the hello interval. No authentication. No password specified. MD5 authentication disabled.
Area	Authentication type: 0 (no authentication). Default cost: 1. Range: Disabled. Stub: No stub area defined. NSSA: No NSSA area defined.
Auto cost	100 Mbps.
Default-information originate	Disabled. When enabled, the default metric setting is 10, and the external route type default is Type 2.
Default metric	Built-in, automatic metric translation, as appropriate for each routing protocol.
Distance OSPF	dist1 (all routes within an area): 110. dist2 (all routes from one area to another): 110. and dist3 (routes from other routing domains): 110.
OSPF database filter	Disabled. All outgoing link-state advertisements (LSAs) are flooded to the interface.
IP OSPF name lookup	Disabled.
Log adjacency changes	Enabled.
Neighbor	None specified.
Neighbor database filter	Disabled. All outgoing LSAs are flooded to the neighbor.
Network area	Disabled.
NSF[1] awareness	Enabled[2]. Allows Layer 3 switches to continue forwarding packets from a neighboring NSF-capable router during hardware or software changes.
Router ID	No OSPF routing process defined.
Summary address	Disabled.

Table 31-6 Default OSPF Configuration (continued)

Feature	Default Setting
Timers LSA group pacing	240 seconds.
Timers shortest path first (spf)	spf delay: 5 seconds. spf-holdtime: 10 seconds.
Virtual link	No area ID or router ID defined. Hello interval: 10 seconds. Retransmit interval: 5 seconds. Transmit delay: 1 second. Dead interval: 40 seconds. Authentication key: no key predefined. Message-digest key (MD5): no key predefined.

1. NSF = Nonstop forwarding
2. OSPF NSF awareness is enabled on Catalyst 3550, 3560 and 3750 switches running the IP services image, Cisco IOS Release 12.2(25)SEC or later.

Nonstop Forwarding Awareness

The OSPF NSF awareness feature is supported in the IP services image, beginning with Cisco IOS Release 12.2(25)SEC. When the neighboring router is NSF-capable, the Layer 3 switch continues to forward packets from the neighboring router during the interval between the primary Route Processor (RP) in a router failing and the backup RP taking over, or while the primary RP is manually reloaded for a nondisruptive software upgrade.

This feature cannot be disabled. For more information on this feature, see the *OSPF Nonstop Forwarding (NSF) Awareness Feature Guide* at this URL: http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a0080153edd.html

Configuring Basic OSPF Parameters

Enabling OSPF requires that you create an OSPF routing process, specify the range of IP addresses to be associated with the routing process, and assign area IDs to be associated with that range.

Beginning in privileged EXEC mode, follow these steps to enable OSPF:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip routing**	Enable IP routing (required only if IP routing is disabled).
Step 3	**router ospf** *process-id*	Enable OSPF routing, and enter router configuration mode. The process ID is an internally used identification parameter that is locally assigned and can be any positive integer. Each OSPF routing process has a unique value.

	Command	Purpose
Step 4	**network** *address wildcard-mask* **area** *area-id*	Define an interface on which OSPF runs and the area ID for that interface. You can use the wildcard-mask to use a single command to define one or more multiple interfaces to be associated with a specific OSPF area. The area ID can be a decimal value or an IP address.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip protocols**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To terminate an OSPF routing process, use the **no router ospf** *process-id* global configuration command.

This example shows how to configure an OSPF routing process and assign it a process number of 109:

```
Switch(config)# router ospf 109
Switch(config-router)# network 131.108.0.0 255.255.255.0 area 24
Switch(config-router)# end
```

Configuring OSPF Interfaces

You can use the **ip ospf** interface configuration commands to modify interface-specific OSPF parameters. You are not required to modify any of these parameters, but some interface parameters (hello interval, dead interval, and authentication key) must be consistent across all routers in an attached network. If you modify these parameters, be sure all routers in the network have compatible values.

Note The **ip ospf** interface configuration commands are all optional.

Beginning in privileged EXEC mode, follow these steps to modify OSPF interface parameters:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**ip ospf** *cost*	(Optional) Explicitly specify the cost of sending a packet on the interface.
Step 4	**ip ospf retransmit-interval** *seconds*	(Optional) Specify the number of seconds between link state advertisement transmissions. The range is 1 to 65535 seconds. The default is 5 seconds.
Step 5	**ip ospf transmit-delay** *seconds*	(Optional) Set the estimated number of seconds to wait before sending a link state update packet. The range is 1 to 65535 seconds. The default is 1 second.
Step 6	**ip ospf priority** *number*	(Optional) Set priority to help determine the OSPF designated router for a network. The range is from 0 to 255. The default is 1.

	Command	Purpose
Step 7	**ip ospf hello-interval** *seconds*	(Optional) Set the number of seconds between hello packets sent on an OSPF interface. The value must be the same for all nodes on a network. The range is 1 to 65535 seconds. The default is 10 seconds.
Step 8	**ip ospf dead-interval** *seconds*	(Optional) Set the number of seconds after the last device hello packet was seen before its neighbors declare the OSPF router to be down. The value must be the same for all nodes on a network. The range is 1 to 65535 seconds. The default is 4 times the hello interval.
Step 9	**ip ospf authentication-key** *key*	(Optional) Assign a password to be used by neighboring OSPF routers. The password can be any string of keyboard-entered characters up to 8 bytes in length. All neighboring routers on the same network must have the same password to exchange OSPF information.
Step 10	**ip ospf message digest-key** *keyid* **md5** *key*	(Optional) Enable MDS authentication. • *keyid*—An identifier from 1 to 255. • *key*—An alphanumeric password of up to 16 bytes.
Step 11	**ip ospf database-filter all out**	(Optional) Block flooding of OSPF LSA packets to the interface. By default, OSPF floods new LSAs over all interfaces in the same area, except the interface on which the LSA arrives.
Step 12	**end**	Return to privileged EXEC mode.
Step 13	**show ip ospf interface** [*interface-name*]	Display OSPF-related interface information.
Step 14	**show ip ospf neighbor detail**	Display NSF awareness status of neighbor switch. The output will match one of the following two examples: • *Options is 0x52* *LLS Options is 0x1 (LR)* When both of these lines appear, the neighbor switch is NSF aware. • *Options is 0x42*—This means the neighbor switch is not NSF aware.
Step 15	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of these commands to remove the configured parameter value or return to the default value.

Configuring OSPF Area Parameters

You can optionally configure several OSPF area parameters. These parameters include authentication for password-based protection against unauthorized access to an area, stub areas, and not-so-stubby-areas (NSSAs). *Stub areas* are areas into which information on external routes is not sent. Instead, the area border router (ABR) generates a default external route into the stub area for destinations outside the autonomous system (AS). An NSSA does not flood all LSAs from the core into the area, but can import AS external routes within the area by redistribution.

Route summarization is the consolidation of advertised addresses into a single summary route to be advertised by other areas. If network numbers are contiguous, you can use the **area range** router configuration command to configure the ABR to advertise a summary route that covers all networks in the range.

Note The OSPF **area** router configuration commands are all optional.

Beginning in privileged EXEC mode, follow these steps to configure area parameters:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router ospf** *process-id*	Enable OSPF routing, and enter router configuration mode.
Step 3	**area** *area-id* **authentication**	(Optional) Allow password-based protection against unauthorized access to the identified area. The identifier can be either a decimal value or an IP address.
Step 4	**area** *area-id* **authentication message-digest**	(Optional) Enable MD5 authentication on the area.
Step 5	**area** *area-id* **stub** [**no-summary**]	(Optional) Define an area as a stub area. The **no-summary** keyword prevents an ABR from sending summary link advertisements into the stub area.
Step 6	**area** *area-id* **nssa** [**no-redistribution**] [**default-information-originate**] [**no-summary**]	(Optional) Defines an area as a not-so-stubby-area. Every router within the same area must agree that the area is NSSA. Select one of these keywords: • **no-redistribution**—Select when the router is an NSSA ABR and you want the **redistribute** command to import routes into normal areas, but not into the NSSA. • **default-information-originate**—Select on an ABR to allow importing type 7 LSAs into the NSSA. • **no-redistribution**—Select to not send summary LSAs into the NSSA.
Step 7	**area** *area-id* **range** *address mask*	(Optional) Specify an address range for which a single route is advertised. Use this command only with area border routers.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show ip ospf** [*process-id*]	Display information about the OSPF routing process in general or for a specific process ID to verify configuration.
	show ip ospf [*process-id* [*area-id*]] **database**	Display lists of information related to the OSPF database for a specific router.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of these commands to remove the configured parameter value or to return to the default value.

Configuring Other OSPF Parameters

You can optionally configure other OSPF parameters in router configuration mode.

- Route summarization: When redistributing routes from other protocols as described in the "Using Route Maps to Redistribute Routing Information" section on page 31-76, each route is advertised individually in an external LSA. To help decrease the size of the OSPF link state database, you can use the **summary-address** router configuration command to advertise a single router for all the redistributed routes included in a specified network address and mask.
- Virtual links: In OSPF, all areas must be connected to a backbone area. You can establish a virtual link in case of a backbone-continuity break by configuring two Area Border Routers as endpoints of a virtual link. Configuration information includes the identity of the other virtual endpoint (the other ABR) and the nonbackbone link that the two routers have in common (the transit area). Virtual links cannot be configured through a stub area.
- Default route: When you specifically configure redistribution of routes into an OSPF routing domain, the route automatically becomes an autonomous system boundary router (ASBR). You can force the ASBR to generate a default route into the OSPF routing domain.
- Domain Name Server (DNS) names for use in all OSPF **show** privileged EXEC command displays makes it easier to identify a router than displaying it by router ID or neighbor ID.
- Default Metrics: OSPF calculates the OSPF metric for an interface according to the bandwidth of the interface. The metric is calculated as *ref-bw* divided by bandwidth, where *ref* is 10 by default, and bandwidth (*bw*) is determined by the **bandwidth** interface configuration command. For multiple links with high bandwidth, you can specify a larger number to differentiate the cost on those links.
- Administrative distance is a rating of the trustworthiness of a routing information source, an integer between 0 and 255, with a higher value meaning a lower trust rating. An administrative distance of 255 means the routing information source cannot be trusted at all and should be ignored. OSPF uses three different administrative distances: routes within an area (interarea), routes to another area (interarea), and routes from another routing domain learned through redistribution (external). You can change any of the distance values.
- Passive interfaces: Because interfaces between two devices on an Ethernet represent only one network segment, to prevent OSPF from sending hello packets for the sending interface, you must configure the sending device to be a passive interface. Both devices can identify each other through the hello packet for the receiving interface.
- Route calculation timers: You can configure the delay time between when OSPF receives a topology change and when it starts the shortest path first (SPF) calculation and the hold time between two SPF calculations.
- Log neighbor changes: You can configure the router to send a syslog message when an OSPF neighbor state changes, providing a high-level view of changes in the router.

Beginning in privileged EXEC mode, follow these steps to configure these OSPF parameters:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router ospf** *process-id*	Enable OSPF routing, and enter router configuration mode.
Step 3	**summary-address** *address mask*	(Optional) Specify an address and IP subnet mask for redistributed routes so that only one summary route is advertised.

	Command	Purpose
Step 4	**area** *area-id* **virtual-link** router-id [**hello-interval** *seconds*] [**retransmit-interval** *seconds*] [**trans**] [[**authentication-key** *key*] | **message-digest-key** *keyid* **md5** *key*]]	(Optional) Establish a virtual link and set its parameters. See the "Configuring OSPF Interfaces" section on page 31-27 for parameter definitions and Table 31-6 on page 31-25 for virtual link defaults.
Step 5	**default-information originate** [**always**] [**metric** *metric-value*] [**metric-type** *type-value*] [**route-map** *map-name*]	(Optional) Force the ASBR to generate a default route into the OSPF routing domain. Parameters are all optional.
Step 6	**ip ospf name-lookup**	(Optional) Configure DNS name lookup. The default is disabled.
Step 7	**ip auto-cost reference-bandwidth** *ref-bw*	(Optional) Specify an address range for which a single route will be advertised. Use this command only with area border routers.
Step 8	**distance ospf** {[**inter-area** *dist1*] [**inter-area** *dist2*] [**external** *dist3*]}	(Optional) Change the OSPF distance values. The default distance for each type of route is 110. The range is 1 to 255.
Step 9	**passive-interface** *type number*	(Optional) Suppress the sending of hello packets through the specified interface.
Step 10	**timers spf** *spf-delay spf-holdtime*	(Optional) Configure route calculation timers. • *spf-delay*—Enter an integer from 0 to 65535. The default is 5 seconds; 0 means no delay. • *spf-holdtime*—Enter an integer from 0 to 65535. The default is 10 seconds; 0 means no delay.
Step 11	**ospf log-adj-changes**	(Optional) Send syslog message when a neighbor state changes.
Step 12	**end**	Return to privileged EXEC mode.
Step 13	**show ip ospf** [*process-id* [*area-id*]] **database**	Display lists of information related to the OSPF database for a specific router. For some of the keyword options, see to the "Monitoring OSPF" section on page 31-33.
Step 14	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Changing LSA Group Pacing

The OSPF LSA group pacing feature allows the router to group OSPF LSAs and pace the refreshing, check-summing, and aging functions for more efficient router use. This feature is enabled by default with a 4-minute default pacing interval, and you will not usually need to modify this parameter. The optimum group pacing interval is inversely proportional to the number of LSAs the router is refreshing, check-summing, and aging. For example, if you have approximately 10,000 LSAs in the database, decreasing the pacing interval would benefit you. If you have a very small database (40 to 100 LSAs), increasing the pacing interval to 10 to 20 minutes might benefit you slightly.

Beginning in privileged EXEC mode, follow these steps to configure OSPF LSA pacing:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router ospf** *process-id*	Enable OSPF routing, and enter router configuration mode.
Step 3	**timers lsa-group-pacing** *seconds*	Change the group pacing of LSAs.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default value, use the **no timers lsa-group-pacing** router configuration command.

Configuring a Loopback Interface

OSPF uses the highest IP address configured on the interfaces as its router ID. If this interface is down or removed, the OSPF process must recalculate a new router ID and resend all its routing information out its interfaces. If a loopback interface is configured with an IP address, OSPF uses this IP address as its router ID, even if other interfaces have higher IP addresses. Because loopback interfaces never fail, this provides greater stability. OSPF automatically prefers a loopback interface over other interfaces, and it chooses the highest IP address among all loopback interfaces.

Beginning in privileged EXEC mode, follow these steps to configure a loopback interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface loopback 0**	Create a loopback interface, and enter interface configuration mode.
Step 3	**ip address** *address mask*	Assign an IP address to this interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip interface**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no interface loopback 0** global configuration command to disable the loopback interface.

Monitoring OSPF

You can display specific statistics such as the contents of IP routing tables, caches, and databases.

Table 31-7 lists some of the privileged EXEC commands for displaying statistics. For more **show ip ospf database** privileged EXEC command options and for explanations of fields in the resulting display, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*.

Table 31-7 Show IP OSPF Statistics Commands

Command	Purpose
show ip ospf [*process-id*]	Display general information about OSPF routing processes.
show ip ospf [*process-id*] **database** [**router**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**router**] [**self-originate**] **show ip ospf** [*process-id*] **database** [**router**] [**adv-router** [*ip-address*]] **show ip ospf** [*process-id*] **database** [**network**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**summary**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**asbr-summary**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**external**] [*link-state-id*] **show ip ospf** [*process-id area-id*] **database** [**database-summary**]	Display lists of information related to the OSPF database.
show ip ospf border-routes	Display the internal OSPF routing ABR and ASBR table entries.
show ip ospf interface [*interface-name*]	Display OSPF-related interface information.
show ip ospf neighbor [*interface-name*] [*neighbor-id*] **detail**	Display OSPF interface neighbor information.
show ip ospf virtual-links	Display OSPF-related virtual links information.

Configuring EIGRP

Enhanced IGRP (EIGRP) is a Cisco proprietary enhanced version of the IGRP. EIGRP uses the same distance vector algorithm and distance information as IGRP; however, the convergence properties and the operating efficiency of EIGRP are significantly improved.

Note The IP base image contains only EIGRP stub routing capability, which only advertises connected or summary routes from the routing tables to other switches in the network. The switch uses EIGRP stub routing at the access layer to eliminate the need for other types of routing advertisements. For enhanced capability and complete EIGRP routing, the switch must be running the IP services image.
On a switch running the IP base image, if you try to configure multi-VRF-CE and EIGRP stub routing at the same time, the configuration is not allowed.

The convergence technology employs an algorithm referred to as the Diffusing Update Algorithm (DUAL), which guarantees loop-free operation at every instant throughout a route computation and allows all devices involved in a topology change to synchronize at the same time. Routers that are not affected by topology changes are not involved in recomputations.

IP EIGRP provides increased network width. With RIP, the largest possible width of your network is 15 hops. Because the EIGRP metric is large enough to support thousands of hops, the only barrier to expanding the network is the transport-layer hop counter. EIGRP increments the transport control field only when an IP packet has traversed 15 routers and the next hop to the destination was learned through EIGRP. When a RIP route is used as the next hop to the destination, the transport control field is incremented as usual.

EIGRP offers these features:

- Fast convergence.
- Incremental updates when the state of a destination changes, instead of sending the entire contents of the routing table, minimizing the bandwidth required for EIGRP packets.
- Protocol-independent neighbor discovery mechanism to learn about neighboring routers.
- Variable-length subnet masks (VLSMs).
- Arbitrary route summarization.
- EIGRP scales to large networks.

EIGRP has these four basic components:

- *Neighbor discovery and recovery* is the process that routers use to dynamically learn of other routers on their directly attached networks. Routers must also discover when their neighbors become unreachable or inoperative. Neighbor discovery and recovery is achieved with low overhead by periodically sending small hello packets. As long as hello packets are received, the Cisco IOS software can determine that a neighbor is alive and functioning. When this status is determined, the neighboring routers can exchange routing information.
- *The reliable transport protocol* is responsible for guaranteed, ordered delivery of EIGRP packets to all neighbors. It supports intermixed transmission of multicast and unicast packets. Some EIGRP packets must be sent reliably, and others need not be. For efficiency, reliability is provided only when necessary. For example, on a multiaccess network that has multicast capabilities (such as Ethernet), it is not necessary to send hellos reliably to all neighbors individually. Therefore, EIGRP sends a single multicast hello with an indication in the packet informing the receivers that the packet need not be acknowledged. Other types of packets (such as updates) require acknowledgment, which

is shown in the packet. The reliable transport has a provision to send multicast packets quickly when there are unacknowledged packets pending. Doing so helps ensure that convergence time remains low in the presence of varying speed links.

- *The DUAL finite state machine* embodies the decision process for all route computations. It tracks all routes advertised by all neighbors. DUAL uses the distance information (known as a metric) to select efficient, loop-free paths. DUAL selects routes to be inserted into a routing table based on feasible successors. A successor is a neighboring router used for packet forwarding that has a least-cost path to a destination that is guaranteed not to be part of a routing loop. When there are no feasible successors, but there are neighbors advertising the destination, a recomputation must occur. This is the process whereby a new successor is determined. The amount of time it takes to recompute the route affects the convergence time. Recomputation is processor-intensive; it is advantageous to avoid recomputation if it is not necessary. When a topology change occurs, DUAL tests for feasible successors. If there are feasible successors, it uses any it finds to avoid unnecessary recomputation.
- The *protocol-dependent modules* are responsible for network layer protocol-specific tasks. An example is the IP EIGRP module, which is responsible for sending and receiving EIGRP packets that are encapsulated in IP. It is also responsible for parsing EIGRP packets and informing DUAL of the new information received. EIGRP asks DUAL to make routing decisions, but the results are stored in the IP routing table. EIGRP is also responsible for redistributing routes learned by other IP routing protocols.

This section briefly describes how to configure EIGRP. It includes this information:

- Default EIGRP Configuration, page 31-35
- Configuring Basic EIGRP Parameters, page 31-37
- Configuring EIGRP Interfaces, page 31-38
- Configuring EIGRP Route Authentication, page 31-39
- Monitoring and Maintaining EIGRP, page 31-40

Default EIGRP Configuration

Table 31-8 shows the default EIGRP configuration.

Table 31-8 Default EIGRP Configuration

Feature	Default Setting
Auto summary	Enabled. Subprefixes are summarized to the classful network boundary when crossing classful network boundaries.
Default-information	Exterior routes are accepted and default information is passed between EIGRP processes when doing redistribution.

Table 31-8 Default EIGRP Configuration (continued)

Feature	Default Setting
Default metric	Only connected routes and interface static routes can be redistributed without a default metric. The metric includes: • Bandwidth: 0 or greater kbps. • Delay (tens of microseconds): 0 or any positive number that is a multiple of 39.1 nanoseconds. • Reliability: any number between 0 and 255 (255 means 100 percent reliability). • Loading: effective bandwidth as a number between 0 and 255 (255 is 100 percent loading). • MTU: maximum transmission unit size of the route in bytes. 0 or any positive integer.
Distance	Internal distance: 90. External distance: 170.
EIGRP log-neighbor changes	Disabled. No adjacency changes logged.
IP authentication key-chain	No authentication provided.
IP authentication mode	No authentication provided.
IP bandwidth-percent	50 percent.
IP hello interval	For low-speed nonbroadcast multiaccess (NBMA) networks: 60 seconds; all other networks: 5 seconds.
IP hold-time	For low-speed NBMA networks: 180 seconds; all other networks: 15 seconds.
IP split-horizon	Enabled.
IP summary address	No summary aggregate addresses are predefined.
Metric weights	tos: 0; k1 and k3: 1; k2, k4, and k5: 0
Network	None specified.
NSF[1] Awareness	Enabled[2]. Allows Layer 3 switches to continue forwarding packets from a neighboring NSF-capable router during hardware or software changes.
Offset-list	Disabled.
Router EIGRP	Disabled.
Set metric	No metric set in the route map.
Traffic-share	Distributed proportionately to the ratios of the metrics.
Variance	1 (equal-cost load balancing).

1. NSF = Nonstop Forwarding
2. EIGRP NSF awareness is enabled on Catalyst 3550, 3560 and 3750 switches running the IP services image, Cisco IOS Release 12.2(25)SEC or later.

To create an EIGRP routing process, you must enable EIGRP and associate networks. EIGRP sends updates to the interfaces in the specified networks. If you do not specify an interface network, it is not advertised in any EIGRP update.

Note If you have routers on your network that are configured for IGRP, and you want to change to EIGRP, you must designate transition routers that have both IGRP and EIGRP configured. In these cases, perform Steps 1 through 3 in the next section. You must use the same AS number for routes to be automatically redistributed.

Nonstop Forwarding Awareness

The EIGRP NSF Awareness feature is supported in the IP services image, beginning with Cisco IOS Release 12.2(25)SEC. When the neighboring router is NSF-capable, the Layer 3 switch continues to forward packets from the neighboring router during the interval between the primary Route Processor (RP) in a router crashing and the backup RP taking over, or while the primary RP is manually reloaded for a non-disruptive software upgrade.

This feature cannot be disabled. For more information on this feature see the *EIGRP Nonstop Forwarding (NSF) Awareness Feature Guide* at this URL: http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a0080160010.html

Configuring Basic EIGRP Parameters

Beginning in privileged EXEC mode, follow these steps to configure EIGRP. Configuring the routing process is required; other steps are optional:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip routing**	Enable IP routing (required only if IP routing is disabled).
Step 3	**router eigrp** *autonomous-system*	Enable an EIGRP routing process, and enter router configuration mode. The AS number identifies the routes to other EIGRP routers and is used to tag routing information.
Step 4	**network** *network-number*	Associate networks with an EIGRP routing process. EIGRP sends updates to the interfaces in the specified networks. If an interface's network is not specified, it is not advertised in any EIGRP update.
Step 5	**eigrp log-neighbor-changes**	(Optional) Enable logging of EIGRP neighbor changes to monitor routing system stability.
Step 6	**metric weights** *tos k1 k2 k3 k4 k5*	(Optional) Adjust the EIGRP metric. Although the defaults have been carefully determined to provide excellent operation in most networks, you can adjust them. **Caution** Determining metrics is complex and is not recommended without guidance from an experienced network designer.

	Command	Purpose
Step 7	**offset list** [*access-list number* | *name*] {**in** | **out**} *offset* [*type number*]	(Optional) Apply an offset list to routing metrics to increase incoming and outgoing metrics to routes learned through EIGRP. You can limit the offset list with an access list or an interface.
Step 8	**no auto-summary**	(Optional) Disable automatic summarization of subnet routes into network-level routes.
Step 9	**ip summary-address eigrp** *autonomous-system-number address mask*	(Optional) Configure a summary aggregate.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**show ip protocols**	Verify your entries. For NSF awareness, the output shows: **** IP Routing is NSF aware **** *EIGRP NSF enabled*
Step 12	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** forms of these commands to disable the feature or return the setting to the default value.

Configuring EIGRP Interfaces

Other optional EIGRP parameters can be configured on an interface basis.

Beginning in privileged EXEC mode, follow these steps to configure EIGRP interfaces:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**ip bandwidth-percent eigrp** *percent*	(Optional) Configure the percentage of bandwidth that can be used by EIGRP on an interface. The default is 50 percent.
Step 4	**ip summary-address eigrp** *autonomous-system-number address mask*	(Optional) Configure a summary aggregate address for a specified interface (not usually necessary if auto-summary is enabled).
Step 5	**ip hello-interval eigrp** *autonomous-system-number seconds*	(Optional) Change the hello time interval for an EIGRP routing process. The range is 1 to 65535 seconds. The default is 60 seconds for low-speed NBMA networks and 5 seconds for all other networks.
Step 6	**ip hold-time eigrp** *autonomous-system-number seconds*	(Optional) Change the hold time interval for an EIGRP routing process. The range is 1 to 65535 seconds. The default is 180 seconds for low-speed NBMA networks and 15 seconds for all other networks. **Caution** Do not adjust the hold time without consulting Cisco technical support.

	Command	Purpose
Step 7	**no ip split-horizon eigrp** *autonomous-system-number*	(Optional) Disable split horizon to allow route information to be advertised by a router out any interface from which that information originated.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show ip eigrp interface**	Display which interfaces EIGRP is active on and information about EIGRP relating to those interfaces.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** forms of these commands to disable the feature or return the setting to the default value.

Configuring EIGRP Route Authentication

EIGRP route authentication provides MD5 authentication of routing updates from the EIGRP routing protocol to prevent the introduction of unauthorized or false routing messages from unapproved sources.

Beginning in privileged EXEC mode, follow these steps to enable authentication:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**ip authentication mode eigrp** *autonomous-system* **md5**	Enable MD5 authentication in IP EIGRP packets.
Step 4	**ip authentication key-chain eigrp** *autonomous-system key-chain*	Enable authentication of IP EIGRP packets.
Step 5	**exit**	Return to global configuration mode.
Step 6	**key chain** *name-of-chain*	Identify a key chain and enter key-chain configuration mode. Match the name configured in Step 4.
Step 7	**key** *number*	In key-chain configuration mode, identify the key number.
Step 8	**key-string** *text*	In key-chain key configuration mode, identify the key string.
Step 9	**accept-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*}	(Optional) Specify the time period during which the key can be received. The *start-time* and *end-time* syntax can be either *hh*:*mm*:*ss Month date year* or *hh*:*mm*:*ss date Month year*. The default is forever with the default *start-time* and the earliest acceptable date as January 1, 1993. The default *end-time* and **duration** is **infinite**.

	Command	Purpose
Step 10	**send-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*}	(Optional) Specify the time period during which the key can be sent. The *start-time* and *end-time* syntax can be either *hh*:*mm*:*ss Month date year* or *hh*:*mm*:*ss date Month year*. The default is forever with the default *start-time* and the earliest acceptable date as January 1, 1993. The default *end-time* and **duration** is **infinite**.
Step 11	**end**	Return to privileged EXEC mode.
Step 12	**show key chain**	Display authentication key information.
Step 13	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** forms of these commands to disable the feature or to return the setting to the default value.

Monitoring and Maintaining EIGRP

You can delete neighbors from the neighbor table. You can also display various EIGRP routing statistics. Table 31-9 lists the privileged EXEC commands for deleting neighbors and displaying statistics. For explanations of fields in the resulting display, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*.

Table 31-9 IP EIGRP Clear and Show Commands

Command	Purpose
clear ip eigrp neighbors [*if-address* | *interface*]	Delete neighbors from the neighbor table.
show ip eigrp interface [*interface*] [*as number*]	Display information about interfaces configured for EIGRP.
show ip eigrp neighbors [*type-number*]	Display EIGRP discovered neighbors.
show ip eigrp topology [*autonomous-system-number*] | [[*ip-address*] *mask*]]	Display the EIGRP topology table for a given process.
show ip eigrp traffic [*autonomous-system-number*]	Display the number of packets sent and received for all or a specified EIGRP process.

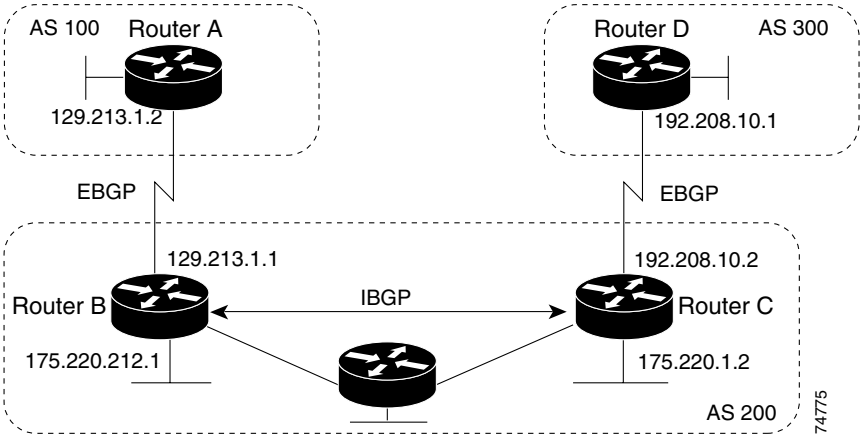
Configuring BGP

The Border Gateway Protocol (BGP) is an exterior gateway protocol used to set up an interdomain routing system that guarantees the loop-free exchange of routing information between autonomous systems. Autonomous systems are made up of routers that operate under the same administration and that run Interior Gateway Protocols (IGPs), such as RIP or OSPF, within their boundaries and that interconnect by using an Exterior Gateway Protocol (EGP). BGP version 4 is the standard EGP for interdomain routing in the Internet. The protocol is defined in RFCs 1163, 1267, and 1771. You can find detailed information about BGP in *Internet Routing Architectures,* published by Cisco Press, and in the "Configuring BGP" chapter in the "IP Routing Protocols" part of the *Cisco IOS IP Configuration Guide, Release 12.2*.

For details about BGP commands and keywords, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*. For a list of BGP commands not supported by the switch, see Appendix C, "Unsupported CLI Commands in Cisco IOS Release 12.2(25)SEE."

Routers that belong to the same autonomous system (AS) and that exchange BGP updates run *internal BGP* (IBGP), and routers that belong to different autonomous systems and that exchange BGP updates run *external BGP* (EBGP). Most configuration commands are the same for configuring EBGP and IBGP. The difference is that the routing updates are exchanged either between autonomous systems (EBGP) or within an AS (IBGP). Figure 31-4 shows a network that is running both EBGP and IBGP.

Figure 31-4 EBGP, IBGP, and Multiple Autonomous Systems



Before exchanging information with an external AS, BGP ensures that networks within the AS can be reached by defining internal BGP peering among routers within the AS and by redistributing BGP routing information to IGPs that run within the AS, such as IGRP and OSPF.

Routers that run a BGP routing process are often referred to as BGP *speakers*. BGP uses the Transmission Control Protocol (TCP) as its transport protocol (specifically port 179). Two BGP speakers that have a TCP connection to each other for exchanging routing information are known as *peers* or *neighbors*. In Figure 31-4, Routers A and B are BGP peers, as are Routers B and C and Routers C and D. The routing information is a series of AS numbers that describe the full path to the destination network. BGP uses this information to construct a loop-free map of autonomous systems.

The network has these characteristics:

- Routers A and B are running EBGP, and Routers B and C are running IBGP. Note that the EBGP peers are directly connected and that the IBGP peers are not. As long as there is an IGP running that allows the two neighbors to reach one another, IBGP peers do not have to be directly connected.
- All BGP speakers within an AS must establish a peer relationship with each other. That is, the BGP speakers within an AS must be fully meshed logically. BGP4 provides two techniques that reduce the requirement for a logical full mesh: *confederations* and *route reflectors*.
- AS 200 is a *transit AS* for AS 100 and AS 300—that is, AS 200 is used to transfer packets between AS 100 and AS 300.

BGP peers initially exchange their full BGP routing tables and then send only incremental updates. BGP peers also exchange keepalive messages (to ensure that the connection is up) and notification messages (in response to errors or special conditions).

In BGP, each route consists of a network number, a list of autonomous systems that information has passed through (the *autonomous system path*), and a list of other *path attributes*. The primary function of a BGP system is to exchange network reachability information, including information about the list of AS paths, with other BGP systems. This information can be used to determine AS connectivity, to prune routing loops, and to enforce AS-level policy decisions.

A router or switch running Cisco IOS does not select or use an IBGP route unless it has a route available to the next-hop router and it has received synchronization from an IGP (unless IGP synchronization is disabled). When multiple routes are available, BGP bases its path selection on *attribute* values. See the "Configuring BGP Decision Attributes" section on page 31-49 for information about BGP attributes.

BGP Version 4 supports classless interdomain routing (CIDR) so you can reduce the size of your routing tables by creating aggregate routes, resulting in *supernets*. CIDR eliminates the concept of network classes within BGP and supports the advertising of IP prefixes.

These sections briefly describe how to configure BGP and supported BGP features:

- Default BGP Configuration, page 31-42
- Enabling BGP Routing, page 31-45
- Managing Routing Policy Changes, page 31-47
- Configuring BGP Decision Attributes, page 31-49
- Configuring BGP Filtering with Route Maps, page 31-51
- Configuring BGP Filtering by Neighbor, page 31-51
- Configuring Prefix Lists for BGP Filtering, page 31-53
- Configuring BGP Community Filtering, page 31-54
- Configuring BGP Neighbors and Peer Groups, page 31-55
- Configuring Aggregate Addresses, page 31-57
- Configuring a Routing Domain Confederation, page 31-58
- Configuring BGP Route Reflectors, page 31-58
- Configuring Route Dampening, page 31-59
- Monitoring and Maintaining BGP, page 31-60

For detailed descriptions of BGP configuration, see the "Configuring BGP" chapter in the "IP Routing Protocols" part of the *Cisco IOS IP Configuration Guide, Release 12.2*. For details about specific commands, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*.

For a list of BGP commands that are visible but not supported by the switch, see Appendix C, "Unsupported CLI Commands in Cisco IOS Release 12.2(25)SEE."

Default BGP Configuration

Table 31-10 shows the basic default BGP configuration. For the defaults for all characteristics, see the specific commands in the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*.

Table 31-10 Default BGP Configuration

Feature	Default Setting
Aggregate address	Disabled: None defined.
AS path access list	None defined.
Auto summary	Enabled.
Best path	• The router considers *as-path* in choosing a route and does not compare similar routes from external BGP peers. • Compare router ID: Disabled.
BGP community list	• Number: None defined. When you permit a value for the community number, the list defaults to an implicit deny for everything else that has not been permitted. • Format: Cisco default format (32-bit number).
BGP confederation identifier/peers	• Identifier: None configured. • Peers: None identified.
BGP Fast external fallover	Enabled.
BGP local preference	100. The range is 0 to 4294967295 with the higher value preferred.
BGP network	None specified; no backdoor route advertised.
BGP route dampening	Disabled by default. When enabled: • Half-life is 15 minutes. • Re-use is 750 (10-second increments). • Suppress is 2000 (10-second increments). • Max-suppress-time is 4 times half-life; 60 minutes.
BGP router ID	The IP address of a loopback interface if one is configured or the highest IP address configured for a physical interface on the router.
Default information originate (protocol or network redistribution)	Disabled.
Default metric	Built-in, automatic metric translations.
Distance	• External route administrative distance: 20 (acceptable values are from 1 to 255). • Internal route administrative distance: 200 (acceptable values are from 1 to 255). • Local route administrative distance: 200 (acceptable values are from 1 to 255).
Distribute list	• In (filter networks received in updates): Disabled. • Out (suppress networks from being advertised in updates): Disabled.
Internal route redistribution	Disabled.
IP prefix list	None defined.
Multi exit discriminator (MED)	• Always compare: Disabled. Does not compare MEDs for paths from neighbors in different autonomous systems. • Best path compare: Disabled. • MED missing as worst path: Disabled. • Deterministic MED comparison is disabled.

Table 31-10 Default BGP Configuration (continued)

Feature	Default Setting
Neighbor	• Advertisement interval: 30 seconds for external peers; 5 seconds for internal peers. • Change logging: Enabled. • Conditional advertisement: Disabled. • Default originate: No default route is sent to the neighbor. • Description: None. • Distribute list: None defined. • External BGP multihop: Only directly connected neighbors are allowed. • Filter list: None used. • Maximum number of prefixes received: No limit. • Next hop (router as next hop for BGP neighbor): Disabled. • Password: Disabled.
Neighbor	• Peer group: None defined; no members assigned. • Prefix list: None specified. • Remote AS (add entry to neighbor BGP table): No peers defined. • Private AS number removal: Disabled. • Route maps: None applied to a peer. • Send community attributes: None sent to neighbors. • Shutdown or soft reconfiguration: Not enabled. • Timers: keepalive: 60 seconds; holdtime: 180 seconds. • Update source: Best local address. • Version: BGP version 4. • Weight: Routes learned through BGP peer: 0; routes sourced by the local router: 32768.
NSF[1] Awareness	Disabled[2]. Allows Layer 3 switches to continue forwarding packets from a neighboring NSF-capable router during hardware or software changes.
Route reflector	None configured.
Synchronization (BGP and IGP)	Enabled.
Table map update	Disabled.
Timers	Keepalive: 60 seconds; holdtime: 180 seconds.

1. NSF = Nonstop Forwarding
2. NSF Awareness can be enabled on Catalyst 3550 switches with the Cisco IOS Release 12.2(25)SEC IP services image by enabling Graceful Restart.

Nonstop Forwarding Awareness

The BGP NSF awareness feature is supported in the IP services image, beginning with Cisco IOS Release 12.2(25)SEC. To enable this feature with BGP routing, you need to enable Graceful Restart. When the neighboring router is NSF-capable, and this feature is enabled, the Layer 3 switch continues

to forward packets from the neighboring router during the interval between the primary Route Processor (RP) in a router failing and the backup RP taking over, or while the primary RP is manually reloaded for a nondisruptive software upgrade.

Disabling Graceful Restart disables NSF awareness.

For more information, see the *BGP Nonstop Forwarding (NSF) Awareness Feature Guid*e at this URL:

http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a008015fede.html

Enabling BGP Routing

To enable BGP routing, you establish a BGP routing process and define the local network. Because BGP must completely understand the relationships with its neighbors, you must also specify a BGP neighbor.

BGP supports two kinds of neighbors: internal and external. *Internal neighbors* are in the same AS; *external neighbors* are in different autonomous systems. External neighbors are usually adjacent to each other and share a subnet, but internal neighbors can be anywhere in the same AS.

The switch supports the use of private AS numbers, usually assigned by service providers and given to systems whose routes are not advertised to external neighbors. The private AS numbers are from 64512 to 65535. You can configure external neighbors to remove private AS numbers from the AS path by using the **neighbor remove-private-as** router configuration command. Then when an update is passed to an external neighbor, if the AS path includes private AS numbers, these numbers are dropped.

If your AS will be passing traffic through it from another AS to a third AS, it is important to be consistent about the routes it advertises. If BGP advertised a route before all routers in the network had learned about the route through the IGP, the AS might receive traffic that some routers could not yet route. To prevent this from happening, BGP must wait until the IGP has propagated information across the AS so that BGP is *synchronized* with the IGP. Synchronization is enabled by default. If your AS does not pass traffic from one AS to another AS, or if all routers in your autonomous systems are running BGP, you can disable synchronization, which allows your network to carry fewer routes in the IGP and allows BGP to converge more quickly.

Beginning in privileged EXEC mode, follow these steps to enable BGP routing, establish a BGP routing process, and specify a neighbor:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip routing**	Enable IP routing (required only if IP routing is disabled).
Step 3	**router bgp** *autonomous-system*	Enable a BGP routing process, assign it an AS number, and enter router configuration mode. The AS number can be from 1 to 65535, with 64512 to 65535 designated as private autonomous numbers.
Step 4	**network** *network-number* [**mask** *network-mask*] [**route-map** *route-map-name*]	Configure a network as local to this AS, and enter it in the BGP table.

	Command	Purpose
Step 5	**neighbor** {*ip-address* | *peer-group-name*} **remote-as** *number*	Add an entry to the BGP neighbor table specifying that the neighbor identified by the IP address belongs to the specified AS. For EBGP, neighbors are usually directly connected, and the IP address is the address of the interface at the other end of the connection. For IBGP, the IP address can be the address of any of the router interfaces.
Step 6	**neighbor** {*ip-address* | *peer-group-name*} **remove-private-as**	(Optional) Remove private AS numbers from the AS-path in outbound routing updates.
Step 7	**no synchronization**	(Optional) Disable synchronization between BGP and an IGP.
Step 8	**no auto-summary**	(Optional) Disable automatic network summarization. By default, when a subnet is redistributed from an IGP into BGP, only the network route is inserted into the BGP table.
Step 9	**bgp fast-external-fallover**	(Optional) Automatically reset a BGP session when a link between external neighbors goes down. By default, the session is not immediately reset.
Step 10	**bgp graceful-restart**	(Optional) Enable NSF awareness on switch. By default, NSF awareness is disabled.
Step 11	**end**	Return to privileged EXEC mode.
Step 12	**show ip bgp network** *network-number* or **show ip bgp neighbor**	Verify the configuration. Verify that NSF awareness (Graceful-Restart) is enabled on the neighbor. If NSF awareness is enabled on the switch and the neighbor, this message appears: *Graceful Restart Capability: advertised and received* If NSF awareness is enabled on the switch, but not on the neighbor, this message appears: *Graceful Restart Capability: advertised*
Step 13	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no router bgp** *autonomous-system* global configuration command to remove a BGP AS. Use the **no network** *network-number* router configuration command to remove the network from the BGP table. Use the **no neighbor** {*ip-address* | *peer-group-name*} **remote-as** *number* router configuration command to remove a neighbor. Use the **no neighbor** {*ip-address* | *peer-group-name*} **remove-private-as** router configuration command to include private AS numbers in updates to a neighbor. Use the **synchronization** router configuration command to re-enable synchronization.

These examples show how to configure BGP on the routers in Figure 31-4.

Router A:

```
Switch(config)# router bgp 100
Switch(config-router)# neighbor 129.213.1.1 remote-as 200
```

Router B:

```
Switch(config)# router bgp 200
Switch(config-router)# neighbor 129.213.1.2 remote-as 100
Switch(config-router)# neighbor 175.220.1.2 remote-as 200
```

Router C:

```
Switch(config)# router bgp 200
Switch(config-router)# neighbor 175.220.212.1 remote-as 200
Switch(config-router)# neighbor 192.208.10.1 remote-as 300
```

Router D:

```
Switch(config)# router bgp 300
Switch(config-router)# neighbor 192.208.10.2 remote-as 200
```

To verify that BGP peers are running, use the **show ip bgp neighbors** privileged EXEC command. This is the output of this command on Router A:

```
Switch# show ip bgp neighbors

BGP neighbor is 129.213.1.1, remote AS 200, external link
 BGP version 4, remote router ID 175.220.212.1
 BGP state = established, table version = 3, up for 0:10:59
 Last read 0:00:29, hold time is 180, keepalive interval is 60 seconds
 Minimum time between advertisement runs is 30 seconds
 Received 2828 messages, 0 notifications, 0 in queue
 Sent 2826 messages, 0 notifications, 0 in queue
 Connections established 11; dropped 10
```

Anything other than *state = established* means that the peers are not running. The remote router ID is the highest IP address on that router (or the highest loopback interface). Each time the table is updated with new information, the table version number increments. A table version number that continually increments means that a route is flapping, causing continual routing updates.

For exterior protocols, a reference to an IP network from the **network** router configuration command controls only which networks are advertised. This is in contrast to Interior Gateway Protocols (IGPs), such as IGRP, which also use the **network** command to determine where to send updates.

For detailed descriptions of BGP configuration, see the "IP Routing Protocols" part in the *Cisco IOS IP Configuration Guide, Release 12.2*. For details about specific commands, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*. For a list of BGP commands that are visible but not supported by the switch, see Appendix C, "Unsupported CLI Commands in Cisco IOS Release 12.2(25)SEE."

Managing Routing Policy Changes

Routing policies for a peer include all the configurations that might affect inbound or outbound routing table updates. When you have defined two routers as BGP neighbors, they form a BGP connection and exchange routing information. If you later change a BGP filter, weight, distance, version, or timer, or make a similar configuration change, you must reset the BGP sessions so that the configuration changes take effect.

There are two types of reset, hard reset and soft reset. Cisco IOS software releases 12.1 and later support a soft reset without any prior configuration. To use a soft reset without preconfiguration, both BGP peers must support the soft route refresh capability, which is advertised in the OPEN message sent when the peers establish a TCP session. A soft reset allows the dynamic exchange of route refresh requests and routing information between BGP routers and the subsequent re-advertisement of the respective outbound routing table.

- When soft reset generates inbound updates from a neighbor, it is called dynamic inbound soft reset.
- When soft reset sends a set of updates to a neighbor, it is called outbound soft reset.

A soft inbound reset enables the new inbound policy to take effect. A soft outbound reset causes the new local outbound policy to take effect without resetting the BGP session. As a new set of updates is sent during outbound policy reset, a new inbound policy can also take effect.

Table 31-11 lists the advantages and disadvantages hard reset and soft reset.

Table 31-11 Advantages and Disadvantages of Hard and Soft Resets

Type of Reset	Advantages	Disadvantages
hard reset	No memory overhead.	The prefixes in the BGP, IP, and FIB tables provided by the neighbor are lost. Not recommended.
outbound soft reset	No configuration, no storing of routing table updates.	Does not reset inbound routing table updates.
dynamic inbound soft reset	Does not clear the BGP session and cache. Does not require storing of routing table updates and has no memory overhead.	Both BGP routers must support the route refresh capability (in Cisco IOS Release 12.1 and later releases).

Beginning in privileged EXEC mode, follow these steps to determine if a BGP peer supports the route refresh capability and to reset the BGP session:

	Command	Purpose
Step 1	**show ip bgp neighbors**	Display whether a neighbor supports the route refresh capability. When supported, this message appears for the router: *Received route refresh capability from peer.*
Step 2	**clear ip bgp** {* | *address* | *peer-group-name*}	Reset the routing table on the specified connection. • Enter an asterisk (*) to specify that all connections be reset. • Enter an IP *address* to specify the connection to be reset. • Enter a peer group name to reset the peer group.
Step 3	**clear ip bgp** {* | *address* | *peer-group-name*} **soft out**	(Optional) Perform an outbound soft reset to reset the inbound routing table on the specified connection. Use this command if route refresh is supported. • Enter an asterisk (*) to specify that all connections be reset. • Enter an IP *address* to specify the connection to be reset. • Enter a peer group name to reset the peer group.
Step 4	**show ip bgp** **show ip bgp neighbors**	Verify the reset by checking information about the routing table and about BGP neighbors.

Configuring BGP Decision Attributes

When a BGP speaker receives updates from multiple autonomous systems that describe different paths to the same destination, it must choose the single best path for reaching that destination. When chosen, the selected path is entered into the BGP routing table and propagated to its neighbors. The decision is based on the value of attributes that the update contains and other BGP-configurable factors.

When a BGP peer learns two EBGP paths for a prefix from a neighboring AS, it chooses the best path and inserts that path in the IP routing table. If BGP multipath support is enabled and the EBGP paths are learned from the same neighboring autonomous systems, instead of a single best path, multiple paths are installed in the IP routing table. Then, during packet switching, per-packet or per-destination load balancing is performed among the multiple paths. The **maximum-paths** router configuration command controls the number of paths allowed.

These factors summarize the order in which BGP evaluates the attributes for choosing the best path:

1. If the path specifies a next hop that is inaccessible, drop the update. The BGP next-hop attribute, automatically determined by the software, is the IP address of the next hop that is going to be used to reach a destination. For EBGP, this is usually the IP address of the neighbor specified by the **neighbor remote-as** router configuration command. You can disable next-hop processing by using route maps or the **neighbor next-hop-self** router configuration command.
2. Prefer the path with the largest weight (a Cisco proprietary parameter). The weight attribute is local to the router and not propagated in routing updates. By default, the weight attribute is 32768 for paths that the router originates and zero for other paths. Routes with the largest weight are preferred. You can use access lists, route maps, or the **neighbor weight** router configuration command to set weights.
3. Prefer the route with the highest local preference. Local preference is part of the routing update and exchanged among routers in the same AS. The default value of the local preference attribute is 100. You can set local preference by using the **bgp default local-preference** router configuration command or by using a route map.
4. Prefer the route that was originated by BGP running on the local router.
5. Prefer the route with the shortest AS path.
6. Prefer the route with the lowest origin type. An interior route or IGP is lower than a route learned by EGP, and an EGP-learned route is lower than one of unknown origin or learned in another way.
7. Prefer the route with the lowest Multi Exit Discriminator (MED) metric attribute if the neighboring AS is the same for all routes considered. You can configure the MED by using route maps or by using the **default-metric** router configuration command. When an update is sent to an IBGP peer, the MED is included.
8. Prefer the external (EBGP) path over the internal (IBGP) path.
9. Prefer the route that can be reached through the closest IGP neighbor (the lowest IGP metric). This means that the router will prefer the shortest internal path within the AS to reach the destination (the shortest path to the BGP next-hop).
10. If the following conditions are all true, insert the route for this path into the IP routing table:
    - Both the best route and this route are external.
    - Both the best route and this route are from the same neighboring autonomous system.
    - **maximum-paths** is enabled.
11. If multipath is not enabled, prefer the route with the lowest IP address value for the BGP router ID. The router ID is usually the highest IP address on the router or the loopback (virtual) address, but might be implementation-specific.

Beginning in privileged EXEC mode, follow these steps to configure some decision attributes:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router bgp** *autonomous-system*	Enable a BGP routing process, assign it an AS number, and enter router configuration mode.
Step 3	**bgp best-path as-path ignore**	(Optional) Configure the router to ignore AS path length in selecting a route.
Step 4	**neighbor** {*ip-address* | *peer-group-name*} **next-hop-self**	(Optional) Disable next-hop processing on BGP updates to a neighbor by entering a specific IP address to be used instead of the next-hop address.
Step 5	**neighbor** {*ip-address* | *peer-group-name*} **weight** *weight*	(Optional) Assign a weight to a neighbor connection. Acceptable values are from 0 to 65535; the largest weight is the preferred route. Routes learned through another BGP peer have a default weight of 0; and routes sourced by the local router have a default weight of 32768.
Step 6	**default-metric** *number*	(Optional) Set a MED metric to set preferred paths to external neighbors. All routes without a MED will also be set to this value. The range is 1 to 4294967295. The lowest value is the most desirable.
Step 7	**bgp bestpath med missing-as-worst**	(Optional) Configure the switch to consider a missing MED as having a value of infinity, making the path without a MED value the least desirable path.
Step 8	**bgp always-compare med**	(Optional) Configure the switch to compare MEDs for paths from neighbors in different autonomous systems. By default, MED comparison is only done among paths in the same AS.
Step 9	**bgp bestpath med confed**	(Optional) Configure the switch to consider the MED in choosing a path from among those advertised by different subautonomous systems within a confederation.
Step 10	**bgp deterministic med**	(Optional) Configure the switch to consider the MED variable when choosing among routes advertised by different peers in the same AS.
Step 11	**bgp default local-preference** *value*	(Optional) Change the default local preference value. The range is 0 to 4294967295; the default value is 100. The highest local preference value is preferred.
Step 12	**maximum-paths** *number*	(Optional) Configure the number of paths to be added to the IP routing table. The default is to only enter the best path in the routing table. The range is from 1 to 8. Having multiple paths allows load balancing among the paths. (Although the switch software allows a maximum of 32 equal-cost routes, the switch hardware will never use more than 8 paths per route.)
Step 13	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 14	**show ip bgp** **show ip bgp neighbors**	Verify the reset by checking information about the routing table and about BGP neighbors.
Step 15	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of each command to return to the default state.

Configuring BGP Filtering with Route Maps

Within BGP, route maps can be used to control and to modify routing information and to define the conditions by which routes are redistributed between routing domains. See the "Using Route Maps to Redistribute Routing Information" section on page 31-76 for more information about route maps. Each route map has a name that identifies the route map (*map tag*) and an optional sequence number.

Beginning in privileged EXEC mode, follow these steps to use a route map to disable next-hop processing:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**route-map** *map-tag* [[**permit** | **deny**] | *sequence-number*]]	Create a route map, and enter route-map configuration mode.
Step 3	**set ip next-hop** *ip-address* [...*ip-address*] [**peer-address**]	(Optional) Set a route map to disable next-hop processing • In an inbound route map, set the next hop of matching routes to be the neighbor peering address, overriding third-party next hops. • In an outbound route map of a BGP peer, set the next hop to the peering address of the local router, disabling the next-hop calculation.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show route-map** [*map-name*]	Display all route maps configured or only the one specified to verify configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no route-map** *map-tag* command to delete the route map. Use the **no set ip next-hop** *ip-address* command to re-enable next-hop processing.

Configuring BGP Filtering by Neighbor

You can filter BGP advertisements by using AS-path filters, such as the **as-path access-list** global configuration command and the **neighbor filter-list** router configuration command. You can also use access lists with the **neighbor distribute-list** router configuration command. Distribute-list filters are applied to network numbers. See the "Controlling Advertising and Processing in Routing Updates" section on page 31-82 for information about the **distribute-list** command.

You can use route maps on a per-neighbor basis to filter updates and to modify various attributes. A route map can be applied to either inbound or outbound updates. Only the routes that pass the route map are sent or accepted in updates. On both inbound and outbound updates, matching is supported based on AS

path, community, and network numbers. Autonomous system path matching requires the **match as-path access-lis**t route-map command, community based matching requires the **match community-list** route-map command, and network-based matching requires the **ip access-list** global configuration command.

Beginning in privileged EXEC mode, follow these steps to apply a per-neighbor route map:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router bgp** *autonomous-system*	Enable a BGP routing process, assign it an AS number, and enter router configuration mode.
Step 3	**neighbor** {*ip-address* | *peer-group name*} **distribute-list** {*access-list-number* | *name*} {**in** | **out**}	(Optional) Filter BGP routing updates to or from neighbors as specified in an access list. **Note** You can also use the **neighbor prefix-list** router configuration command to filter updates, but you cannot use both commands to configure the same BGP peer.
Step 4	**neighbor** {*ip-address* | *peer-group name*} **route-map** *map-tag* {**in** | **out**}	(Optional) Apply a route map to filter an incoming or outgoing route.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip bgp neighbors**	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no neighbor distribute-list** command to remove the access list from the neighbor. Use the **no neighbor route-map** *map-tag* router configuration command to remove the route map from the neighbor.

Another method of filtering is to specify an access list filter on both incoming and outbound updates, based on the BGP autonomous system paths. Each filter is an access list based on regular expressions. (See the "Regular Expressions" appendix in the *Cisco IOS Dial Technologies Command Reference, Release 12.1* for more information on forming regular expressions.) To use this method, define an autonomous system path access list, and apply it to updates to and from particular neighbors.

Beginning in privileged EXEC mode, follow these steps to configure BGP path filtering:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip as-path access-list** *access-list-number* {**permit** | **deny**} *as-regular-expressions*	Define a BGP-related access list.
Step 3	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
Step 4	**neighbor** {*ip-address* | *peer-group name*} **filter-list** {*access-list-number* | *name*} {**in** | **out** | **weight** *weight*}	Establish a BGP filter based on an access list.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip bgp neighbors** [**paths regular-expression**]	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring Prefix Lists for BGP Filtering

You can use prefix lists as an alternative to access lists in many BGP route filtering commands, including the **neighbor distribute-list** router configuration command. The advantages of using prefix lists include performance improvements in loading and lookup of large lists, incremental update support, easier CLI configuration, and greater flexibility.

Filtering by a prefix list involves matching the prefixes of routes with those listed in the prefix list, as when matching access lists. When there is a match, the route is used. Whether a prefix is permitted or denied is based upon these rules:

- An empty prefix list permits all prefixes.
- An implicit deny is assumed if a given prefix does not match any entries in a prefix list.
- When multiple entries of a prefix list match a given prefix, the sequence number of a prefix list entry identifies the entry with the lowest sequence number.

By default, sequence numbers are generated automatically and incremented in units of five. If you disable the automatic generation of sequence numbers, you must specify the sequence number for each entry. You can specify sequence values in any increment. If you specify increments of one, you cannot insert additional entries into the list; if you choose very large increments, you might run out of values.

You do not need to specify a sequence number when removing a configuration entry. **Show** commands include the sequence numbers in their output.

Before using a prefix list in a command, you must set up the prefix list. Beginning in privileged EXEC mode, follow these steps to create a prefix list or to add an entry to a prefix list:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip prefix-list** *list-name* [**seq** *seq-value*] **deny** | **permit** *network/len* [**ge** *ge-value*] [**le** *le-value*]	Create a prefix list with an optional sequence number to **deny** or **permit** access for matching conditions. You must enter at least one **permit** or **deny** clause. • *network/len* is the network number and length (in bits) of the network mask. • (Optional) **ge** and **le** values specify the range of the prefix length to be matched.The specified *ge-value* and *le-value* must satisfy this condition: *len* < *ge-value* < *le-value* < 32
Step 3	**ip prefix-list** *list-name* **seq** *seq-value* **deny** | **permit** *network/len* [**ge** *ge-value*] [**le** *le-value*]	(Optional) Add an entry to a prefix list, and assign a sequence number to the entry.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip prefix list** [**detail** | **summary**] *name* [*network/len*] [**seq** *seq-num*] [**longer**] [**first-match**]	Verify the configuration by displaying information about a prefix list or prefix list entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete a prefix list and all of its entries, use the **no ip prefix-list** *list-name* global configuration command. To delete an entry from a prefix list, use the **no ip prefix-list seq** *seq-value* global configuration command. To disable automatic generation of sequence numbers, use the **no ip prefix-list**

sequence number command; to reenable automatic generation, use the **ip prefix-list sequence number** command. To clear the hit-count table of prefix list entries, use the **clear ip prefix-list** privileged EXEC command.

Configuring BGP Community Filtering

One way that BGP controls the distribution of routing information based on the value of the COMMUNITIES attribute. The attribute is a way to groups destinations into communities and to apply routing decisions based on the communities. This method simplifies configuration of a BGP speaker to control distribution of routing information.

A *community* is a group of destinations that share some common attribute. Each destination can belong to multiple communities. AS administrators can define to which communities a destination belongs. By default, all destinations belong to the general Internet community. The community is identified by the COMMUNITIES attribute, an optional, transitive, global attribute in the numerical range from 1 to 4294967200. These are some predefined, well-known communities:

- **internet**—Advertise this route to the Internet community. All routers belong to it.
- **no-export**—Do not advertise this route to EBGP peers.
- **no-advertise**—Do not advertise this route to any peer (internal or external).
- **local-as**—Do not advertise this route to peers outside the local autonomous system.

Based on the community, you can control which routing information to accept, prefer, or distribute to other neighbors. A BGP speaker can set, append, or modify the community of a route when learning, advertising, or redistributing routes. When routes are aggregated, the resulting aggregate has a COMMUNITIES attribute that contains all communities from all the initial routes.

You can use community lists to create groups of communities to use in a match clause of a route map. As with an access list, a series of community lists can be created. Statements are checked until a match is found. As soon as one statement is satisfied, the test is concluded.

To set the COMMUNITIES attribute and match clauses based on communities, see the **match community-list** and **set community** route-map configuration commands in the "Using Route Maps to Redistribute Routing Information" section on page 31-76.

By default, no COMMUNITIES attribute is sent to a neighbor. You can specify that the COMMUNITIES attribute be sent to the neighbor at an IP address by using the **neighbor send-community** router configuration command.

Beginning in privileged EXEC mode, follow these steps to create and to apply a community list:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip community-list** *community-list-number* {**permit** | **deny**} *community-number*	Create a community list and assign it a number. • The *community-list-number* is an integer from 1 to 99 that identifies one or more permit or deny groups of communities. • The *community-number* is the number configured by a **set community** route-map configuration command.
Step 3	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
Step 4	**neighbor** {*ip-address* | *peer-group name*} **send-community**	Specify that the COMMUNITIES attribute be sent to the neighbor at this IP address.

	Command	Purpose
Step 5	**set comm-list** *list-num* **delete**	(Optional) Remove communities from the community attribute of an inbound or outbound update that match a standard or extended community list specified by a route map.
Step 6	**exit**	Return to global configuration mode.
Step 7	**ip bgp-community new-format**	(Optional) Display and parse BGP communities in the format AA:NN. A BGP community appears in a 2-part format two bytes long. The Cisco default community format is in the format NNAA. In the most recent RFC for BGP, a community takes the form AA:NN, where the first part is the AS number and the second part is a 2-byte number.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show ip bgp community**	Verify the configuration.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring BGP Neighbors and Peer Groups

Often many BGP neighbors are configured with the same update policies (that is, the same outbound route maps, distribute lists, filter lists, update source, and so on). Neighbors with the same update policies can be grouped into peer groups to simplify configuration and to make updating more efficient. When you have configured many peers, we recommend this approach.

To configure a BGP peer group, you create the peer group, assign options to the peer group, and add neighbors as peer group members. You configure the peer group by using the **neighbor** router configuration commands. By default, peer group members inherit all the configuration options of the peer group, including the remote-as (if configured), version, update-source, out-route-map, out-filter-list, out-dist-list, minimum-advertisement-interval, and next-hop-self. All peer group members also inherit changes made to the peer group. Members can also be configured to override the options that do not affect outbound updates.

To assign configuration options to an individual neighbor, specify any of these router configuration commands by using the neighbor IP address. To assign the options to a peer group, specify any of the commands by using the peer group name. You can disable a BGP peer or peer group without removing all the configuration information by using the **neighbor shutdown** router configuration command.

Beginning in privileged EXEC mode, use these commands to configure BGP peers:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
Step 3	**neighbor** *peer-group-name* **peer-group**	Create a BGP peer group.
Step 4	**neighbor** *ip-address* **peer-group** *peer-group-name*	Make a BGP neighbor a member of the peer group.
Step 5	**neighbor** {*ip-address* | *peer-group-name*} **remote-as** *number*	Specify a BGP neighbor. If a peer group is not configured with a **remote-as** *number*, use this command to create peer groups containing EBGP neighbors. The range is 1 to 65535.
Step 6	**neighbor** {*ip-address* | *peer-group-name*} **description** *text*	(Optional) Associate a description with a neighbor.

	Command	Purpose
Step 7	**neighbor** {*ip-address* | *peer-group-name*} **default-originate** [**route-map** *map-name*]	(Optional) Allow a BGP speaker (the local router) to send the default route 0.0.0.0 to a neighbor for use as a default route.
Step 8	**neighbor** {*ip-address* | *peer-group-name*} **send-community**	(Optional) Specify that the COMMUNITIES attribute be sent to the neighbor at this IP address.
Step 9	**neighbor** {*ip-address* | *peer-group-name*} **update-source** *interface*	(Optional) Allow internal BGP sessions to use any operational interface for TCP connections.
Step 10	**neighbor** {*ip-address* | *peer-group-name*} **ebgp-multihop**	(Optional) Allow BGP sessions, even when the neighbor is not on a directly connected segment. The multihop session is not established if the only route to the multihop peer's address is the default route (0.0.0.0).
Step 11	**neighbor** {*ip-address* | *peer-group-name*} **local-as** *number*	(Optional) Specify an AS number to use as the local AS. The range is 1 to 65535.
Step 12	**neighbor** {*ip-address* | *peer-group-name*} **advertisement-interval** *seconds*	(Optional) Set the minimum interval between sending BGP routing updates.
Step 13	**neighbor** {*ip-address* | *peer-group-name*} **maximum-prefix** *maximum* [*threshold*]	(Optional) Control how many prefixes can be received from a neighbor. The range is 1 to 4294967295. The *threshold* (optional) is the percentage of maximum at which a warning message is generated. The default is 75 percent.
Step 14	**neighbor** {*ip-address* | *peer-group-name*} **next-hop-self**	(Optional) Disable next-hop processing on the BGP updates to a neighbor.
Step 15	**neighbor** {*ip-address* | *peer-group-name*} **password** *string*	(Optional) Set MD5 authentication on a TCP connection to a BGP peer. The same password must be configured on both BGP peers, or the connection between them is not made.
Step 16	**neighbor** {*ip-address* | *peer-group-name*} **route-map** *map-name* {**in** | **out**}	(Optional) Apply a route map to incoming or outgoing routes.
Step 17	**neighbor** {*ip-address* | *peer-group-name*} **send-community**	(Optional) Specify that the COMMUNITIES attribute be sent to the neighbor at this IP address.
Step 18	**neighbor** {*ip-address* | *peer-group-name*} **timers** *keepalive holdtime*	(Optional) Set timers for the neighbor or peer group. • The *keepalive* interval is the time within which keepalive messages are sent to peers. The range is 1 to 4294967295 seconds; the default is 60. • The *holdtime* is the interval after which a peer is declared inactive after not receiving a keepalive message from it. The range is 1 to 4294967295 seconds; the default is 180.
Step 19	**neighbor** {*ip-address* | *peer-group-name*} **weight** *weight*	(Optional) Specify a weight for all routes from a neighbor.
Step 20	**neighbor** {*ip-address* | *peer-group-name*} **distribute-list** {*access-list-number* | *name*} {**in** | **out**}	(Optional) Filter BGP routing updates to or from neighbors, as specified in an access list.
Step 21	**neighbor** {*ip-address* | *peer-group-name*} **filter-list** *access-list-number* {**in** | **out** | **weight** *weight*}	(Optional) Establish a BGP filter.
Step 22	**neighbor** {*ip-address* | *peer-group-name*} **version** *value*	(Optional) Specify the BGP version to use when communicating with a neighbor.

	Command	Purpose
Step 23	**neighbor** {*ip-address* | *peer-group-name*} **soft-reconfiguration inbound**	(Optional) Configure the software to start storing received updates.
Step 24	**end**	Return to privileged EXEC mode.
Step 25	**show ip bgp neighbors**	Verify the configuration.
Step 26	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable an existing BGP neighbor or neighbor peer group, use the **neighbor shutdown** router configuration command. To enable a previously existing neighbor or neighbor peer group that had been disabled, use the **no neighbor shutdown** router configuration command.

Configuring Aggregate Addresses

Classless interdomain routing (CIDR) enables you to create aggregate routes (or *supernets*) to minimize the size of routing tables. You can configure aggregate routes in BGP either by redistributing an aggregate route into BGP or by creating an aggregate entry in the BGP routing table. An aggregate address is added to the BGP table when there is at least one more specific entry in the BGP table.

Beginning in privileged EXEC mode, use these commands to create an aggregate address in the routing table:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
Step 3	**aggregate-address** *address mask*	Create an aggregate entry in the BGP routing table. The aggregate route is advertised as coming from the AS, and the atomic aggregate attribute is set to indicate that information might be missing.
Step 4	**aggregate-address** *address mask* **as-set**	(Optional) Generate AS set path information. This command creates an aggregate entry following the same rules as the previous command, but the advertised path will be an AS_SET consisting of all elements contained in all paths. Do not use this keyword when aggregating many paths because this route must be continually withdrawn and updated.
Step 5	**aggregate-address** *address-mask* **summary-only**	(Optional) Advertise summary addresses only.
Step 6	**aggregate-address** *address mask* **suppress-map** *map-name*	(Optional) Suppress selected, more specific routes.
Step 7	**aggregate-address** *address mask* **advertise-map** *map-name*	(Optional) Generate an aggregate based on conditions specified by the route map.
Step 8	**aggregate-address** *address mask* **attribute-map** *map-name*	(Optional) Generate an aggregate with attributes specified in the route map.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show ip bgp neighbors** [**advertised-routes**]	Verify the configuration.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an aggregate entry, use the **no aggregate-address** *address mask* router configuration command. To return options to the default values, use the command with keywords.

Configuring a Routing Domain Confederation

One way to reduce the IBGP mesh is to divide an autonomous system into multiple subautonomous systems and to group them into a single confederation that appears as a single autonomous system. Each autonomous system is fully meshed within itself and has a few connections to other autonomous systems in the same confederation. Even though the peers in different autonomous systems have EBGP sessions, they exchange routing information as if they were IBGP peers. Specifically, the next hop, MED, and local preference information is preserved. You can then use a single IGP for all of the autonomous systems.

To configure a BGP confederation, you must specify a confederation identifier that acts as the autonomous system number for the group of autonomous systems.

Beginning in privileged EXEC mode, use these commands to configure a BGP confederation:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
Step 3	**bgp confederation identifier** *autonomous-system*	Configure a BGP confederation identifier.
Step 4	**bgp confederation peers** *autonomous-system* [*autonomous-system* ...]	Specify the autonomous systems that belong to the confederation and that will be treated as special EBGP peers.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip bgp neighbor** **show ip bgp network**	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring BGP Route Reflectors

BGP requires that all of the IBGP speakers be fully meshed. When a router receives a route from an external neighbor, it must advertise it to all internal neighbors. To prevent a routing information loop, all IBPG speakers must be connected. The internal neighbors do not send routes learned from internal neighbors to other internal neighbors.

With route reflectors, all IBGP speakers need not be fully meshed because another method is used to pass learned routes to neighbors. When you configure an internal BGP peer to be a *route reflector*, it is responsible for passing IBGP learned routes to a set of IBGP neighbors. The internal peers of the route reflector are divided into two groups: *client peers* and *nonclient peers (*all the other routers in the autonomous system). A route reflector reflects routes between these two groups. The route reflector and its client peers form a *cluster*. The nonclient peers must be fully meshed with each other, but the client peers need not be fully meshed. The clients in the cluster do not communicate with IBGP speakers outside their cluster.

When the route reflector receives an advertised route, it takes one of these actions, depending on the neighbor:

- A route from an external BGP speaker is advertised to all clients and nonclient peers.
- A route from a nonclient peer is advertised to all clients.
- A route from a client is advertised to all clients and nonclient peers. Hence, the clients need not be fully meshed.

Usually a cluster of clients have a single route reflector, and the cluster is identified by the route reflector router ID. To increase redundancy and to avoid a single point of failure, a cluster might have more than one route reflector. In this case, all route reflectors in the cluster must be configured with the same 4-byte cluster ID so that a route reflector can recognize updates from route reflectors in the same cluster. All the route reflectors serving a cluster should be fully meshed and should have identical sets of client and nonclient peers.

Beginning in privileged EXEC mode, use these commands to configure a route reflector and clients:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
Step 3	**neighbor** *ip-address* | *peer-group-name* **route-reflector-client**	Configure the local router as a BGP route reflector and the specified neighbor as a client.
Step 4	**bgp cluster-id** *cluster-id*	(Optional) Configure the cluster ID if the cluster has more than one route reflector.
Step 5	**no bgp client-to-client reflection**	(Optional) Disable client-to-client route reflection. By default, the routes from a route reflector client are reflected to other clients. However, if the clients are fully meshed, the route reflector does not need to reflect routes to clients.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show ip bgp**	Verify the configuration. Display the originator ID and the cluster-list attributes.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring Route Dampening

Route flap dampening is a BGP feature designed to minimize the propagation of flapping routes across an internetwork. A route is considered to be flapping when it is repeatedly available, then unavailable, then available, then unavailable, and so on. When route dampening is enabled, a numeric *penalty* value is assigned to a route when it flaps. When a route's accumulated penalties reach a configurable limit, BGP suppresses advertisements of the route, even if the route is running. The *reuse limit* is a configurable value that is compared with the penalty. If the penalty is less than the reuse limit, a suppressed route that is up will be advertised again.

Dampening is not applied to routes that are learned by IBGP. This policy prevents the IBGP peers from having a higher penalty for routes external to the AS.

Beginning in privileged EXEC mode, use these commands to configure BGP route dampening:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
Step 3	**bgp dampening**	Enable BGP route dampening.
Step 4	**bgp dampening** *half-life reuse suppress max-suppress* [**route-map** *map*]	(Optional) Change the default values of route dampening factors.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show ip bgp flap-statistics** [{**regexp** *regexp*} | {**filter-list** *list*} | {*address mask* [**longer-prefix**]}]	(Optional) Monitor the flaps of all paths that are flapping. The statistics are deleted once the route is not suppressed and is stable.
Step 7	**show ip bgp dampened-paths**	(Optional) Display the dampened routes, including the time remaining before they are suppressed.
Step 8	**clear ip bgp flap-statistics** [{**regexp** *regexp*} | {**filter-list** *list*} | {*address mask* [**longer-prefix**]}	(Optional) Clear BGP flap statistics to make it less likely that a route will be dampened.
Step 9	**clear ip bgp dampening**	(Optional) Clear route dampening information and unsuppress the suppressed routes.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable flap dampening, use the **no bgp dampening** router configuration command without keywords. To set dampening factors back to the default values, use the **no bgp dampening** router configuration command with values.

Monitoring and Maintaining BGP

You can remove all contents of a particular cache, table, or database. This might be necessary when the contents of the particular structure have become or are suspected to be invalid.

You can display specific statistics, such as the contents of BGP routing tables, caches, and databases. You can use the information to determine resource utilization and solve network problems. You can also display information about node reachability and discover the routing path your device's packets are taking through the network.

Table 31-9 lists the privileged EXEC commands for clearing and displaying BGP. For explanations of the display fields, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*.

Table 31-12 IP BGP Clear and Show Commands

Command	Purpose
clear ip bgp *address*	Reset a particular BGP connection.
clear ip bgp *	Reset all BGP connections.
clear ip bgp peer-group *tag*	Remove all members of a BGP peer group.

Table 31-12 IP BGP Clear and Show Commands (continued)

Command	Purpose
show ip bgp *prefix*	Display peer groups and peers not in peer groups to which the prefix has been advertised. Also displays prefix attributes such as the next hop and the local prefix.
show ip bgp cidr-only	Display all BGP routes that contain subnet and supernet network masks.
show ip bgp community [*community-number*] [**exact**]	Display routes that belong to the specified communities.
show ip bgp community-list *community-list-number* [**exact-match**]	Display routes that are permitted by the community list.
show ip bgp filter-list *access-list-number*	Display routes that are matched by the specified AS path access list.
show ip bgp inconsistent-as	Display the routes with inconsistent originating autonomous systems.
show ip bgp regexp *regular-expression*	Display the routes that have an AS path that matches the specified regular expression entered on the command line.
show ip bgp	Display the contents of the BGP routing table.
show ip bgp neighbors [*address*]	Display detailed information on the BGP and TCP connections to individual neighbors.
show ip bgp neighbors [*address*] [**advertised-routes** | **dampened-routes** | **flap-statistics** | **paths** *regular-expression* | **received-routes** | **routes**]	Display routes learned from a particular BGP neighbor.
show ip bgp paths	Display all BGP paths in the database.
show ip bgp peer-group [*tag*] [**summary**]	Display information about BGP peer groups.
show ip bgp summary	Display the status of all BGP connections.

You can also enable the logging of messages generated when a BGP neighbor resets, comes up, or goes down, by using the **bgp log-neighbor changes** router configuration command.

Configuring Multi-VRF CE

Virtual Private Networks (VPNs) provide a secure way for customers to share bandwidth over an ISP backbone network. A VPN is a collection of sites sharing a common routing table. A customer site is connected to the service-provider network by one or more interfaces, and the service provider associates each interface with a VPN routing table, called a VPN routing/forwarding (VRF) table.

Note VRF Lite (multi-VRF-CE) is not supported in the IP base image. In Cisco IOS Release 12.2(35)SE and later, you receive an error message if you configure the feature on the IP base image.

The switch supports multiple VPN routing/forwarding (multi-VRF) instances in customer edge (CE) devices (multi-VRF CE). Multi-VRF CE allows a service provider to support two or more VPNs with overlapping IP addresses.

Note The switch does not use Multiprotocol Label Switching (MPLS) to support VPNs. For information about MPLS VRF, see the *Cisco IOS Switching Services Configuration Guide, Release 12.2*.

This section includes these topics:

- Understanding Multi-VRF CE, page 31-62
- Default Multi-VRF CE Configuration, page 31-64
- Multi-VRF CE Configuration Guidelines, page 31-64
- Configuring VRFs, page 31-65
- Configuring a VPN Routing Session, page 31-66
- Configuring BGP PE to CE Routing Sessions, page 31-67
- Multi-VRF CE Configuration Example, page 31-67
- Displaying Multi-VRF CE Status, page 31-71

Understanding Multi-VRF CE

Multi-VRF CE is a feature that allows a service provider to support two or more VPNs, where IP addresses can be overlapped among the VPNs. Multi-VRF CE uses input interfaces to distinguish routes for different VPNs and forms virtual packet-forwarding tables by associating one or more Layer 3 interfaces with each VRF. Interfaces in a VRF can be either physical, such as Ethernet ports, or logical, such as VLAN SVIs, but an interface cannot belong to more than one VRF at any time.

Note Multi-VRF CE interfaces must be Layer 3 interfaces.

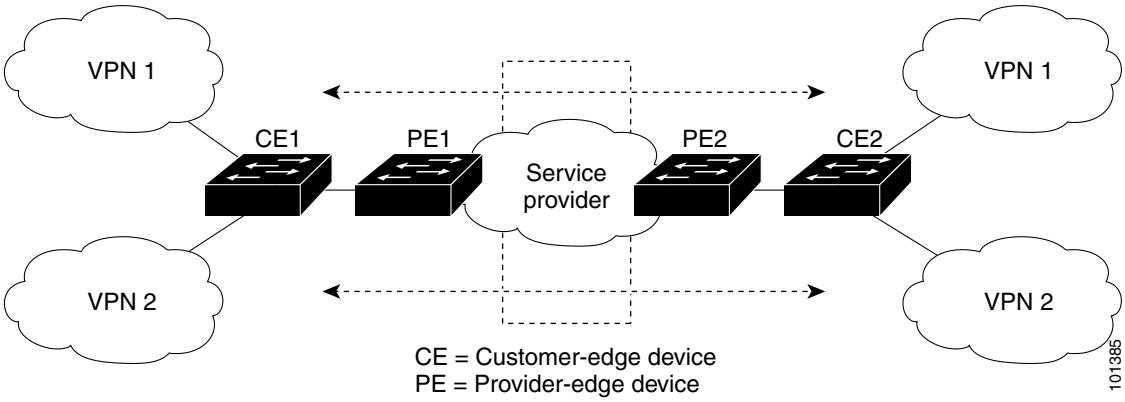
Multi-VRF CE includes these devices:

- Customer edge (CE) devices provide customers access to the service-provider network over a data link to one or more provider edge routers. The CE device advertises the site's local routes to the router and learns the remote VPN routes from it. A Catalyst 3550 switch can be a CE.
- Provider edge (PE) routers exchange routing information with CE devices by using static routing or a routing protocol such as BGP, RIPv2, OSPF, or EIGRP. The PE is only required to maintain VPN routes for those VPNs to which it is directly attached, eliminating the need for the PE to maintain all of the service-provider VPN routes. Each PE router maintains a VRF for each of its directly connected sites. Multiple interfaces on a PE router can be associated with a single VRF if all of these sites participate in the same VPN. Each VPN is mapped to a specified VRF. After learning local VPN routes from CEs, a PE router exchanges VPN routing information with other PE routers by using internal BGP (IBPG).
- Provider routers or core routers are any routers in the service provider network that do not attach to CE devices.

With multi-VRF CE, multiple customers can share one CE, and only one physical link is used between the CE and the PE. The shared CE maintains separate VRF tables for each customer and switches or routes packets for each customer based on its own routing table. Multi-VRF CE extends limited PE functionality to a CE device, giving it the ability to maintain separate VRF tables to extend the privacy and security of a VPN to the branch office.

Figure 31-5 shows a configuration using Catalyst 3550 switches as multiple virtual CEs. This scenario is suited for customers who have low bandwidth requirements for their VPN service, for example, small companies. In this case, multi-VRF CE support is required in the Catalyst 3550 switches. Because multi-VRF CE is a Layer 3 feature, each interface in a VRF must be a Layer 3 interface.

Figure 31-5 Catalyst 3550 Switches Acting as Multiple Virtual CEs



When the CE switch receives a command to add a Layer 3 interface to a VRF, it sets up the appropriate mapping between the VLAN ID and the policy label (PL) in multi-VRF-CE-related data structures and adds the VLAN ID and PL to the VLAN database.

When multi-VRF CE is configured, the Layer 3 forwarding table is conceptually partitioned into two sections:

- The multi-VRF CE routing section contains the routes from different VPNs.
- The global routing section contains routes to non-VPN networks, such as the Internet.

VLAN IDs from different VRFs are mapped into different policy labels, which are used to distinguish the VRFs during processing. If no route is found in the multi-VRF CE section of the Layer 3 forwarding table, the global routing section is used to determine the forwarding path. For each new VPN route learned, the Layer 3 setup function retrieves the policy label by using the VLAN ID of the ingress port and inserts the policy label and new route to the multi-VRF CE routing section. If the packet is received from a routed port, the port internal VLAN ID number is used; if the packet is received from an SVI, the VLAN number is used.

This is the packet-forwarding process in a multi-VRF-CE-enabled network:

- When the switch receives a packet from a VPN, the switch looks up the routing table based on the input policy label number. When a route is found, the switch forwards the packet to the PE.
- When the ingress PE receives a packet from the CE, it performs a VRF lookup. When a route is found, the router adds a corresponding MPLS label to the packet and sends it to the MPLS network.
- When an egress PE receives a packet from the network, it strips the label and uses the label to identify the correct VPN routing table. Then it performs the normal route lookup. When a route is found, it forwards the packet to the correct adjacency.
- When a CE receives a packet from an egress PE, it uses the input policy label to look up the correct VPN routing table. If a route is found, it forwards the packet within the VPN.

To configure VRF, you create a VRF table and specify the Layer 3 interface associated with the VRF. Then configure the routing protocols in the VPN and between the CE and the PE. BGP is the preferred routing protocol used to distribute VPN routing information across the provider's backbone.

The multi-VRF CE network has three major components:

- VPN route target communities—lists of all other members of a VPN community. You need to configure VPN route targets for each VPN community member.
- Multiprotocol BGP peering of VPN community PE routers—propagates VRF reachability information to all members of a VPN community. You need to configure BGP peering in all PE routers within a VPN community.
- VPN forwarding—transports all traffic between all VPN community members across a VPN service-provider network.

Default Multi-VRF CE Configuration

Table 31-13 shows the default VRF configuration.

Table 31-13 Default VRF Configuration

Feature	Default Setting
VRF	Disabled. No VRFs are defined.
Maps	No import maps, export maps, or route maps are defined.
VRF maximum routes	Fast Ethernet switches: 8000 Gigabit Ethernet switches: 12000.
Forwarding table	The default for an interface is the global routing table.

Multi-VRF CE Configuration Guidelines

Note To use multi-VRF CE, you must have the enhanced multilayer software image installed on your switch.

These are considerations when configuring VRF in your network:

- A switch with multi-VRF CE is shared by multiple customers, and each customer has its own routing table.
- Because customers use different VRF tables, the same IP addresses can be reused. Overlapped IP addresses are allowed in different VPNs.
- Multi-VRF CE lets multiple customers share the same physical link between the PE and the CE. Trunk ports with multiple VLANs separate packets among customers. Each customer has its own VLAN.
- Multi-VRF CE does not support all MPLS-VRF functionality. It does not support label exchange, LDP adjacency, or labeled packets.
- For the PE router, there is no difference between using multi-VRF CE or using multiple CEs. In Figure 31-5, multiple virtual Layer 3 interfaces are connected to the multi-VRF CE device.
- The switch supports configuring VRF by using physical ports, VLAN SVIs, or a combination of both. The SVIs can be connected through an access port or a trunk port.

- A customer can use multiple VLANs as long as they do not overlap with those of other customers. A customer's VLANs are mapped to a specific routing table ID that is used to identify the appropriate routing tables stored on the switch.
- To support multi-VRF CE, multiple routing tables are entered into the Layer 3 TCAM table. Because an extra field is needed in the routing table to identify the table to which a route belongs, you must modify the SDM template to enable the switch to support 144-bit Layer 3 TCAM. Use the **sdm prefer extended-match**, **sdm prefer access extended-match**, or **sdm prefer routing extended-match** global configuration command to reformat the TCAM space allocated to unicast routing in the default, access, or routing template, respectively. Reformatting the unicast routing TCAM halves the number of supported unicast routes in the template.

Note For more information on the SDM templates, see the "Optimizing System Resources for User-Selected Features" section on page 6-26.

- A Catalyst 3550 switch supports one global network and up to seven VRFs. The total number of routes supported are limited by the size of the TCAM and specified in the SDM template.
- Most routing protocols (BGP, OSPF, RIP, and static routing) can be used between the CE and the PE. However, we recommend using external BGP (EBGP) for these reasons:
  - BGP does not require multiple algorithms to communicate with multiple CEs.
  - BGP is designed for passing routing information between systems run by different administrations.
  - BGP makes it easy to pass attributes of the routes to the CE.
- Multi-VRF-CE does not support EIGRP.
- Multi-VRF CE does not affect the packet switching rate.
- VPN multicast is not supported.
- You cannot configure the Web Cache Communication Protocol (WCCP) and multi-VRF CE on the same switch at the same time.
- When multi-VRF CE is configured, you cannot assign the same Hot Standby Routing Protocol (HSRP) standby address to two different VPNs.
- VRF and policy-based routing (PBR) are mutually-exclusive on a switch interface. You cannot enable VRF when PBR is enabled on an interface. In contrast, you cannot enable PBR when VRF is enabled on an interface.

Configuring VRFs

Beginning in privileged EXEC mode, follow these steps to configure one or more VRFs. For complete syntax and usage information for the commands, see the switch command reference for this release and the *Cisco IOS Switching Services Command Reference, Release 12.2*.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip routing**	Enable IP routing.
Step 3	**ip vrf** *vrf-name*	Name the VRF, and enter VRF configuration mode.

	Command	Purpose
Step 4	**rd** *route-distinguisher*	Create a VRF table by specifying a route distinguisher. Enter either an AS number and an arbitrary number (xxx:y) or an IP address and arbitrary number (A.B.C.D:y)
Step 5	**route-target** {**export** | **import** | **both**} *route-target-ext-community*	Create a list of import, export, or import and export route target communities for the specified VRF. Enter either an AS system number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y). The *route-target-ext-community* should be the same as the *route-distinguisher* entered in Step 4.
Step 6	**import map** *route-map*	(Optional) Associate a route map with the VRF.
Step 7	**interface** *interface-id*	Enter interface configuration mode and specify the Layer 3 interface to be associated with the VRF. The interface can be a routed port or SVI.
Step 8	**ip vrf forwarding** *vrf-name*	Associate the VRF with the Layer 3 interface.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show ip vrf** [**brief** | **detail** | **interfaces**] [*vrf-name*]	Verify the configuration. Display information about the configured VRFs.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip vrf** *vrf-name* global configuration command to delete a VRF and to remove all interfaces from it. Use the **no ip vrf forwarding** interface configuration command to remove an interface from the VRF.

Configuring a VPN Routing Session

Routing within the VPN can be configured with any supported routing protocol (RIP, OSPF, EIGRP, or BGP) or with static routing. The configuration shown here is for OSPF, but the process is the same for other protocols.

Beginning in privileged EXEC mode, follow these steps to configure OSPF in the VPN:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router ospf** *process-id* **vrf** *vrf-name*	Enable OSPF routing, specify a VPN forwarding table, and enter router configuration mode.
Step 3	**log-adjacency-changes**	(Optional) Log changes in the adjacency state. This is the default state.
Step 4	**redistribute bgp** *autonomous-system-number* **subnets**	Set the switch to redistribute information from the BGP network to the OSPF network.
Step 5	**network** *network-number* **area** *area-id*	Define a network address and mask on which OSPF runs and the area ID for that network address.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show ip ospf** *process-id*	Verify the configuration of the OSPF network.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no router ospf** *process-id* **vrf** *vrf-name* global configuration command to disassociate the VPN forwarding table from the OSPF routing process.

Configuring BGP PE to CE Routing Sessions

Beginning in privileged EXEC mode, follow these steps to configure a BGP PE to CE routing session:

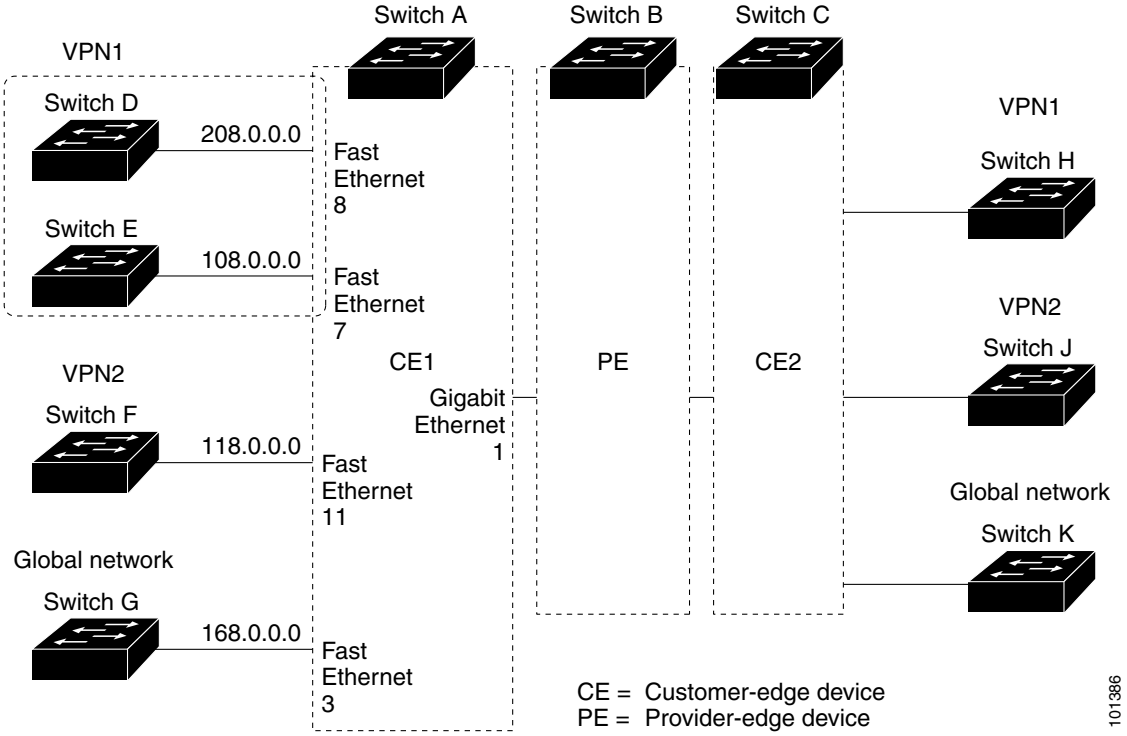
	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router bgp** *autonomous-system-number*	Configure the BGP routing process with the AS number passed to other BGP routers, and enter router configuration mode.
Step 3	**network** *network-number* **mask** *network-mask*	Specify a network and mask to announce using BGP.
Step 4	**redistribute ospf** *process-id* **match internal**	Set the switch to redistribute OSPF internal routes.
Step 5	**network** *network-number* **area** *area-id*	Define a network address and mask on which OSPF runs and the area ID for that network address.
Step 6	**address-family ipv4 vrf** *vrf-name*	Define BGP parameters for PE to CE routing sessions, and enter VRF address-family mode.
Step 7	**neighbor** *address* **remote-as** *as-number*	Define a BGP session between PE and CE routers.
Step 8	**neighbor** *address* **activate**	Activate the advertisement of the IPv4 address family.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show ip bgp** [**ipv4**] [**neighbors**]	Verify BGP configuration.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no router bgp** *autonomous-system-number* global configuration command to delete the BGP routing process. Use the command with keywords to delete routing characteristics.

Multi-VRF CE Configuration Example

Figure 31-6 is a simplified example of the physical connections in a network similar to that in Figure 31-5. OSPF is the protocol used in VPN1, VPN2, and the global network. BGP is used in the CE to PE connections. The examples following the illustration show how to configure a Catalyst 3550 switch as CE Switch A, and the VRF configuration for customer switches D and F. Commands for configuring CE Switch C and the other customer switches are not included but would be similar. The example also includes commands for configuring traffic to Switch A for a Catalyst 6000 or Catalyst 6500 switch acting as a PE router.

Figure 31-6 Multi-VRF CE Configuration Example



Configuring Switch A

On Switch A, enable routing and configure VRF.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip routing
Switch(config)# ip vrf v11
Switch(config-vrf)# rd 800:1
Switch(config-vrf)# route-target export 800:1
Switch(config-vrf)# route-target import 800:1
Switch(config-vrf)# exit
Switch(config)# ip vrf v12
Switch(config-vrf)# rd 800:2
Switch(config-vrf)# route-target export 800:2
Switch(config-vrf)# route-target import 800:2
Switch(config-vrf)# exit
```

Configure the loopback and physical interfaces on Switch A. Gigabit Ethernet port 1 is a trunk connection to the PE. Fast Ethernet ports 8 and 11 connect to VPNs:

```
Switch(config)# interface loopback1
Switch(config-if)# ip vrf forwarding v11
Switch(config-if)# ip address 8.8.1.8 255.255.255.0
Switch(config-if)# exit

Switch(config)# interface loopback2
Switch(config-if)# ip vrf forwarding v12
Switch(config-if)# ip address 8.8.2.8 255.255.255.0
Switch(config-if)# exit
```

```
Switch(config)# interface gigabitethernet0/5
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# no ip address
Switch(config-if)# exit

Switch(config)# interface fastethernet0/8
Switch(config-if)# switchport access vlan 208
Switch(config-if)# no ip address
Switch(config-if)# exit

Switch(config)# interface fastethernet0/11
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# no ip address
Switch(config-if)# exit
```

Configure the VLANs used on Switch A. VLAN 10 is used by VRF 11 between the CE and the PE. VLAN 20 is used by VRF 12 between the CE and the PE. VLANs 118 and 208 are used for VRF for the VPNs that include Switch F and Switch D, respectively:

```
Switch(config)# interface vlan10
Switch(config-if)# ip vrf forwarding v11
Switch(config-if)# ip address 38.0.0.8 255.255.255.0
Switch(config-if)# exit

Switch(config)# interface vlan20
Switch(config-if)# ip vrf forwarding v12
Switch(config-if)# ip address 83.0.0.8 255.255.255.0
Switch(config-if)# exit

Switch(config)# interface vlan118
Switch(config-if)# ip vrf forwarding v12
Switch(config-if)# ip address 118.0.0.8 255.255.255.0
Switch(config-if)# exit

Switch(config)# interface vlan208
Switch(config-if)# ip vrf forwarding v11
Switch(config-if)# ip address 208.0.0.8 255.255.255.0
Switch(config-if)# exit
```

Configure OSPF routing in VPN1 and VPN2.

```
Switch(config)# router ospf 1 vrf vl1
Switch(config-router)# redistribute bgp 800 subnets
Switch(config-router)# network 208.0.0.0 0.0.0.255 area 0
Switch(config-router)# exit
Switch(config)# router ospf 2 vrf vl2
Switch(config-router)# redistribute bgp 800 subnets
Switch(config-router)# network 118.0.0.0 0.0.0.255 area 0
Switch(config-router)# exit
```

Configure BGP for CE to PE routing.

```
Switch(config)# router bgp 800
Switch(config-router)# address-family ipv4 vrf vl2
Switch(config-router-af)# redistribute ospf 2 match internal
Switch(config-router-af)# neighbor 83.0.0.3 remote-as 100
Switch(config-router-af)# neighbor 83.0.0.3 activate
Switch(config-router-af)# network 8.8.2.0 mask 255.255.255.0
Switch(config-router-af)# exit
```

```
Switch(config-router)# address-family ipv4 vrf vl1
Switch(config-router-af)# redistribute ospf 1 match internal
Switch(config-router-af)# neighbor 38.0.0.3 remote-as 100
Switch(config-router-af)# neighbor 38.0.0.3 activate
Switch(config-router-af)# network 8.8.1.0 mask 255.255.255.0
Switch(config-router-af)# end
```

Configuring Switch D

Switch D belongs to VPN 1 and is connected to Switch A by using these commands.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip routing
Switch(config)# interface fastethernet0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 208.0.0.20 255.255.255.0
Switch(config-if)# exit

Switch(config)# router ospf 101
Switch(config-router)# network 208.0.0.0 0.0.0.255 area 0
Switch(config-router)# end
```

Configuring Switch F

Switch F belongs to VPN 2 and is connected to Switch A by using these commands.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip routing
Switch(config)# interface fastethernet0/1
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# no ip address
Switch(config-if)# exit

Switch(config)# interface vlan118
Switch(config-if)# ip address 118.0.0.11 255.255.255.0
Switch(config-if)# exit
Switch(config)# router ospf 101
Switch(config-router)# network 118.0.0.0 0.0.0.255 area 0
Switch(config-router)# end
```

Configuring the PE Switch B

On Switch B (the PE router), these commands only configure the connections to the CE device, Switch A.

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ip vrf v1
Router(config-vrf)# rd 100:1
Router(config-vrf)# route-target export 100:1
Router(config-vrf)# route-target import 100:1
Router(config-vrf)# exit

Router(config)# ip vrf v2
Router(config-vrf)# rd 100:2
Router(config-vrf)# route-target export 100:2
Router(config-vrf)# route-target import 100:2
Router(config-vrf)# exit

Router(config)# ip cef
```

```
Router(config)# interface Loopback1
Router(config-if)# ip vrf forwarding v1
Router(config-if)# ip address 3.3.1.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface Loopback2
Router(config-if)# ip vrf forwarding v2
Router(config-if)# ip address 3.3.2.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface gigabitethernet1/0.10
Router(config-if)# encapsulation dot1q 10
Router(config-if)# ip vrf forwarding v1
Router(config-if)# ip address 38.0.0.3 255.255.255.0
Router(config-if)# exit

Router(config)# interface gigabitethernet1/0.20
Router(config-if)# encapsulation dot1q 20
Router(config-if)# ip vrf forwarding v2
Router(config-if)# ip address 83.0.0.3 255.255.255.0
Router(config-if)# exit

Router(config)# router bgp 100
Router(config-router)# address-family ipv4 vrf v2
Router(config-router-af)# neighbor 83.0.0.8 remote-as 800
Router(config-router-af)# neighbor 83.0.0.8 activate
Router(config-router-af)# network 3.3.2.0 mask 255.255.255.0
Router(config-router-af)# exit
Router(config-router)# address-family ipv4 vrf v1
Router(config-router-af)# neighbor 83.0.0.8 remote-as 800
Router(config-router-af)# neighbor 83.0.0.8 activate
Router(config-router-af)# network 3.3.1.0 mask 255.255.255.0
Router(config-router-af)# end
```

Displaying Multi-VRF CE Status

You can use the privileged EXEC commands in Table 31-14 to display information about multi-VRF CE configuration and status.

Table 31-14 Commands for Displaying Multi-VRF CE Information

Command	Purpose
show ip protocols vrf *vrf-name*	Display routing protocol information associated with a VRF.
show ip route vrf *vrf-name* [**connected**] [*protocol* [*as-number*]] [**list**] [**mobile**] [**odr**] [**profile**] [**static**] [**summary**] [**supernets-only**]	Display IP routing table information associated with a VRF.
show ip vrf [**brief** | **detail** | **interfaces**] [*vrf-name*]	Display information about the defined VRF instances.

For more information about the information in the displays, see the *Cisco IOS Switching Services Command Reference, Release 12.2*.

Configuring Protocol-Independent Features

This section describes how to configure IP routing protocol-independent features. These features are available on switches running the IP base image or the IP services image, but protocol-related features with the IP base image are available only for RIP. For a complete description of the IP routing protocol-independent commands in this chapter, see the "IP Routing Protocol-Independent Commands" chapter of the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*.

This section includes these procedures:

- Configuring Cisco Express Forwarding, page 31-72
- Configuring the Number of Equal-Cost Routing Paths, page 31-73
- Configuring Static Unicast Routes, page 31-74
- Specifying Default Routes and Networks, page 31-75
- Using Route Maps to Redistribute Routing Information, page 31-76
- Configuring Policy-Based Routing, page 31-79
- Filtering Routing Information, page 31-81
- Managing Authentication Keys, page 31-83

Configuring Cisco Express Forwarding

Cisco Express Forwarding (CEF) is a Layer 3 IP switching technology used to optimize network performance. CEF implements an advanced IP look-up and forwarding algorithm to deliver maximum Layer 3 switching performance. CEF is less CPU-intensive than fast switching route caching, allowing more CPU processing power to be dedicated to packet forwarding. In dynamic networks, fast switching cache entries are frequently invalidated because of routing changes, which can cause traffic to be process-switched by using the routing table, instead of fast-switched by using the route cache. CEF uses the forwarding information base (FIB) lookup table to perform destination-based switching of IP packets.

The two main components in CEF are the FIB and adjacency tables.

- The FIB is similar to a routing table or information base and maintains a mirror image of the forwarding information in the IP routing table. When routing or topology changes occur in the network, the IP routing table is updated, and those changes are reflected in the FIB. The FIB maintains next-hop address information based on the information in the IP routing table. Because the FIB contains all known routes that exist in the routing table, CEF eliminates route cache maintenance, is more efficient for switching traffic, and is not affected by traffic patterns.
- Nodes in the network are said to be adjacent if they can reach each other with a single hop across a link layer. CEF uses adjacency tables to prepend Layer 2 addressing information. The adjacency table maintains Layer 2 next-hop addresses for all FIB entries.

Because the switch uses an Application Specific Integrated Circuit (ASIC) to achieve Gigabit-speed line rate IP traffic, CEF forwarding applies only to the software forwarding path, that is, traffic that is forwarded by the CPU.

CEF is enabled globally by default. If for some reason it is disabled, you can re-enable it by using the **ip cef** global configuration command.

The default configuration, which we recommend, is CEF enabled on all Layer 3 interfaces. On the switch, you can use the **no ip route-cache cef** interface configuration command to disable CEF for traffic that is being forwarded by software. This command does not affect the hardware forwarding path.

Disabling CEF and using the **debug ip packet detail** privileged EXEC command can be useful when you want to debug software-forwarded traffic. You can enable CEF on an interface for the software-forwarding path by using the **ip route-cache cef** interface configuration command.

Beginning in privileged EXEC mode, follow these steps to enable CEF on an interface for software-forwarded traffic after it has been disabled:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
Step 3	**ip route-cache cef**	Enable CEF on the interface for software-forwarded traffic.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip cef**	Display the CEF status on all interfaces.
Step 6	**show adjacency**	Display CEF adjacency table information.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable CEF on an interface for software-forwarded traffic, use the **no ip route-cache cef** interface configuration command.

Configuring the Number of Equal-Cost Routing Paths

When a router has two or more routes to the same network with the same metrics, these routes can be thought of as having an equal cost. The term *parallel path* is another way to refer to occurrences of equal-cost routes in a routing table. If a router has two or more equal-cost paths to a network, it can use them concurrently. Parallel paths provide redundancy in case of a circuit failure and also enable a router to load balance packets over the available paths for more efficient use of available bandwidth.

Even though the router automatically learns about and configures equal-cost routes, you can control the maximum number of parallel paths supported by an IP routing protocol in its routing table. Although the switch software allows a maximum of 32 equal-cost routes, the switch hardware will never use more than 8 paths per route.

Beginning in privileged EXEC mode, follow these steps to change the maximum number of parallel paths installed in a routing table from the default:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router** {**bgp** |**rip** | **ospf** | **eigrp**}	Enter router configuration mode.
Step 3	**maximum-paths** *maximum*	Set the maximum number of parallel paths for the protocol routing table. The range is from 1 to 8; the default is 4 for most IP routing protocols, but only 1 for BGP.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip protocols**	Verify the setting in the *Maximum path* field.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no maximum-paths** router configuration command to restore the default value.

Configuring Static Unicast Routes

Static unicast routes are user-defined routes that cause packets moving between a source and a destination to take a specified path. Static routes can be important if the router cannot build a route to a particular destination and are useful for specifying a gateway of last resort to which all unroutable packets are sent.

Beginning in privileged EXEC mode, follow these steps to configure a static unicast route:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip route** *prefix mask* {*address* | *interface*} [*distance*]	Establish a static route.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip route**	Display the current state of the routing table to verify the configuration.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip route** *prefix mask* global configuration command to remove a static route.

The switch retains static routes until you remove them. However, you can override static routes with dynamic routing information by assigning administrative distance values. Each dynamic routing protocol has a default administrative distance, as listed in Table 31-15. If you want a static route to be overridden by information from a dynamic routing protocol, set the administrative distance of the static route higher than that of the dynamic protocol.

Table 31-15 Dynamic Routing Protocol Default Administrative Distances

Route Source	Default Distance
Connected interface	0
Static route	1
EIGRP summary route	5
External BGP	20
Internal Enhanced IGRP	90
OSPF	110
RIP	120
EIGRP summary route	170
Internal BGP	200
Unknown	225

Static routes that point to an interface are advertised through RIP and other dynamic routing protocols, whether or not static **redistribute** router configuration commands were specified for those routing protocols. These static routes are advertised because static routes that point to an interface are considered

in the routing table to be connected and hence lose their static nature. However, if you define a static route to an interface that is not one of the networks defined in a network command, no dynamic routing protocols advertise the route unless a **redistribute** static command is specified for these protocols.

When an interface goes down, all static routes through that interface are removed from the IP routing table. When the software can no longer find a valid next hop for the address specified as the forwarding router's address in a static route, the static route is also removed from the IP routing table.

Specifying Default Routes and Networks

A router might not be able to determine the routes to all other networks. To provide complete routing capability, you can use some routers as smart routers and give the remaining routers default routes to the smart router. (Smart routers have routing table information for the entire internetwork.) These default routes can be dynamically learned or can be configured in the individual routers. Most dynamic interior routing protocols include a mechanism for causing a smart router to generate dynamic default information that is then forwarded to other routers.

If a router has a directly connected interface to the specified default network, the dynamic routing protocols running on that device generate a default route. In RIP, it advertises the pseudonetwork 0.0.0.0.

A router that is generating the default for a network also might need a default of its own. One way a router can generate its own default is to specify a static route to the network 0.0.0.0 through the appropriate device.

Beginning in privileged EXEC mode, follow these steps to define a static route to a network as the static default route:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip default-network** *network number*	Specify a default network.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip route**	Display the selected default route in the gateway of last resort display.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip default-network** *network number* global configuration command to remove the route.

When default information is passed through a dynamic routing protocol, no further configuration is required. The system periodically scans its routing table to choose the optimal default network as its default route. Cisco routers use administrative distance and metric information to determine the default route or the gateway of last resort.

If dynamic default information is not being passed to the system, candidates for the default route are specified with the **ip default-network** global configuration command. If this network appears in the routing table from any source, it is flagged as a possible choice for the default route. If the router has no interface on the default network, but does have a path to it, the network is considered as a possible candidate, and the gateway to the best default path becomes the gateway of last resort.

Using Route Maps to Redistribute Routing Information

The switch can run multiple routing protocols simultaneously, and it can redistribute information from one routing protocol to another. Redistributing information from one routing protocol to another applies to all supported IP-based routing protocols.

You can conditionally control the redistribution of routes between routing domains by defining route maps between the two domains. The **match** and **set** route-map configuration commands define the condition portion of a route map. The **match** command specifies that a criterion must be matched; the **set** command specifies an action to be taken if the routing update meets the conditions defined by the match command. Although redistribution is a protocol-independent feature, some of the **match** and **set** route-map configuration commands are specific to a particular protocol.

One or more **match** commands and one or more **set** commands follow a **route-map** command. If there are no **match** commands, everything matches. If there are no **set** commands, nothing is done, other than the match. Therefore, you need at least one **match** or **set** command.

Note Although each of Steps 3 through 16 in the following section is optional, you must enter at least one **match** route-map configuration command and one **set** route-map configuration command. For complete syntax information for the command, see the *Cisco IOS IP and IP Routing Command Reference, Release 12.2*.

Beginning in privileged EXEC mode, follow these steps to configure a route map for redistribution:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**route-map** *map-tag* [**permit** | **deny**] [*sequence number*]	Define any route maps used to control redistribution and enter route-map configuration mode. *map-tag*—A meaningful name for the route map. The **redistribute** router configuration command uses this name to reference the route map. Multiple route maps might share the same map tag name. (Optional) If **permit** is specified and the match criteria are met for this route map, the route is redistributed as controlled by the set actions. If **deny** is specified, the route is not redistributed. *sequence number* (Optional)— Number that indicates the position of a new route map in the list of route maps already configured with the same name.
Step 3	**match as-path** *path-list-number*	Match a BGP AS path access list.
Step 4	**match community-list** *community-list-number* [**exact**]	Match a BGP community list.
Step 5	**match ip address** {*access-list-number* | *access-list-name*} [*...access-list-number* | *...access-list-name*]	Match a standard access list by specifying the name or number. It can be an integer from 1 to 199.
Step 6	**match ip next-hop** {*access-list-number* | *access-list-name*} [*...access-list-number* | *...access-list-name*]	Match a next-hop router address passed by one of the access lists specified (numbered from 1 to 199).

	Command	Purpose
Step 7	**match tag** *tag value* [*...tag-value*]	Match the specified tag value in a list of one or more route tag values. Each can be an integer from 0 to 4294967295.
Step 8	**match interface** *type number* [*...type number*]	Match the specified next hop route out one of the specified interfaces.
Step 9	**match ip route-source** {*access-list-number* | *access-list-name*} [*...access-list-number* | *...access-list-name*]	Match the address specified by the specified advertised access lists.
Step 10	**match route-type** {**local** | **internal** | **external** [**type-1** | **type-2**]	Match the specified **route-type**: • **local**—Locally generated BGP routes. • **internal**—OSPF intra-area and interarea routes or EIGRP internal routes. • **external**—OSPF external routes (Type 1 or Type 2) or EIGRP external routes.
Step 11	**set dampening** *halflife reuse suppress max-suppress-time*	Set BGP route dampening factors.
Step 12	**set local-preference** *value*	Assign a value to a local BGP path.
Step 13	**set origin** {**igp** | **egp** as | **incomplete**}	Set the BGP origin code.
Step 14	**set as-path** {**tag** | **prepend** *as-path-string*}	Modify the BGP autonomous system path.
Step 15	**set level** {**level-1** | **level-2** | **level-1-2** | **stub-area** | **backbone**}	Set the level for routes that are advertised into the specified area of the routing domain. The **stub-area** and **backbone** are OSPF NSSA and backbone areas.
Step 16	**set metric** *metric value*	Set the metric value to give the redistributed routes (for any protocol except EIGRP). The *metric value* is an integer from –294967295 to 294967295.
Step 17	**set metric** *bandwidth delay reliability loading mtu*	Set the metric value to give the redistributed routes (for EIGRP only): • *bandwidth*—Metric value or in kilobits per second in the range 0 to 4294967295. • *delay*—Route delay in tens of microseconds in the range 0 to 4294967295. • *reliability*—Likelihood of successful packet transmission expressed as a number between 0 (no reliability) and 255 (100 percent reliability). • *loading*— Effective bandwidth of the route expressed as a number from 0 to 255 (100 percent loading). • *mtu*—Maximum transmission unit (MTU) size of the route in bytes in the range 0 to 4294967295.
Step 18	**set metric-type** {**internal** | **external** | **type-1** | **type-2**}	Set the metric type to give redistributed routes.
Step 19	**set metric-type internal**	Set the multi-exit discriminator (MED) value on prefixes advertised to External BGP neighbor to match the IGP metric of the next hop
Step 20	**set weight**	Set the BGP weight for the routing table. The value can be from 1 to 65535.

	Command	Purpose
Step 21	**end**	Return to privileged EXEC mode.
Step 22	**show route-map**	Display all route maps configured or only the one specified to verify configuration.
Step 23	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an entry, use the **no route-map** *map tag* global configuration command or the **no match** or **no set** route-map configuration commands.

You can distribute routes from one routing domain into another and control route distribution.

Beginning in privileged EXEC mode, follow these steps to control route redistribution. Note that the keywords are the same as defined in the previous procedure.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router** {**bgp** |**rip** | **ospf** | **eigrp**}	Enter router configuration mode.
Step 3	**redistribute** *protocol* [*process-id*] {**level-1** | **level-1-2** | **level-2**} [**metric** *metric-value*] [**metric-type** *type-value*] [**match internal** | **external** *type-value*] [**tag** *tag-value*] [**route-map** *map-tag*] [**weight** *weight*] [**subnets**]	Redistribute routes from one routing protocol to another routing protocol.
Step 4	**default-metric** *number*	Cause the current routing protocol to use the same metric value for all redistributed routes (BGP, RIP, and OSPF).
Step 5	**default-metric** *bandwidth delay reliability loading mtu*	Cause the EIGRP routing protocol to use the same metric value for all non-EIGRP redistributed routes.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show route-map**	Display all route maps configured or only the one specified to verify configuration.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable redistribution, use the **no** form of the commands.

The metrics of one routing protocol do not necessarily translate into the metrics of another. For example, the RIP metric is a hop count. In these situations, an artificial metric is assigned to the redistributed route. Uncontrolled exchanging of routing information between different routing protocols can create routing loops and seriously degrade network operation.

If you have not defined a default redistribution metric that replaces metric conversion, some automatic metric translations occur between routing protocols:

- RIP can automatically redistribute static routes. It assigns static routes a metric of 1 (directly connected).
- Any protocol can redistribute other routing protocols if a default mode is in effect.

Configuring Policy-Based Routing

You can use policy-based routing (PBR) to configure a defined policy for traffic flows. By using PBR, you can have more control over routing by reducing the reliance on routes derived from routing protocols. PBR can determine and implement routing policies that allow or deny paths based on:

- Identity of a particular end system
- Application
- Protocol

You can use PBR to provide equal-access and source-sensitive routing, routing based on interactive versus batch traffic, or routing based on dedicated links. For example, you could transfer stock records to a corporate office on a high-bandwidth, high-cost link for a short time while transmitting routine application data such as e-mail over a low-bandwidth, low-cost link.

PBR is applied to incoming packets. All packets received on an interface with PBR enabled are considered for PBR. The switch passes the packets through route maps. Based on the criteria defined in the route maps, packets are forwarded (routed) to the appropriate next hop. For more information about configuring route maps see the "Using Route Maps to Redistribute Routing Information" section on page 31-76.

For details about PBR commands and keywords, see the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.2*. For a list of PBR commands not supported by the switch, see Appendix C, "Unsupported CLI Commands in Cisco IOS Release 12.2(25)SEE."

PBR Configuration Guidelines

Before configuring PBR, you should be aware of this information:

- To use PBR, you must have the IP services image installed on your switch.
- Multicast traffic is not policy-routed. PBR applies to only to unicast traffic.
- You can enable PBR on a routed port, an SVI, or an EtherChannel port channel in Layer 3 mode.
- You can define a maximum of 247 IP policy route-maps on the switch.
- VRF and PBR are mutually-exclusive on a switch interface. You cannot enable VRF when PBR is enabled on an interface. In contrast, you cannot enable PBR when VRF is enabled on an interface.
- WCCP and PBR are mutually-exclusive on a switch interface. You cannot enable WCCP when PBR is enabled on an interface. In contrast, you cannot enable PBR when WCCP is enabled on an interface.
- The number of TCAM entries used by PBR depends on the route-map itself, the ACLs used, and the order of the ACLs and route-map entries.
- You must modify the SDM template to enable the switch to support the 144-bit Layer 3 TCAM. Use the **sdm prefer extended-match**, **sdm prefer access extended-match**, or the **sdm prefer routing extended-match** global configuration commands to reformat the TCAM space allocated to unicast routing in the default, access, or routing template, respectively. Reformatting the unicast routing TCAM reduces by half the number of supported unicast routes in the template.

  See the "Optimizing System Resources for User-Selected Features" section on page 6-26 and the "Displaying ACL Resource Usage and Configuration Problems" section on page 29-42 for more information about managing the memory resources in the switch.

Enabling PBR

By default, PBR is disabled on the switch. To enable PBR, you must create a route map that specifies the match criteria and the resulting action if all of the match clauses are met. Then, you must enable PBR for that route map on an interface. All packets arriving on the specified interface matching the match clauses will be subject to PBR.

Beginning in privileged EXEC mode, follow these steps to configure PBR:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**route-map** *map-tag* [**permit** | **deny**] [*sequence number*]	Define any route maps used to control where packets are output and enter route-map configuration mode. *map-tag*—A meaningful name for the route map. The **ip policy route-map** interface configuration command uses this name to reference the route map. Multiple route maps might share the same map tag name. (Optional) If **permit** is specified and the match criteria are met for this route map, the route is policy-routed as controlled by the set actions. If **deny** is specified, the route is not policy-routed. *sequence number* (Optional)— Number that shows the position of a new route map in the list of route maps already configured with the same name.
Step 3	**match ip address** {*access-list-number* | *access-list-name*} [*...access-list-number* | *...access-list-name*]	Match the source and destination IP address that is permitted by one or more standard or extended access lists. If you do not specify a **match** command, the route map applies to all packets.
Step 4	**set ip next-hop** *ip-address* [*...ip-address*]	Specify the action to take on the packets that match the criteria. Set next hop to which to route the packet (the next hop must be adjacent).
Step 5	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
Step 6	**ip policy route-map** *map-tag*	Enable PBR on a Layer 3 interface, and identify the route map to use. You can configure only one route map on an interface. However, you can have multiple route map entries with different sequence numbers. These entries are evaluated in sequence number order until the first match. If there is no match, packets are routed as usual.
Step 7	**ip route-cache policy**	(Optional) Enable fast-switching PBR. You must first enable PBR before enabling fast-switching PBR.
Step 8	**exit**	Return to global configuration mode.
Step 9	**ip local policy route-map** *map-tag*	(Optional) Enable local PBR to perform policy-based routing on packets originating at the switch. This applies to packets generated by the switch and not to incoming packets.

	Command	Purpose
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**show route-map** [*map-name*]	Display all route maps configured or only the one specified to verify configuration.
Step 12	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no route-map** *map-tag* global configuration command or the **no match** or **no set** route-map configuration commands to delete an entry. Use the **no ip policy route-map** *map-tag* interface configuration command to disable PBR on an interface.

Filtering Routing Information

You can filter routing protocol information by performing the tasks described in this section.

Note When routes are redistributed between OSPF processes, no OSPF metrics are preserved.

Setting Passive Interfaces

To prevent other routers on a local network from dynamically learning about routes, you can use the **passive-interface** router configuration command to keep routing update messages from being sent through a router interface. When you use this command in the OSPF protocol, the interface address you specify as passive appears as a stub network in the OSPF domain. OSPF routing information is neither sent nor received through the specified router interface.

In networks with many interfaces, to avoid having to manually set them as passive, you can set all interfaces to be passive by default by using the **passive-interface default** router configuration command and manually setting interfaces where adjacencies are desired.

Beginning in privileged EXEC mode, follow these steps to configure passive interfaces:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router** {**bgp** |**rip** | **ospf** | **eigrp**}	Enter router configuration mode.
Step 3	**passive-interface** *interface-id*	Suppress sending routing updates through the specified Layer 3 interface.
Step 4	**passive-interface default**	(Optional) Set all interfaces as passive by default.
Step 5	**no passive-interface** *interface type*	(Optional) Activate only those interfaces that need to have adjacencies sent.
Step 6	**network** *network-address*	(Optional) Specify the list of networks for the routing process. The *network-address* is an IP address.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use a network monitoring privileged EXEC command such as **show ip ospf interface** to verify the interfaces that you enabled as passive, or use the **show ip interface** privileged EXEC command to verify the interfaces that you enabled as active.

To re-enable the sending of routing updates, use the **no passive-interface** *interface-id* router configuration command. The **default** keyword sets all interfaces as passive by default. You can then configure individual interfaces where you want adjacencies by using the **no passive-interface** router configuration command. The **default** keyword is useful in Internet service provider and large enterprise networks where many of the distribution routers have more than 200 interfaces.

Controlling Advertising and Processing in Routing Updates

You can use the **distribute-list** router configuration command with access control lists to suppress routes from being advertised in routing updates and to prevent other routers from learning one or more routes. When used in OSPF, this feature applies to only external routes, and you cannot specify an interface name.

You can also use a **distribute-list** router configuration command to avoid processing certain routes listed in incoming updates. (This feature does not apply to OSPF.)

Beginning in privileged EXEC mode, follow these steps to control the advertising or processing of routing updates:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router** {**bgp** |**rip** | **ospf** | **eigrp**}	Enter router configuration mode.
Step 3	**distribute-list** {*access-list-number* | *access-list-name*} **out** [*interface-name* | *routing process* | *autonomous-system-number*]	Permit or deny routes from being advertised in routing updates, depending upon the action listed in the access list.
Step 4	**distribute-list** {*access-list-number* | *access-list-name*} **in** [*type-number*]	Suppress processing in routes listed in updates.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no distribute-list in** router configuration command to change or cancel a filter. To cancel suppression of network advertisements in updates, use the **no distribute-list out** router configuration command.

Filtering Sources of Routing Information

Because some routing information might be more accurate than others, you can use filtering to prioritize information coming from different sources. An *administrative distance* is a rating of the trustworthiness of a routing information source, such as a router or group of routers. In a large network, some routing protocols can be more reliable than others. By specifying administrative distance values, you enable the router to intelligently discriminate between sources of routing information. The router always picks the route whose routing protocol has the lowest administrative distance. Table 31-15 on page 31-74 shows the default administrative distances for various routing information sources.

Because each network has its own requirements, there are no general guidelines for assigning administrative distances.

Beginning in privileged EXEC mode, follow these steps to filter sources of routing information:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**router** {**bgp** |**rip** | **ospf** | **eigrp**}	Enter router configuration mode.
Step 3	**distance** *weight* {*ip-address* {*ip-address mask*}} [*ip access list*]	Define an administrative distance. *weight*—Administrative distance as an integer from 10 to 255. Used alone, *weight* specifies a default administrative distance used when no other specification exists for a routing information source. Routes with a distance of 255 are not installed in the routing table. (Optional) *ip access list*—IP standard or extended access list to be applied to incoming routing updates.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip protocols**	Display the default administrative distance for a specified routing process.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a distance definition, use the **no distance** router configuration command.

Managing Authentication Keys

Key management is a method of controlling authentication keys used by routing protocols. Not all protocols can use key management. Authentication keys are available for EIGRP and RIP Version 2.

Before you manage authentication keys, you must enable authentication. See the appropriate protocol section to see how to enable authentication for that protocol. To manage authentication keys, define a key chain, identify the keys that belong to the key chain, and specify how long each key is valid. Each key has its own key identifier (specified with the **key** *number* key chain configuration command), which is stored locally. The combination of the key identifier and the interface associated with the message uniquely identifies the authentication algorithm and Message Digest 5 (MD5) authentication key in use.

You can configure multiple keys with lifetimes. Only one authentication packet is sent, regardless of how many valid keys exist. The software examines the key numbers in order from lowest to highest and uses the first valid key it encounters. The lifetimes allow for overlap during key changes. Note that the router must know these lifetimes.

Beginning in privileged EXEC mode, follow these steps to manage authentication keys:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**key chain** *name-of-chain*	Identify a key chain, and enter key chain configuration mode.
Step 3	**key** *number*	Identify the key number. The range is 0 to 2147483647.
Step 4	**key-string** *text*	Identify the key string. The string can contain from 1 to 80 uppercase and lowercase alphanumeric characters, but the first character cannot be a number.

	Command	Purpose
Step 5	**accept-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*}	(Optional) Specify the time period during which the key can be received. The *start-time* and *end-time* syntax can be either *hh*:*mm*:*ss Month date year* or *hh*:*mm*:*ss date Month year*. The default is forever with the default *start-time* and the earliest acceptable date as January 1, 1993. The default *end-time* and **duration** is **infinite**.
Step 6	**send-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*}	(Optional) Specify the time period during which the key can be sent. The *start-time* and *end-time* syntax can be either *hh*:*mm*:*ss Month date year* or *hh*:*mm*:*ss date Month year*. The default is forever with the default *start-time* and the earliest acceptable date as January 1, 1993. The default *end-time* and **duration** is **infinite**.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show key chain**	Display authentication key information.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the key chain, use the **no key chain** *name-of-chain* global configuration command.

Monitoring and Maintaining the IP Network

You can remove all contents of a particular cache, table, or database. You can also display specific statistics. Use the privileged EXEC commands in Table 31-16 to clear routes or display status:

Table 31-16 Commands to Clear IP Routes or Display Route Status

Command	Purpose
clear ip route {*network* [*mask* | *]}	Clear one or more routes from the IP routing table.
show ip protocols	Display the parameters and state of the active routing protocol process.
show ip route [*address* [*mask*] [**longer-prefixes**]] | [*protocol* [*process-id*]]	Display the current state of the routing table.
show ip route summary	Display the current state of the routing table in summary form.
show ip route supernets-only	Display supernets.
show ip cache	Display the routing table used to switch IP traffic.
show route-map [*map-name*]	Display all route maps configured or only the one specified.

CHAPTER 32

Configuring HSRP

This chapter describes how to use Hot Standby Router Protocol (HSRP) on your Catalyst 3550 switch to provide routing redundancy for routing IP traffic without being dependent on the availability of any single router.

Note You can also use a version of HSRP in Layer 2 mode to configure a redundant command switch to take over cluster management if the cluster command switch fails. For more information about clustering, see Chapter 5, "Clustering Switches" and see *Getting Started with Cisco Network Assistant*, available on Cisco.com.

Note For complete syntax and usage information for the commands used in this chapter, see the switch command reference for this release and the *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2*.

This chapter consists of these sections:

- Understanding HSRP, page 32-1
- Configuring HSRP, page 32-4
- Displaying HSRP Configurations, page 32-10

Understanding HSRP

HSRP is Cisco's standard method of providing high network availability by providing first-hop redundancy for IP hosts on an IEEE 802 LAN configured with a default gateway IP address. HSRP routes IP traffic without relying on the availability of any single router. It enables a set of router interfaces to work together to present the appearance of a single virtual router or default gateway to the hosts on a LAN. When HSRP is configured on a network or segment, it provides a virtual Media Access Control (MAC) address and an IP address that is shared among a group of configured routers. HSRP allows two or more HSRP-configured routers to use the MAC address and IP network address of a virtual router. The virtual router does not exist; it represents the common target for routers that are configured to provide backup to each other. One of the routers is selected to be the active router and another to be the standby router, which assumes control of the group MAC address and IP address should the designated active router fail.

Note Routers in an HSRP group can be any router interface that supports HSRP, including Catalyst 3550 routed ports and switch virtual interfaces (SVIs).

HSRP provides high network availability by providing redundancy for IP traffic from hosts on networks. In a group of router interfaces, the active router is the router of choice for routing packets. The standby router is the router that takes over the routing duties when an active router fails or when preset conditions are met.

The **standby ip** interface configuration command activates HSRP on a Layer 3 interface. If an IP address is specified, that address is used as the designated address for the Hot Standby group. If no IP address is specified, the address is learned through the standby function.

HSRP is useful for hosts that do not support a router discovery protocol and that cannot switch to a new router when the selected router reloads or loses power. When HSRP is configured on a network segment, it provides a virtual MAC address and IP address that is shared among grouped router interfaces that are running HSRP. The router selected by the protocol to be the active router receives and routes packets destined for the group MAC address. For *n* routers running HSRP, there are *n +1* IP and MAC addresses assigned.

HSRP detects when the designated active router fails, and a selected standby router assumes control of the Hot Standby group MAC and IP addresses. A new standby router is also selected at that time. Devices running HSRP send and receive multicast UDP-based hello packets to detect router failure and to designate active and standby routers. When HSRP is configured on an interface, Internet Control Message Protocol (ICMP) redirect messages are disabled by default for the interface.

The switch supports HSRP MAC addresses for up to 16 unique HSRP groups. Because each group address can be used on up to 16 Layer 3 interfaces, the maximum number of HSRP interfaces is 256. However, the relationship between the number of HSRP interfaces and the number of active IP routing protocols and other configured features might have an impact on CPU utilization. Because of other switch feature configurations, we recommend that you do not assign more than 64 HSRP interfaces. The switch returns an error message after a period of up to 1 minute if you exceed the HSRP MAC address limitation of 256.

Each of the 16 HSRP MAC addresses can be used by 16 consecutive Layer 3 interfaces because each address is associated with a group of VLANs by using a 4-bit mask. The mask requires that all Layer 3 interfaces be the same multiple of 16. When you create an HSRP group, you can use the same HSRP MAC address on a single Layer 3 interface, several Layer 3 interfaces that are all the same multiple of 16, or a consecutive range of 16 Layer 3 interfaces that are all the same multiple of 16.

For example, HSRP Group 1 might be assigned to interface VLANs 16 to 31, which equals the group maximum of 16 VLANs. VLAN IDs between 16 and 31 are all the same multiple of 16 (1, or 1 plus some small amount.) Therefore, if Group 1 is assigned to interface VLANs 16 to 31, only one HSRP MAC address entry is used in hardware. If Group 1 is also assigned to interface VLAN 32, an additional MAC address entry is used in hardware because it is not in the same multiple of 16 as VLANs 16 to 31.

Instead, if Group 1 is assigned to interface VLAN 16, and Group 2 is assigned to interface VLAN 17, two HSRP MAC address entries are used in hardware. Group 1 uses one MAC address entry, and Group 2 uses the other MAC address entry. If Group 1 or Group 2 are later configured on interfaces VLAN 18 through VLAN 31, an additional HSRP MAC address entry is not used because the MAC address entries for these two groups have already been created and can be used for all VLAN interfaces between 16 and 31.

The SVI VLAN ID number is the same as the interface VLAN ID number (for example, *interface Vlan 16* uses VLAN 16). For routed ports, the switch automatically assigns a VLAN ID to the interface. Assigned numbers begin at the first available VLAN above 1024. These assigned numbers are also limited to the range of 16 consecutive VLANs per group.

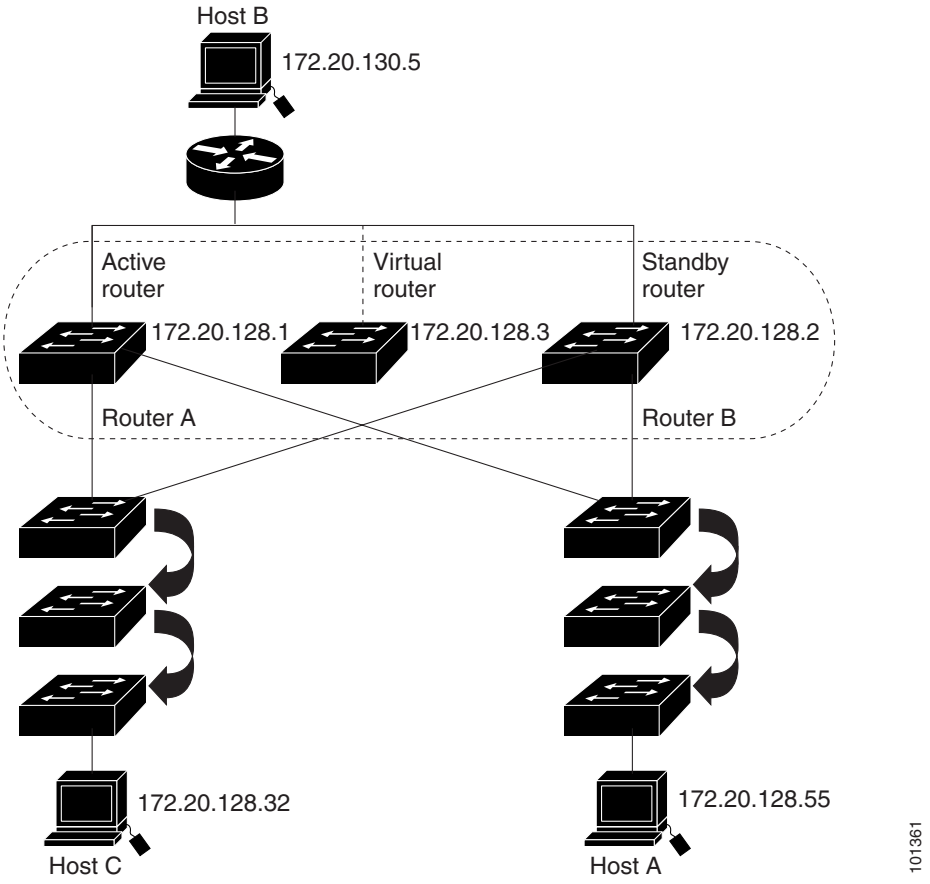
You can verify the VLAN ID assigned to a routed port by using the **show vlan internal usage** privileged EXEC command.

An interface can belong to multiple HSRP groups, and the same HSRP group can be applied to different interfaces.

Note Identically-numbered HSRP groups use the same virtual MAC address and might cause errors if you configure bridge groups.

Figure 32-1 shows a segment of a network configured for HSRP. Each router is configured with the MAC address and IP network address of the virtual router. Instead of configuring hosts on the network with the IP address of Router A, you configure them with the IP address of the virtual router as their default router. When Host C sends packets to Host B, it sends them to the MAC address of the virtual router. If for any reason, Router A stops transferring packets, Router B responds to the virtual IP address and virtual MAC address and becomes the active router, assuming the active router duties. Host C continues to use the IP address of the virtual router to address packets destined for Host B, which Router B now receives and sends to Host B. Until Router A resumes operation, HSRP allows Router B to provide uninterrupted service to users on Host C's segment that need to communicate with users on Host B's segment and also continues to perform its normal function of handling packets between the Host A segment and Host B.

Figure 32-1 Typical HSRP Configuration



Configuring HSRP

These sections include HSRP configuration information:

- Default HSRP Configuration, page 32-4
- HSRP Configuration Guidelines and Limitations, page 32-4
- Enabling HSRP, page 32-5
- Configuring HSRP Priority, page 32-6
- Configuring HSRP Authentication and Timers, page 32-8
- Configuring HSRP Groups and Clustering, page 32-10

Default HSRP Configuration

Table 32-1 shows the default HSRP configuration.

Table 32-1 Default HSRP Configuration

Feature	Default Setting
HSRP groups	None configured
Standby group number	0
Standby MAC address	System assigned as: 0000.0c07.acXX, where *XX* is the HSRP group number
Standby priority	100
Standby delay	0 (no delay)
Standby track interface priority	10
Standby hello time	3 seconds
Standby holdtime	10 seconds

HSRP Configuration Guidelines and Limitations

Follow these guidelines when configuring HSRP:

- In the following procedures, the specified interface must be one of these Layer 3 interfaces:
  - Routed port: a physical port configured as a Layer 3 port by entering the **no switchport** interface configuration command.
  - SVI: a VLAN interface created by using the **interface vlan** *vlan_id* global configuration command and by default a Layer 3 interface.
  - Etherchannel port channel in Layer 3 mode: a port-channel logical interface created by using the **interface port-channel** *port-channel-number* global configuration command and binding the Ethernet interface into the channel group. For more information, see the "Configuring Layer 3 EtherChannels" section on page 30-12.
- All Layer 3 interfaces must have IP addresses assigned to them. See the "Configuring Layer 3 Interfaces" section on page 9-19.

- The switch supports HSRP MAC address entries in hardware for up to 16 unique HSRP groups. Because of other switch feature configurations, we recommend that you do not assign more than 64 HSRP interfaces.
- An HSRP group can use the same HSRP MAC address on a single Layer 3 interface, several Layer 3 interfaces that are all the same multiple of 16, or a consecutive range of 16 Layer 3 interfaces that are all the same multiple of 16. For more information about HSRP groups, see the "Understanding HSRP" section on page 32-1.
- An interface can belong to multiple HSRP groups, and the same HSRP group can be applied to different interfaces.
- If you configure the same HSRP group on multiple VLANs, do not use bridge groups to tie the multiple interfaces together. Identically-numbered HSRP groups use the same virtual MAC address and might cause errors if you configure bridge groups.

Enabling HSRP

The **standby ip** interface configuration command activates HSRP on the configured interface. If an IP address is specified, that address is used as the designated address for the Hot Standby group. If no IP address is specified, the address is learned through the standby function. You must configure at least one routing port on the cable with the designated address. Configuring an IP address always overrides another designated address currently in use.

When the **standby ip** command is enabled on an interface and proxy ARP is enabled, if the interface's Hot Standby state is active, proxy ARP requests are answered using the Hot Standby group MAC address. If the interface is in a different state, proxy ARP responses are suppressed.

Note When multi-VRF CE is configured, you cannot assign the same HSRP standby address to two different VPNs.

Beginning in privileged EXEC mode, follow these steps to create or enable HSRP on a Layer 3 interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the Layer 3 interface on which you want to enable HSRP.

	Command	Purpose
Step 3	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]]	Create (or enable) the HSRP group using its number and virtual IP address. • (Optional) *group-number*—The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) *ip-address*—The virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. • (Optional) **secondary**—The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show standby** [*interface-id* [*group*]]	Verify the configuration.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no standby** [*group-number*] **ip** [*ip-address*] interface configuration command to disable HSRP.

This example shows how to activate HSRP for group 1 on Gigabit Ethernet interface 0/1. The IP address used by the hot standby group is learned by using HSRP.

Note This procedure is the minimum number of steps required to enable HSRP.

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# standby 1 ip
Switch(config-if)# end
Switch# show standby
```

Configuring HSRP Priority

The **standby priority**, **standby preempt**, and **standby track** interface configuration commands are all used to set characteristics for determining active and standby routers and behavior regarding when a new active router takes over. When configuring priority, follow these guidelines:

- Assigning priority helps select the active and standby routers. If preemption is enabled, the router with the highest priority becomes the designated active router. If priorities are equal, the primary IP addresses are compared, and the higher IP address has priority.
- The highest number (1 to 255) represents the highest priority (most likely to become the active router).
- When setting the priority, preempt, or both, you must specify at least one keyword (**priority**, **preempt**, or both).
- The priority of the device can change dynamically if an interface is configured with the **standby track** command and another interface on the router goes down.

- The **standby track** interface configuration command ties the router hot standby priority to the availability of its interfaces and is useful for tracking interfaces that are not configured for HSRP. When a tracked interface fails, the hot standby priority on the device on which tracking has been configured decreases by 10. If an interface is not tracked, its state changes do not affect the hot standby priority of the configured device. For each interface configured for hot standby, you can configure a separate list of interfaces to be tracked.
- The **standby track** *interface-priority* interface configuration command specifies how much to decrement the hot standby priority when a tracked interface goes down. When the interface comes back up, the priority is incremented by the same amount.
- When multiple tracked interfaces are down and *interface-priority* values have been configured, the configured priority decrements are cumulative. If tracked interfaces that were not configured with priority values fail, the default decrement is 10, and it is noncumulative.
- When routing is first enabled for the interface, it does not have a complete routing table. If it is configured to preempt, it becomes the active router, even though it is unable to provide adequate routing services. To solve this problem, configure a delay time to allow the router to update its routing table.

Beginning in privileged EXEC mode, use one or more of these steps to configure HSRP priority characteristics on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the HSRP interface on which you want to set priority.
Step 3	**standby** [*group-number*] **priority** *priority* [**preempt** [**delay** *delay*]]	Set a **priority** value used in choosing the active router. The range is 1 to 255; the default priority is 100. The highest number represents the highest priority. • (Optional) *group-number*—The group number to which the command applies. • (Optional) **preempt—**Select so that when the local router has a higher priority than the active router, it assumes control as the active router. • (Optional) **delay**—Set to cause the local router to postpone taking over the active role for the shown number of seconds. The range is 0 to 3600 (1 hour); the default is 0 (no delay before taking over). Use the **no** form of the command to restore the default values.
Step 4	**standby** [*group-number*] [**priority** *priority*] **preempt** [**delay** *delay*]	Configure the router to **preempt**, which means that when the local router has a higher priority than the active router, it assumes control as the active router. • (Optional) *group-number*—The group number to which the command applies. • (Optional) **priority**—Enter to set or change the group priority. The range is 1 to 255; the default is 100. • (Optional) **delay**—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 36000 (1 hour); the default is 0 (no delay before taking over). Use the **no** form of the command to restore the default values.

	Command	Purpose
Step 5	**standby** [*group-number*] **track** *type number* [*interface-priority*]	Configure an interface to track other interfaces so that if one of the other interfaces goes down, the device's Hot Standby priority is lowered. • (Optional) *group-number*—The group number to which the command applies. • *type*—Enter the interface type (combined with interface number) that is tracked. • *number*—Enter the interface number (combined with interface type) that is tracked. • (Optional) *interface-priority*—Enter the amount by which the hot standby priority for the router is decremented or incremented when the interface goes down or comes back up. The default value is 10.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify the configuration of the standby groups.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no standby** [*group-number*] **priority** *priority* [**preempt** [**delay** *delay*]] and **no standby** [*group-number*] [**priority** *priority*] **preempt** [**delay** *delay*] interface configuration commands to restore default priority, preempt, and delay values.

Use the **no standby** [*group-number*] **track** *type number* [*interface-priority*] interface configuration command to remove the tracking.

This example shows how to activate an interface as a standby router, set an IP address and a priority of 120 (higher than the default value), and a delay time of 300 seconds (5 minutes) before the router attempts to become the active router:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# standby ip 172.20.128.3
Switch(config-if)# standby priority 120 preempt delay 300
Switch(config-if)# end
```

Configuring HSRP Authentication and Timers

You can optionally configure an HSRP authentication string or change the hello-time interval and holdtime.

When configuring these attributes, follow these guidelines:

- The authentication string is sent unencrypted in all HSRP messages. You must configure the same authentication string on all routers and access servers on a cable to ensure interoperation. Authentication mismatch prevents a device from learning the designated Hot Standby IP address and timer values from other routers configured with HSRP.
- Routers or access servers on which standby timer values are not configured can learn timer values from the active or standby router. The timers configured on an active router always override any other timer settings.
- All routers in a Hot Standby group should use the same timer values. Normally, the *holdtime* is greater than or equal to 3 times the *hellotime*.

Beginning in privileged EXEC mode, use one or more of these steps to configure HSRP authentication and timers on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the HSRP interface on which you want to set authentication.
Step 3	**standby** [*group-number*] **authentication** *string*	(Optional) **authentication** *string*—Enter a string to be carried in all HSRP messages. The authentication string can be up to eight characters in length; the default string is **cisco.** (Optional) *group-number*—The group number to which the command applies.
Step 4	**standby** [*group-number*] **timers** *hellotime holdtime*	(Optional) Configure the time between hello packets and the time before other routers declare the active router to be down. • *group-number*—The group number to which the command applies. • *hellotime*—The hello interval in seconds. The range is from 1 to 255; the default is 3 seconds. • *holdtime*—The time in seconds before the active or standby router is declared to be down. The range is from 1 to 255; the default is 10 seconds.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify the configuration of the standby groups.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no standby** [*group-number*] **authentication** *string* interface configuration command to delete an authentication string. Use the **no standby** [*group-number*] **timers** *hellotime holdtime* interface configuration command to restore timers to their default values.

This example shows how to configure *word* as the authentication string required to allow Hot Standby routers in group 1 to interoperate:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# standby 1 authentication word
Switch(config-if)# end
```

This example shows how to set the timers on standby group 1 with the time between hello packets at 5 seconds and the time after which a router is considered down to be 15 seconds:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# standby 1 ip
Switch(config-if)# standby 1 timers 5 15
Switch(config-if)# end
```

Configuring HSRP Groups and Clustering

When a device is participating in an HSRP standby routing and clustering is enabled, you can use the same standby group for command switch redundancy and HSRP redundancy. Use the **cluster standby-group** *HSRP-group-name* [**routing-redundancy**] global configuration command to enable the same HSRP standby group to be used for command switch and routing redundancy. If you create a cluster with the same HSRP standby group name without entering the **routing-redundancy** keyword, HSRP standby routing is disabled for the group.

This example shows how to bind standby group my_hsrp to the cluster and enable the same HSRP group to be used for command switch redundancy and router redundancy. The command can only be executed on the command switch. If the standby group name or number does not exist, or if the switch is a member switch, an error message appears.

```
Switch# configure terminal
Switch(config)# cluster standby-group my_hsrp routing-redundancy
Switch(config)# end
```

Displaying HSRP Configurations

From privileged EXEC mode, use this command to display HSRP settings:

show standby [*interface-id* [*group*]] [**brief**] [**detail**]

You can display HSRP information for the whole switch, for a specific interface, for an HSRP group, or for an HSRP group on an interface. You can also specify whether to display a concise overview of HSRP information or detailed HSRP information. The default display is **detail**. If there are a large number of HSRP groups, using the **show standby** command without qualifiers can result in an unwieldy display.

This is a an example of output from the **show standby** privileged EXEC command, displaying HSRP information for two standby groups (group 1 and group 100):

```
Switch# show standby
VLAN1 - Group 1
   Local state is Standby, priority 105, may preempt
   Hellotime 3 holdtime 10
   Next hello sent in 00:00:02.182
   Hot standby IP address is 172.20.128.3 configured
   Active router is 172.20.128.1 expires in 00:00:09
   Standby router is local
   Standby virtual mac address is 0000.0c07.ac01
   Name is bbb
VLAN1 - Group 100
   Local state is Active, priority 105, may preempt
   Hellotime 3 holdtime 10
   Next hello sent in 00:00:02.262
   Hot standby IP address is 172.20.138.51 configured
   Active router is local
   Standby router is unknown expired
   Standby virtual mac address is 0000.0c07.ac64
   Name is test
```

CHAPTER 1

Configuring Cisco IOS IP SLAs Operations

This chapter describes how to use Cisco IOS IP Service Level Agreements (SLAs) on your Catalyst 3550 switch. Cisco IP SLAs is a part of Cisco IOS software that allows Cisco customers to analyze IP service levels for IP applications and services by using active traffic monitoring—the generation of traffic in a continuous, reliable, and predictable manner—for measuring network performance. With Cisco IOS IP SLAs, service provider customers can measure and provide service level agreements, and enterprise customers can verify service levels, verify outsourced service level agreements, and understand network performance. Cisco IOS IP SLAs can perform network assessments, verify quality of service (QoS), ease the deployment of new services, and assist with network troubleshooting.

Note For more information about IP SLAs, see the *Cisco IOS IP SLAs Configuration Guide, Release 12.4T* at this URL:
http://www.cisco.com/en/US/products/ps6441/products_configuration_guide_book09186a0080707055.html
For command syntax information, see the command reference at this URL:
http://www.cisco.com/en/US/products/ps6441/products_command_reference_book09186a008049739b.html

This chapter consists of these sections:

- Understanding Cisco IOS IP SLAs, page 1-1
- Configuring IP SLAs Operations, page 1-6
- Monitoring IP SLAs Operations, page 1-13

Understanding Cisco IOS IP SLAs

Cisco IOS IP SLAs sends data across the network to measure performance between multiple network locations or across multiple network paths. It simulates network data and IP services and collects network performance information in real time. Cisco IOS IP SLAs generates and analyzes traffic either between Cisco IOS devices or from a Cisco IOS device to a remote IP device such as a network application server. Measurements provided by the various Cisco IOS IP SLAs operations can be used for troubleshooting, for problem analysis, and for designing network topologies.

Depending on the specific Cisco IOS IP SLAs operation, various network performance statistics are monitored within the Cisco device and stored in both command-line interface (CLI) and Simple Network Management Protocol (SNMP) MIBs. IP SLAs packets have configurable IP and application layer

options such as source and destination IP address, User Datagram Protocol (UDP)/TCP port numbers, a type of service (ToS) byte (including Differentiated Services Code Point [DSCP] and IP Prefix bits), Virtual Private Network (VPN) routing/forwarding instance (VRF), and URL web address.

Because Cisco IP SLAs is Layer 2 transport independent, you can configure end-to-end operations over disparate networks to best reflect the metrics that an end user is likely to experience. IP SLAs collects a unique subset of these performance metrics:

- Delay (both round-trip and one-way)
- Jitter (directional)
- Packet loss (directional)
- Packet sequencing (packet ordering)
- Path (per hop)
- Connectivity (directional)
- Server or website download time

Because Cisco IOS IP SLAs is SNMP-accessible, it can also be used by performance-monitoring applications like CiscoWorks Internetwork Performance Monitor (IPM) and other third-party Cisco partner performance management products. You can find more details about network management products that use Cisco IOS IP SLAs at this URL:

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

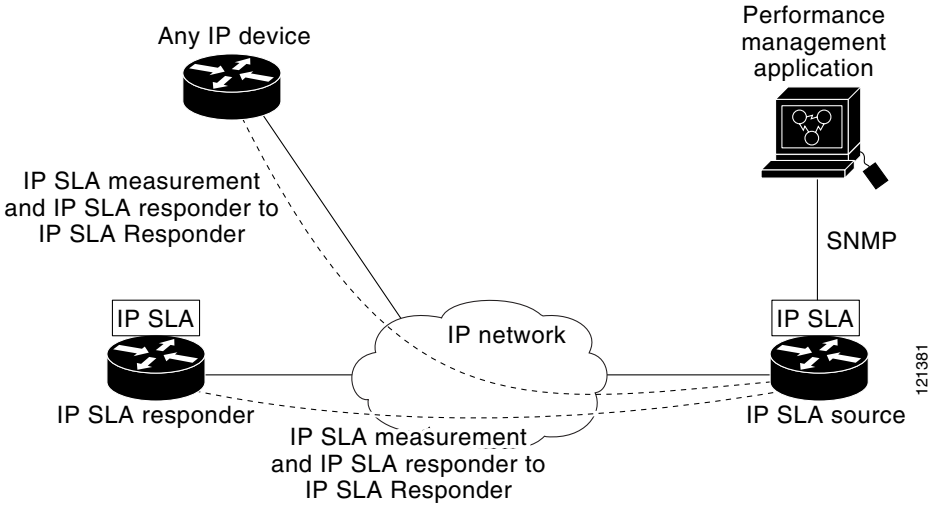
Using IP SLAs can provide these benefits:

- Service-level agreement monitoring, measurement, and verification.
- Network performance monitoring
  - Measures the jitter, latency, or packet loss in the network.
  - Provides continuous, reliable, and predictable measurements.
- IP service network health assessment to verify that the existing QoS is sufficient for new IP services.
- Edge-to-edge network availability monitoring for proactive verification and connectivity testing of network resources (for example, shows the network availability of an NFS server used to store business critical data from a remote site).
- Troubleshooting of network operation by providing consistent, reliable measurement that immediately identifies problems and saves troubleshooting time.
- Multiprotocol Label Switching (MPLS) performance monitoring and network verification (if the switch supports MPLS)

Using Cisco IOS IP SLAs to Measure Network Performance

You can use IP SLAs to monitor the performance between any area in the network—core, distribution, and edge—without deploying a physical probe. It uses generated traffic to measure network performance between two networking devices. Figure 1-1 shows how IP SLAs begins when the source device sends a generated packet to the destination device. After the destination device receives the packet, depending on the type of IP SLAs operation, it responds with time-stamp information for the source to make the calculation on performance metrics. An IP SLAs operation performs a network measurement from the source device to a destination in the network using a specific protocol such as UDP.

Figure 1-1 Cisco IOS IP SLAs Operation



To implement IP SLAs network performance measurement, you need to perform these tasks:

1. Enable the IP SLAs responder, if required.
2. Configure the required IP SLAs operation type.
3. Configure any options available for the specified operation type.
4. Configure threshold conditions, if required.
5. Schedule the operation to run, then let the operation run for a period of time to gather statistics.
6. Display and interpret the results of the operation using the Cisco IOS CLI or a network management system (NMS) system with SNMP.

For more information about IP SLAs operations, see the operation-specific chapters in the *Cisco IOS IP SLAs Configuration Guide* at this URL:
http://www.cisco.com/en/US/products/ps6441/products_configuration_guide_book09186a0080707055.html

Note The switch does not support Voice over IP (VoIP) service level analysis or IP service level analysis using DLSw+ operation. Before configuring any IP SLAs application, you can use the **show ip sla application** privileged EXEC command to verify that the operation type is supported on your software image.

IP SLAs Responder and IP SLAs Control Protocol

The IP SLAs responder is a component embedded in the destination Cisco device that allows the system to anticipate and respond to IP SLAs request packets. The responder provides accurate measurements without the need for dedicated probes. The responder uses the Cisco IOS IP SLAs Control Protocol to provide a mechanism through which it can be notified on which port it should listen and respond. Only a Cisco IOS device can be a source for a destination IP SLAs Responder.

Note The IP SLAs responder can be a Cisco IOS Layer 2, responder-configurable switch, such as a Catalyst 2960 or Cisco ME 2400 switch. The responder does not need to support full IP SLAs functionality.

Figure 1-1 shows where the Cisco IOS IP SLAs responder fits in the IP network. The responder listens on a specific port for control protocol messages sent by an IP SLAs operation. Upon receipt of the control message, it enables the specified UDP or TCP port for the specified duration. During this time, the responder accepts the requests and responds to them. It disables the port after it responds to the IP SLAs packet, or when the specified time expires. MD5 authentication for control messages is available for added security.

You do not need to enable the responder on the destination device for all IP SLAs operations. For example, a responder is not required for services that are already provided by the destination router (such as Telnet or HTTP). You cannot configure the IP SLAs responder on non-Cisco devices and Cisco IOS IP SLAs can send operational packets only to services native to those devices.

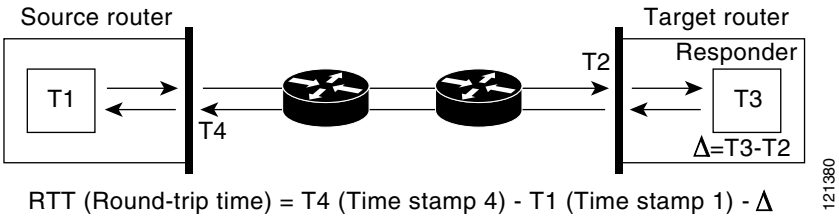
Response Time Computation for IP SLAs

Switches and routers can take tens of milliseconds to process incoming packets due to other high priority processes. This delay affects the response times because the test-packet reply might be in a queue while waiting to be processed. In this situation, the response times would not accurately represent true network delays. IP SLAs minimizes these processing delays on the source device as well as on the target device (if the responder is being used) to determine true round-trip times. IP SLAs test packets use time stamping to minimize the processing delays.

When the IP SLAs responder is enabled, it allows the target device to take time stamps when the packet arrives on the interface at interrupt level and again just as it is leaving, eliminating the processing time. This time stamping is made with a granularity of sub-milliseconds (ms).

Figure 1-2 demonstrates how the responder works. Four time stamps are taken to make the calculation for round-trip time. At the target router, with the responder functionality enabled, time stamp 2 (TS2) is subtracted from time stamp 3 (TS3) to produce the time spent processing the test packet as represented by delta. This delta value is then subtracted from the overall round-trip time. Notice that the same principle is applied by IP SLAs on the source router where the incoming time stamp 4 (TS4) is also taken at the interrupt level to allow for greater accuracy.

Figure 1-2 Cisco IOS IP SLAs Responder Time Stamping



An additional benefit of the two time stamps at the target device is the ability to track one-way delay, jitter, and directional packet loss. Because much network behavior is asynchronous, it is critical to have these statistics. However, to capture one-way delay measurements, you must configure both the source router and target router with Network Time Protocol (NTP) so that the source and target are synchronized to the same clock source. One-way jitter measurements do not require clock synchronization.

IP SLAs Operation Scheduling

When you configure an IP SLAs operation, you must schedule the operation to begin capturing statistics and collecting error information. You can schedule an operation to start immediately or to start at a certain month, day, and hour. You can use the *pending* option to set the operation to start at a later time. The pending option is an internal state of the operation that is visible through SNMP. The pending state is also used when an operation is a reaction (threshold) operation waiting to be triggered. You can schedule a single IP SLAs operation or a group of operations at one time.

You can schedule several IP SLAs operations by using a single command through the Cisco IOS CLI or the CISCO RTTMON-MIB. Scheduling the operations to run at evenly distributed times allows you to control the amount of IP SLAs monitoring traffic. This distribution of IP SLAs operations helps minimize the CPU utilization and thus improves network scalability.

For more details about the IP SLAs multioperations scheduling functionality, see the "IP SLAs—Multiple Operation Scheduling" chapter of the *Cisco IOS IP SLAs Configuration Guide* at this URL:
http://www.cisco.com/en/US/products/ps6441/products_configuration_guide_book09186a0080707055.html

IP SLAs Operation Threshold Monitoring

To support successful service level agreement monitoring, you must have mechanisms that notify you immediately of any possible violation. IP SLAs can send SNMP traps that are triggered by events such as these:

- Connection loss
- Timeout
- Round-trip time threshold
- Average jitter threshold
- One-way packet loss
- One-way jitter
- One-way mean opinion score (MOS)
- One-way latency

An IP SLAs threshold violation can also trigger another IP SLAs operation for further analysis. For example, the frequency could be increased or an ICMP path echo or ICMP path jitter operation could be initiated for troubleshooting.

Determining the type of threshold and the level to set can be complex, and depends on the type of IP service being used in the network. For more details on using thresholds with Cisco IOS IP SLAs operations, see the "IP SLAs—Proactive Threshold Monitoring" chapter of the Cisco IOS IP SLAs Configuration Guide at this URL:
http://www.cisco.com/en/US/products/ps6441/products_configuration_guide_book09186a0080707055.html

Configuring IP SLAs Operations

This section does not include configuration information for all available operations as the configuration information details are included in the *Cisco IOS IP SLAs Configuration Guide*. It does include several operations as examples, including configuring the responder, configuring UDP jitter operation, which requires a responder, and configuring ICMP echo operation, which does not require a responder. For details about configuring other operations, see he *Cisco IOS IP SLAs Configuration Guide* at this URL: http://www.cisco.com/en/US/products/ps6441/products_configuration_guide_book09186a0080707055.html

This section includes this information:

- Default Configuration, page 1-6
- Configuration Guidelines, page 1-6
- Configuring the IP SLAs Responder, page 1-7
- Analyzing IP Service Levels by Using the UDP Jitter Operation, page 1-8
- Analyzing IP Service Levels by Using the ICMP Echo Operation, page 1-10

Default Configuration

No IP SLAs operations are configured.

Configuration Guidelines

For information on the IP SLAs commands, see the *Cisco IOS IP SLAs Command Reference, Release 12.4T* command reference at this URL:
http://www.cisco.com/en/US/products/ps6441/products_command_reference_book09186a008049739b.html

For detailed descriptions and configuration procedures, see the *Cisco IOS IP SLAs Configuration Guide, Release 12.4T* at this URL:
http://www.cisco.com/en/US/products/ps6441/products_configuration_guide_book09186a0080707055.html

Note that not all of the IP SLAs commands or operations described in this guide are supported on the switch. The switch supports IP service level analysis by using UDP jitter, UDP echo, HTTP, TCP connect, ICMP echo, ICMP path echo, ICMP path jitter, FTP, DNS, and DHCP, as well as multiple operation scheduling and proactive threshold monitoring. It does not support VoIP service analysis or IP service level analysis by using DLSw+ operation.

Before configuring any IP SLAs application, you can use the **show ip sla application** privileged EXEC command to verify that the operation type is supported on your software image. This is an example of the output from the command:

```
Switch# show ip sla application
        IP SLAs
Version: 2.2.0 Round Trip Time MIB, Infrastructure Engine-II
Time of last change in whole IP SLAs: 22:17:39.117 UTC Fri Jun
Estimated system max number of entries: 15801

Estimated number of configurable operations: 15801
Number of Entries configured  : 0
Number of active Entries      : 0
```

```
Number of pending Entries     : 0
Number of inactive Entries    : 0

        Supported Operation Types
Type of Operation to Perform: 802.1agEcho
Type of Operation to Perform: 802.1agJitter
Type of Operation to Perform: dhcp
Type of Operation to Perform: dns
Type of Operation to Perform: echo
Type of Operation to Perform: ftp
Type of Operation to Perform: http
Type of Operation to Perform: jitter
Type of Operation to Perform: pathEcho
Type of Operation to Perform: pathJitter
Type of Operation to Perform: tcpConnect
Type of Operation to Perform: udpEcho

IP SLAs low memory water mark: 21741224
```

Configuring the IP SLAs Responder

The IP SLAs responder is available only on Cisco IOS software-based devices, including some Layer 2 switches that do not support full IP SLAs functionality, such as the Catalyst 2960 or the Cisco ME 2400 switch. Beginning in privileged EXEC mode, follow these steps to configure the IP SLAs responder on the target device (the operational target):

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip sla responder** {**tcp-connect** | **udp-echo**} **ipaddress** *ip-address* **port** *port-number*	Configure the switch as an IP SLAs responder. The keywords have these meanings: • **tcp-connect**—Enable the responder for TCP connect operations. • **udp-echo**—Enable the responder for User Datagram Protocol (UDP) echo or jitter operations. • **ipaddress** *ip-address*—Enter the destination IP address. • **port** *port-number*—Enter the destination port number. **Note** The IP address and port number must match those configured on the source device for the IP SLAs operation.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show ip sla responder**	Verify the IP SLAs responder configuration on the device.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the IP SLAs responder, enter the **no ip sla responder** global configuration command. This example shows how to configure the device as a responder for the UDP jitter IP SLAs operation in the next procedure:

```
Switch(config)# ip sla responder udp-echo 172.29.139.134 5000
```

Analyzing IP Service Levels by Using the UDP Jitter Operation

Jitter means interpacket delay variance. When multiple packets are sent consecutively 10 ms apart from source to destination, if the network is behaving correctly, the destination should receive them 10 ms apart. But if there are delays in the network (like queuing, arriving through alternate routes, and so on) the arrival delay between packets might be more than or less than 10 ms with a positive jitter value meaning that the packets arrived more than 10 ms apart. If the packets arrive 12 ms apart, positive jitter is 2 ms; if the packets arrive 8 ms apart, negative jitter is 2 ms. For delay-sensitive networks, positive jitter values are undesirable, and a jitter value of 0 is ideal.

In addition to monitoring jitter, the IP SLAs UDP jitter operation can be used as a multipurpose data gathering operation. The packets IP SLAs generates carry packet sending and receiving sequence information and sending and receiving time stamps from the source and the operational target. Based on these, UDP jitter operations measure this data:

- Per-direction jitter (source to destination and destination to source)
- Per-direction packet-loss
- Per-direction delay (one-way delay)
- Round-trip delay (average round-trip time)

Because the paths for the sending and receiving of data can be different (asymmetric), you can use the per-direction data to more readily identify where congestion or other problems are occurring in the network.

The UDP jitter operation generates synthetic (simulated) UDP traffic and sends a number of UDP packets, each of a specified size, sent a specified number of milliseconds apart, from a source router to a target router, at a given frequency. By default, ten packet-frames, each with a payload size of 10 bytes are generated every 10 ms, and the operation is repeated every 60 seconds. You can configure each of these parameters to best simulate the IP service you want to provide.

To provide accurate one-way delay (latency) measurements, time synchronization, such as that provided by NTP, is required between the source and the target device. Time synchronization is not required for the one-way jitter and packet loss measurements. If the time is not synchronized between the source and target devices, one-way jitter and packet loss data is returned, but values of *0* are returned for the one-way delay measurements provided by the UDP jitter operation

Note Before you configure a UDP jitter operation on the source device, you must enable the IP SLAs responder on the target device (the operational target).

Beginning in privileged EXEC mode, follow these steps to configure UDP jitter operation on the source device:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip sla** *operation-number*	Create an IP SLAs operation, and enter IP SLAs configuration mode.

	Command	Purpose
Step 3	**udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*]	Configure the IP SLAs operation as a UDP jitter operation, and enter UDP jitter configuration mode. • *destination-ip-address* | *destination-hostname*—Specify the destination IP address or hostname. • *destination-port*—Specify the destination port number in the range from 1 to 65535. • (Optional) **source-ip** {*ip-address* | *hostname*}—Specify the source IP address or hostname. When a source IP address or hostname is not specified, IP SLAs chooses the IP address nearest to the destination • (Optional) **source-port** *port-number*—Specify the source port number in the range from 1 to 65535. When a port number is not specified, IP SLAs chooses an available port. • (Optional) **control**—Enable or disable sending of IP SLAs control messages to the IP SLAs responder. By default, IP SLAs control messages are sent to the destination device to establish a connection with the IP SLAs responder • (Optional) **num-packets** *number-of-packets*—Enter the number of packets to be generated. The range is 1 to 6000; the default is 10. • (Optional) **interval** *inter-packet-interval*—Enter the interval between sending packets in milliseconds. The range is 1 to 6000; the default value is 20 ms.
Step 4	**frequency** *seconds*	(Optional) Set the rate at which a specified IP SLAs operation repeats. The range is from 1 to 604800 seconds; the default is 60 seconds.
Step 5	**exit**	Exit UDP jitter configuration mode, and return to global configuration mode.
Step 6	**ip sla monitor schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh*:*mm* [:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]	Configure the scheduling parameters for an individual IP SLAs operation. • *operation-number*—Enter the RTR entry number. • (Optional) **life**—Set the operation to run indefinitely (**forever**) or for a specific number of *seconds*. The range is from 0 to 2147483647. The default is 3600 seconds (1 hour). • (Optional) **start-time**—Enter the time for the operation to begin collecting information: – To start at a specific time, enter the hour, minute, second (in 24-hour notation), and day of the month. If no month is entered, the default is the current month. – Enter **pending** to select no information collection until a start time is selected. – Enter **now** to start the operation immediately. – Enter **after** *hh:mm:ss* to show that the operation should start after the entered time has elapsed. • (Optional) **ageout** *seconds*—Enter the number of seconds to keep the operation in memory when it is not actively collecting information. The range is 0 to 2073600 seconds, the default is 0 seconds (never ages out). • (Optional) **recurring**—Set the operation to automatically run every day.

	Command	Purpose
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show ip sla configuration** [*operation-number*]	(Optional) Display configuration values, including all defaults for all IP SLAs operations or a specified operation.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the IP SLAs operation, enter the no **ip sla** *operation-number* global configuration command. This example shows how to configure a UDP jitter IP SLAs operation:

```
Switch(config)# ip sla 10
Switch(config-ip-sla)# udp-jitter 172.29.139.134 5000
Switch(config-ip-sla-jitter)# frequency 30
Switch(config-ip-sla-jitter)# exit
Switch(config)# ip sla schedule 5 start-time now life forever
Switch(config)# end
Switch# show ip sla configuration 10
IP SLAs, Infrastructure Engine-II.

Entry number: 10
Owner:
Tag:
Type of operation to perform: udp-jitter
Target address/Source address: 1.1.1.1/0.0.0.0
Target port/Source port: 2/0
Request size (ARR data portion): 32
Operation timeout (milliseconds): 5000
Packet Interval (milliseconds)/Number of packets: 20/10
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Control Packets: enabled
Schedule:
    Operation frequency (seconds): 30
    Next Scheduled Start Time: Pending trigger
    Group Scheduled : FALSE
    Randomly Scheduled : FALSE
    Life (seconds): 3600
    Entry Ageout (seconds): never
    Recurring (Starting Everyday): FALSE
    Status of entry (SNMP RowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
    Number of statistic hours kept: 2
    Number of statistic distribution buckets kept: 1
    Statistic distribution interval (milliseconds): 20
Enhanced History:
```

Analyzing IP Service Levels by Using the ICMP Echo Operation

The ICMP echo operation measures end-to-end response time between a Cisco device and any devices using IP. Response time is computed by measuring the time taken between sending an ICMP echo request message to the destination and receiving an ICMP echo reply. Many customers use IP SLAs ICMP-based operations, in-house ping testing, or ping-based dedicated probes for response time measurements between the source IP SLAs device and the destination IP device. The IP SLAs ICMP echo operation conforms to the same specifications as ICMP ping testing, and the two methods result in the same response times.

Note This operation does not require the IP SLAs responder to be enabled.

Beginning in privileged EXEC mode, follow these steps to configure an ICMP echo operation on the source device:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip sla** *operation-number*	Create an IP SLAs operation and enter IP SLAs configuration mode.
Step 3	**icmp-echo** {*destination-ip-address* | *destination-hostname*} [**source-ip** {*ip-address* | *hostname*} | **source-interface** *interface-id*]	Configure the IP SLAs operation as an ICMP Echo operation and enter ICMP echo configuration mode. • *destination-ip-address* | *destination-hostname*—Specify the destination IP address or hostname. • (Optional) **source-ip** {*ip-address* | *hostname*}—Specify the source IP address or hostname. When a source IP address or hostname is not specified, IP SLAs chooses the IP address nearest to the destination • (Optional) **source-interface** *interface-id*—Specify the source interface for the operation.
Step 4	**frequency** *seconds*	(Optional) Set the rate at which a specified IP SLAs operation repeats. The range is from 1 to 604800 seconds; the default is 60 seconds.
Step 5	**exit**	Exit UDP jitter configuration mode, and return to global configuration mode.
Step 6	**ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh*:*mm* [:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]	Configure the scheduling parameters for an individual IP SLAs operation. • *operation-number*—Enter the RTR entry number. • (Optional) **life**—Set the operation to run indefinitely (**forever**) or for a specific number of *seconds*. The range is from 0 to 2147483647. The default is 3600 seconds (1 hour) • (Optional) **start-time**—Enter the time for the operation to begin collecting information: – To start at a specific time, enter the hour, minute, second (in 24-hour notation), and day of the month. If no month is entered, the default is the current month. – Enter **pending** to select no information collection until a start time is selected. – Enter **now** to start the operation immediately. – Enter **after** *hh:mm:ss* to indicate that the operation should start after the entered time has elapsed. • (Optional) **ageout** *seconds*—Enter the number of seconds to keep the operation in memory when it is not actively collecting information. The range is 0 to 2073600 seconds; the default is 0 seconds (never ages out). • (Optional) **recurring**—Set the operation to automatically run every day.
Step 7	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 8	**show ip sla configuration** [*operation-number*]	(Optional) Display configuration values including all defaults for all IP SLAs operations or a specified operation.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the IP SLAs operation, enter the **no ip sla** *operation-number* global configuration command. This example shows how to configure an ICMP echo IP SLAs operation:

```
Switch(config)# ip sla 12
Switch(config-ip-sla)# icmp-echo 172.29.139.134
Switch(config-ip-sla-echo)# frequency 30
Switch(config-ip-sla-echo)# exit
Switch(config)# ip sla schedule 5 start-time now life forever
Switch(config)# end
Switch# show ip sla configuration 22
IP SLAs, Infrastructure Engine-II.

Entry number: 12
Owner:
Tag:
Type of operation to perform: echo
Target address: 2.2.2.2
Source address: 0.0.0.0
Request size (ARR data portion): 28
Operation timeout (milliseconds): 5000
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Schedule:
    Operation frequency (seconds): 60
    Next Scheduled Start Time: Pending trigger
    Group Scheduled : FALSE
    Randomly Scheduled : FALSE
    Life (seconds): 3600
    Entry Ageout (seconds): never
    Recurring (Starting Everyday): FALSE
    Status of entry (SNMP RowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
    Number of statistic hours kept: 2
    Number of statistic distribution buckets kept: 1
    Statistic distribution interval (milliseconds): 20
History Statistics:
    Number of history Lives kept: 0
    Number of history Buckets kept: 15
    History Filter Type: None
Enhanced History:
```

Monitoring IP SLAs Operations

Use the User EXEC or Privileged EXEC commands in Table 1-1 to display IP SLAs operations configuration and results.

Table 1-1 Monitoring IP SLAs Operations

Command	Purpose
show ip sla application	Display global information about Cisco IOS IP SLAs.
show ip sla authentication	Display IP SLAs authentication information.
show ip sla configuration [*entry-number*]	Display configuration values including all defaults for all IP SLAs operations or a specific operation.
show ip sla enhanced-history {**collection-statistics** | **distribution statistics**} [*entry-number*]	Display enhanced history statistics for collected history buckets or distribution statistics for all IP SLAs operations or a specific operation.
show ip sla ethernet-monitor configuration [*entry-number*]	Display IP SLAs automatic Ethernet configuration.
show ip sla group schedule [*schedule-entry-number*]	Display IP SLAs group scheduling configuration and details.
show ip sla history [*entry-number* | **full** | **tabular**]	Display history collected for all IP SLAs operations
show ip sla mpls-lsp-monitor {**collection-statistics** | **configuration** | **ldp operational-state** | **scan-queue** | **summary** [*entry-number*] | **neighbors**}	Display MPLS label switched path (LSP) Health Monitor operations,
show ip sla reaction-configuration [*entry-number*]	Display the configured proactive threshold monitoring settings for all IP SLAs operations or a specific operation.
show ip sla reaction-trigger [*entry-number*]	Display the reaction trigger information for all IP SLAs operations or a specific operation.
show ip sla responder	Display information about the IP SLAs responder.
show ip sla statistics [*entry-number* | **aggregated** | **details**]	Display current or aggregated operational status and statistics.

CHAPTER 2

Configuring Web Cache Services By Using WCCP

This chapter describes how to configure your Catalyst 3550 switch to redirect traffic to cache engines (web caches such as the Cisco Cache Engine 550) by using the Web Cache Communication Protocol (WCCP). WCCP is a Cisco-developed content-routing technology that you can use to integrate cache engines into your network infrastructure. The cache engines transparently store frequently accessed content and then fulfill successive requests for the same content, eliminating repetitive transmissions of identical content from web servers. Cache engines accelerate content delivery and ensure maximum scalability and availability of content. In a service-provider network, you can deploy the WCCP and cache engine solution at the points of presence (POPs). In an enterprise network, you can deploy the WCCP and cache engine solution at the regional site and the small branch office.

To use this feature, you must have the IP services image (formerly known as the enhanced multilayer image [EMI]) installed on your switch.

Note For complete syntax and usage information for the commands used in this chapter, see the "WCCP Router Configuration Commands" section in the "*System Management Commands*" part of the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.2*.

This chapter consists of these sections:

- Understanding WCCP, page 2-1
- Configuring WCCP, page 2-4
- Monitoring and Maintaining WCCP, page 2-8

Understanding WCCP

The WCCP and Cisco cache engines (or other caches running WCCP) localize web-traffic patterns in the network, enabling content requests to be fulfilled locally.

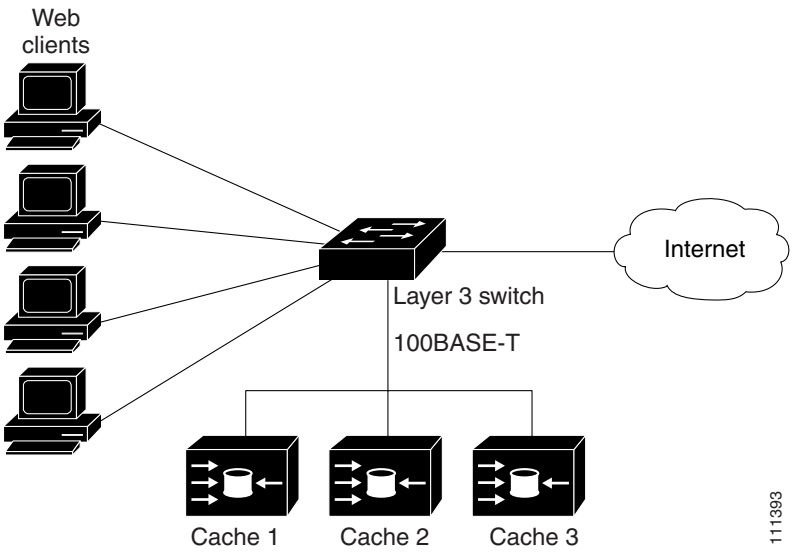
WCCP enables supported Cisco routers and switches to transparently redirect content requests. With transparent redirection, users do not have to configure their browsers to use a web proxy. Instead, they can use the target URL to request content, and their requests are automatically redirected to a cache engine. The word *transparent* means that the end user does not know that a requested file (such as a web page) came from the cache engine instead of from the originally specified server.

When a cache engine receives a request, it attempts to service it from its own local cache. If the requested information is not present, the cache engine sends a separate request to the end server to retrieve the requested information. After receiving the requested information, the cache engine forwards it to the requesting client and also caches it to fulfill future requests.

This software release supports only WCCP version 2 (WCCPv2). Only a subset of WCCPv2 features are supported. For more information, see the "Unsupported WCCPv2 Features" section on page 2-4.

With WCCPv2, multiple routers or switches can service the cache-engine cluster (a series of cache engines); however, in this release, only one Catalyst 3550 switch can service the cluster, as shown Figure 2-1. Content is not duplicated on the cache engines.

Figure 2-1 Cisco Cache Engine and WCCPv2 Network Configuration



WCCP Message Exchange

This sequence of events describes the WCCP message exchange:

1. The cache engines send their IP addresses to the WCCP-enabled switch by using WCCP, signaling their presence through a *Here I am* message. The switch and cache engines communicate to each other through a control channel based on UDP port 2048.
2. The WCCP-enabled switch uses the cache engine IP information to create a cluster view (a list of caches in the cluster). This view is sent through an *I see you* message to each cache engine in the cluster, essentially making all the cache engines aware of each other. A stable view is established after the membership of the cluster remains the same for a certain amount of time.
3. When a stable view is established, the cache engine in the cluster with the lowest IP address is elected as the designated cache engine.

WCCP Negotiation

In the exchange of WCCP protocol messages, the designated cache engine and the WCCP-enabled switch negotiate these items:

- Forwarding method (the method by which the switch forwards packets to the cache engine). The switch rewrites the Layer 2 header by replacing the packet's destination MAC address with the target cache engine's MAC address. It then forwards the packet to the cache engine. This forwarding method requires the target cache engine to be directly connected to the switch at Layer 2.
- Assignment method (the method by which packets are distributed among the cache engines in the cluster). The switch uses some of the least-significant bits of the destination IP address to determine which cache engine receives the redirected packet. The number of bits used is based on the number of cache engines. If the number of cache engines is equal to a power of 2 (for example, 1, 2, 4 and so forth), the switch evenly distributes (load balances) the traffic among the cache engines.

  The switch does not support the mask assignment method described in the *WCCP V2.0 Internet Draft*.
- Packet-return method (the method by which packets are returned from the cache engine to the switch for normal forwarding). These are the typical reasons why a cache engine rejects packets and initiates the packet-return feature:
  - The cache engine is overloaded and has no room to service the packets.
  - The cache engine receives an error message (such as a protocol or authentication error) from the web server and implements the dynamic client bypass feature. The bypass enables clients to bypass the cache engines and to connect directly to the web server.

  The cache engine returns a packet to the WCCP-enabled switch to forward to the web server as if the cache engine is not present. The cache engine does not intercept the reconnection attempt. In this way, the cache engine effectively cancels the redirection of a packet to the cache engine and creates a bypass flow. The switch receives the returned packet through a generic-route encapsulation (GRE) tunnel. The switch CPU uses Cisco express forwarding (CEF) to send these packets to the target web server. When the server responds with the requested information, the switch uses the normal Layer 3 forwarding to return the information to the requesting client.

MD5 Security

WCCPv2 provides an optional security component in each protocol message to enable the switch to use MD5 authentication on messages between the switch and the cache engine. Messages that do not authenticate (when authentication of the switch is enabled) are discarded by the switch. You enable the security feature by using the **ip wccp web-cache password** *password* global configuration command. The password string is combined with the MD5 value to create security for the connection between the switch and the cache engine. You must configure the same password on each cache engine.

Packet Redirection

After WCCP is configured on the switch, the switch forwards all HTTP TCP port 80 packets received from clients to the cache engines. However, these packets are not redirected:

- Packets originating from the cache engine and targeted to the web server.
- Packets originating from the cache engine and targeted to the client.
- Packets returned or rejected by the cache engine. These packets are sent to the web server.

Unsupported WCCPv2 Features

These WCCPv2 features are not supported in this software release:

- WCCP service numbers, which are configured by using the **ip wccp** [*service-number*] global and interface configuration commands. These commands are not supported.

  This software release supports caching only for TCP port 80.
- Packet redirection on an outbound interface, which is configured by using the **ip wccp redirect out** interface configuration command. This command is not supported.

  This software release supports packet redirection only on an inbound interface.
- The connection of multiple Catalyst 3550 switches to multiple cache engines.

  This software release supports the connection of only one switch to multiple cache engines.
- WCCP multicasting. The **ip wccp web-cache group-address** and **ip wccp web-cache group listen** global configuration commands are not supported.
- WCCP access lists. The **ip wccp web-cache redirect-list** and **ip wccp web-cache group-list** global configuration commands are not supported.
- Statistics for WCCP-related counters. Statistics for counters are not provided; they appear as zeros in the **show ip wccp web-cache view** privileged EXEC command output.

Configuring WCCP

These sections describe how to configure WCCP on your switch:

- Default WCCP Configuration, page 2-4
- WCCP Configuration Guidelines, page 2-4
- Enabling the Web Cache Service, Setting the Password, and Redirecting Traffic Received From a Client, page 2-5 (required)

Default WCCP Configuration

Table 2-1 shows the default WCCP configuration.

Table 2-1 Default WCCP Configuration

Feature	Default Setting
WCCP enable state.	WCCP services are disabled.
Protocol version.	WCCPv2.
Redirecting traffic received on an interface.	Disabled.

WCCP Configuration Guidelines

Before configuring WCCP on your switch, make sure to follow these configuration guidelines:

- Do not configure the cache engine for GRE because the switch does not support traffic forwarding by using GRE. For more information, see the documentation that shipped with the cache engines.

- Make a direct Layer 2 connection from the cache engines to the switch so that the switch can perform Layer 2 rewrites for WCCP redirection. The Cisco Cache Engines require the use of a Fast Ethernet interface for a direct connection. You also can connect the switch to the cache engine by using a 10/100/1000 port if the connection is a direct Layer 2 connection.
- Connect up to 32 cache engines to a single Catalyst 3550 switch.
- Connect only one Catalyst 3550 switch to multiple cache engines. Do not connect multiple Catalyst 3550 switches to multiple cache engines.
- Configure the switch interfaces that are connected to the web clients, the cache engines, and the web server as Layer 3 interfaces (routed ports and switch virtual interfaces [SVIs]). For HTTP packet redirection to work, the servers, cache engines, and clients must be on different subnets.
- Do not configure the clients, cache engines, or web servers on the same switch interface.
- Do not configure the switch with both WCCP and multiple VPN routing/forwarding (multi-VRF) instances in customer edge (CE) devices.
- Modify the Switch Database Management (SDM) template to enable the switch to support 144-bit Layer 3 TCAM by using the **sdm prefer extended-match**, **sdm prefer access extended-match**, or **sdm prefer routing extended-match** global configuration command. For more information on the SDM templates, see the "Optimizing System Resources for User-Selected Features" section on page 6-26.
- Do not configure WCCP and policy-based routing (PBR) on the same switch interface.

Enabling the Web Cache Service, Setting the Password, and Redirecting Traffic Received From a Client

MD5 password security requires that the switch and cache engines be configured with the same password. Each cache engine or switch authenticates the security component in a received WCCP packet immediately after validating the WCCP message header. Packets failing authentication are discarded.

For WCCP packet redirection to operate, you must configure the switch interface connected to the client to redirect inbound HTTP packets.

This procedure shows how to configure these features on routed ports. To configure these features on SVIs, see the configuration examples that follow the procedure.

Beginning in privileged EXEC mode, follow these steps to enable the web cache service, to set a password, to configure routed interfaces, and to redirect inbound packets received from a client to the cache engine. This procedure is required.

Note Before configuring WCCP commands, configure the SDM template, and reboot the switch. For more information, see the "Optimizing System Resources for User-Selected Features" section on page 6-26.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip wccp web-cache** [**password** *encryption-number password*]	Enable the web cache service on your switch. By default, this feature is disabled. (Optional) For [**password** *encryption-number password*], specify an encryption number. The range is 0 to 7. Use 0 for not encrypted, and use 7 for proprietary. Specify a password name up to seven characters in length. The switch combines the password with the MD5 authentication value to create security for the connection between the switch and the cache engine. By default, no password is configured, and no authentication is performed. You must configure the same password on each cache engine. When authentication is enabled, the switch discards messages that are not authenticated.
Step 3	**interface** *interface-id*	Specify the interface connected to the cache engine or the web server, and enter interface configuration mode.
Step 4	**no switchport**	Enter Layer 3 mode.
Step 5	**ip address** *ip-address subnet-mask*	Configure the IP address and subnet mask.
Step 6	**no shutdown**	Enable the interface.
Step 7	**exit**	Return to global configuration mode. Repeat Steps 3 through 7 for each cache engine and web server.
Step 8	**interface** *interface-id*	Specify the interface connected to the client, and enter interface configuration mode.
Step 9	**no switchport**	Enter Layer 3 mode.
Step 10	**ip address** *ip-address subnet-mask*	Configure the IP address and subnet mask.
Step 11	**no shutdown**	Enable the interface.
Step 12	**ip wccp web-cache redirect in**	Redirect packets received from the client to the cache engine.
Step 13	**exit**	Return to global configuration mode. Repeat Steps 8 through 13 for each client.
Step 14	**end**	Return to privileged EXEC mode.
Step 15	**show ip wccp web-cache** and **show running-config**	Verify your entries.
Step 16	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the web cache service, use the **no ip wccp web-cache** global configuration command. To disable inbound packet redirection, use the **no ip wccp web-cache redirect in** interface configuration command.

This example shows how to configure routed interfaces and to enable the web cache service. Fast Ethernet port 1 is connected to the cache engine, is configured as a routed port with an IP address of 172.20.10.30, and is re-enabled. Gigabit Ethernet port 1 is connected through the Internet to the web server, is configured as a routed port with an IP address of 175.20.20.10, and is re-enabled. Fast Ethernet

ports 2 to 5 are connected to the clients and are configured as routed ports with IP addresses 175.20.30.20, 175.20.40.30, 175.20.50.40, and 175.20.60.50. The switch redirects HTTP packets received from the client interfaces to the cache engine.

```
Switch# configure terminal
Switch(config)# ip wccp web-cache
Switch(config)# interface fastethernet0/1
Switch(config-if)# no switchport
Switch(config-if)# ip address 172.20.10.30 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# no switchport
Switch(config-if)# ip address 175.20.20.10 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# exit
Switch(config)# interface fastethernet0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 175.20.30.20 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# ip wccp web-cache redirect in
Switch(config-if)# exit
Switch(config)# interface fastethernet0/3
Switch(config-if)# no switchport
Switch(config-if)# ip address 175.20.40.30 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# ip wccp web-cache redirect in
Switch(config-if)# exit
Switch(config)# interface fastethernet0/4
Switch(config-if)# no switchport
Switch(config-if)# ip address 175.20.50.40 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# ip wccp web-cache redirect in
Switch(config-if)# exit
Switch(config)# interface fastethernet0/5
Switch(config-if)# no switchport
Switch(config-if)# ip address 175.20.60.50 255.255.255.0
Switch(config-if)# no shutdown
Switch(config-if)# ip wccp web-cache redirect in
Switch(config-if)# exit
```

This example shows how to configure SVIs and how to enable the web cache service. VLAN 299 is created and configured with an IP address of 175.20.20.10. Gigabit Ethernet port 1 is connected through the Internet to the web server and is configured as an access port in VLAN 299. VLAN 300 is created and configured with an IP address of 172.20.10.30. Fast Ethernet port 1 is connected to the cache engine and is configured as an access port in VLAN 300. VLAN 301 is created and configured with an IP address of 175.20.30.50. Fast Ethernet ports 2 to 5, which are connected to the clients, are configured as access ports in VLAN 301. The switch redirects HTTP packets received from the client interfaces to the cache engine.

```
Switch# configure terminal
Switch(config)# ip wccp web-cache
Switch(config)# vlan 299
Switch(config-vlan)# exit
Switch(config)# interface vlan 299
Switch(config-if)# ip address 175.20.20.10 255.255.255.0
Switch(config-if)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 299
Switch(config)# vlan 300
Switch(config-vlan)# exit
```

```
Switch(config)# interface vlan 300
Switch(config-if)# ip address 172.20.10.30 255.255.255.0
Switch(config-if)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 300
Switch(config-if)# exit
Switch(config)# vlan 301
Switch(config-vlan)# exit
Switch(config)# interface vlan 301
Switch(config-if)# ip address 175.20.30.20 255.255.255.0
Switch(config-if)# ip wccp web-cache redirect in
Switch(config-if)# exit
Switch(config)# interface range fastethernet0/2 - 5
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 301
Switch(config-if-range)# exit
```

Monitoring and Maintaining WCCP

To monitor and maintain WCCP, use one or more of the privileged EXEC commands in Table 2-2:

Table 2-2 Commands for Monitoring and Maintaining WCCP

Command	Purpose
clear ip wccp web-cache	Removes statistics for the web-cache service.
show ip wccp web-cache	Displays global information related to WCCP.
show ip wccp web-cache detail	Displays information for the switch and all cache engines in the WCCP cluster.
show ip interface	Displays status about any IP WCCP redirection commands that are configured on an interface; for example, *Web Cache Redirect is enabled / disabled.*
show ip wccp web-cache view	Displays which other members have or have not been detected.

CHAPTER 3

Configuring IP Multicast Routing

IP multicasting is a more efficient way to use network resources, especially for bandwidth-intensive services such as audio and video. IP multicast allows a host (source) to send packets to a group of hosts (receivers) anywhere within the IP network by using a special form of IP address called the IP *multicast group address*. The sending host inserts the multicast group address into the IP destination address field of the packet, and IP multicast routers and multilayer switches forward incoming IP multicast packets out all interfaces that lead to members of the multicast group.

IP multicast addresses are assigned to the old class D address space by the Internet Assigned Number Authority (IANA). The high-order bits of a Class D address are 1110. Therefore, host group addresses can be in the range 224.0.0.0 through 239.255.255.255. The address 224.0.0.0 is guaranteed not to be assigned to any group. The address 224.0.0.1 is assigned to the all-hosts multicast group on a subnet. The address 224.0.0.2 is assigned to the all-multicast-routers group on a subnet.

Any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group receive the message. Membership in a multicast group is dynamic; hosts can join and leave at any time. There is no restriction on the location or number of members in a multicast group. A host can be a member of more than one multicast group at a time. How active a multicast group is and what members it has can vary from group to group and from time to time. A multicast group can be active for a long time, or it can be very short-lived. Membership in a group can constantly change. A group that has members can have no activity.

Note For complete syntax and usage information for the commands used in this chapter, see the *Cisco IOS IP Command Reference, Volume 3 of 3: Multicast, Release 12.2*.

This chapter describes how to configure IP multicast routing on your Catalyst 3550 multilayer switch. To use this feature, you must have the IP services image, formerly known as the enhanced multilayer software image (EMI), installed on your switch.

This chapter consists of these sections:

- Understanding Cisco's Implementation of IP Multicast Routing, page 3-2
- Configuring IP Multicast Routing, page 3-8
- Configuring Advanced PIM Features, page 3-23
- Configuring Optional IGMP Features, page 3-26
- Configuring Optional Multicast Routing Features, page 3-32
- Configuring Basic DVMRP Interoperability Features, page 3-38
- Configuring Advanced DVMRP Interoperability Features, page 3-43
- Monitoring and Maintaining IP Multicast Routing, page 3-51

For information on configuring the Multicast Source Discovery Protocol (MSDP), see Chapter 4, "Configuring MSDP."

Note When you are configuring multicast routing parameters for the switch, to allocate system resources to maximize the number of possible multicast routes allowed, you can use the **sdm prefer routing** global configuration command to set the Switch Database Management feature to the routing template. For more information on the SDM templates, see the "Optimizing System Resources for User-Selected Features" section on page 6-26.

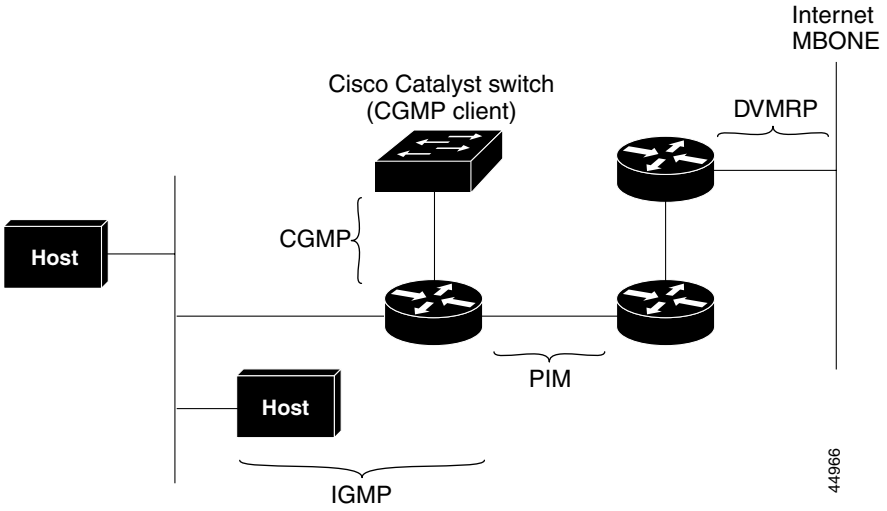
Understanding Cisco's Implementation of IP Multicast Routing

The Cisco IOS software supports these protocols to implement IP multicast routing:

- Internet Group Management Protocol (IGMP) is used among hosts on a LAN and the routers (and multilayer switches) on that LAN to track the multicast groups of which hosts are members.
- Protocol-Independent Multicast (PIM) protocol is used among routers and multilayer switches to track which multicast packets to forward to each other and to their directly connected LANs.
- Distance Vector Multicast Routing Protocol (DVMRP) is used on the multicast backbone of the Internet (MBONE). The Cisco IOS software supports PIM-to-DVMRP interaction.
- Cisco Group Management Protocol (CGMP) is used on Cisco routers and multilayer switches connected to Layer 2 Catalyst switches to perform tasks similar to those performed by IGMP.

Figure 3-1 shows where these protocols operate within the IP multicast environment.

Figure 3-1 IP Multicast Routing Protocols



Understanding IGMP

To participate in IP multicasting, multicast hosts, routers, and multilayer switches must have the IGMP operating. This protocol defines the querier and host roles:

- A querier is a network device that sends query messages to discover which network devices are members of a given multicast group.
- A host is a receiver that sends report messages (in response to query messages) to inform a querier of a host membership.

A set of queriers and hosts that receive multicast data streams from the same source is called a multicast group. Queriers and hosts use IGMP messages to join and leave multicast groups.

Any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group receive the message. Membership in a multicast group is dynamic; hosts can join and leave at any time. There is no restriction on the location or number of members in a multicast group. A host can be a member of more than one multicast group at a time. How active a multicast group is and what members it has can vary from group to group and from time to time. A multicast group can be active for a long time, or it can be very short-lived. Membership in a group can constantly change. A group that has members can have no activity.

IP multicast traffic uses group addresses, which are class D addresses. The high-order bits of a Class D address are 1110. Therefore, host group addresses can be in the range 224.0.0.0 through 239.255.255.255. Multicast addresses in the range 224.0.0.0 to 224.0.0.255 are reserved for use by routing protocols and other network control traffic. The address 224.0.0.0 is guaranteed not to be assigned to any group.

IGMP packets are sent using these IP multicast group addresses:

- IGMP general queries are destined to the address 224.0.0.1 (all systems on a subnet).
- IGMP group-specific queries are destined to the group IP address for which the switch is querying.
- IGMP group membership reports are destined to the group IP address for which the switch is reporting.
- IGMP Version 2 (IGMPv2) leave messages are destined to the address 224.0.0.2 (all-multicast-routers on a subnet). In some old host IP stacks, leave messages might be destined to the group IP address rather than to the all-routers address.

IGMP Version 1

IGMP Version 1 (IGMPv1) primarily uses a query-response model that enables the multicast router and multilayer switch to find which multicast groups are active (have one or more hosts interested in a multicast group) on the local subnet. IGMPv1 has other processes that enable a host to join and leave a multicast group. For more information, see RFC 1112.

IGMP Version 2

IGMPv2 extends IGMP functionality by providing such features as the IGMP leave process to reduce leave latency, group-specific queries, and an explicit maximum query response time. IGMPv2 also adds the capability for routers to elect the IGMP querier without depending on the multicast protocol to perform this task. For more information, see RFC 2236.

Understanding PIM

PIM is called *protocol-independent*: regardless of the unicast routing protocols used to populate the unicast routing table, PIM uses this information to perform multicast forwarding instead of maintaining a separate multicast routing table.

PIM is defined in RFC 2362, *Protocol-Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification*. PIM is defined in these Internet Engineering Task Force (IETF) Internet drafts:

- *Protocol Independent Multicast (PIM): Motivation and Architecture*
- *Protocol Independent Multicast (PIM), Dense Mode Protocol Specification*
- *Protocol Independent Multicast (PIM), Sparse Mode Protocol Specification*
- *draft-ietf-idmr-igmp-v2-06.txt, Internet Group Management Protocol, Version 2*
- *draft-ietf-pim-v2-dm-03.txt, PIM Version 2 Dense Mode*

PIM Versions

PIMv2 includes these improvements over PIMv1:

- A single, active rendezvous point (RP) exists per multicast group, with multiple backup RPs. This single RP compares to multiple active RPs for the same group in PIMv1.
- A bootstrap router (BSR) provides a fault-tolerant, automated RP discovery and distribution mechanism that enables routers and multilayer switches to dynamically learn the group-to-RP mappings.
- Sparse mode and dense mode are properties of a group, as opposed to an interface. We strongly recommend sparse-dense mode, as opposed to either sparse mode or dense mode only.
- PIM join and prune messages have more flexible encoding for multiple address families.
- A more flexible hello packet format replaces the query packet to encode current and future capability options.
- Register messages to an RP specify whether they are sent by a border router or a designated router.
- PIM packets are no longer inside IGMP packets; they are standalone packets.

PIM Modes

PIM can operate in dense mode (DM), sparse mode (SM), or in sparse-dense mode (PIM DM-SM), which handles both sparse groups and dense groups at the same time.

PIM DM

PIM DM builds source-based multicast distribution trees. In dense mode, a PIM DM router or multilayer switch assumes that all other routers or multilayer switches forward multicast packets for a group. If a PIM DM device receives a multicast packet and has no directly connected members or PIM neighbors present, a prune message is sent back to the source to stop unwanted multicast traffic. Subsequent multicast packets are not flooded to this router or switch on this pruned branch because branches without receivers are pruned from the distribution tree, leaving only branches that contain receivers.

When a new receiver on a previously pruned branch of the tree joins a multicast group, the PIM DM device detects the new receiver and immediately sends a graft message up the distribution tree toward the source. When the upstream PIM DM device receives the graft message, it immediately puts the interface on which the graft was received into the forwarding state so that the multicast traffic begins flowing to the receiver.

PIM SM

PIM SM uses shared trees and shortest-path-trees (SPTs) to distribute multicast traffic to multicast receivers in the network. In PIM SM, a router or multilayer switch assumes that other routers or switches do not forward multicast packets for a group, unless there is an explicit request for the traffic (join message). When a host joins a multicast group using IGMP, its directly connected PIM SM device sends PIM join messages toward the root, also known as the RP. This join message travels router-by-router toward the root, constructing a branch of the shared tree as it goes.

The RP keeps track of multicast receivers. It also registers sources through register messages received from the source's first-hop router (*designated router* [DR]) to complete the shared tree path from the source to the receiver. When using a shared tree, sources must send their traffic to the RP so that the traffic reaches all receivers.

Prune messages are sent up the distribution tree to prune multicast group traffic. This action permits branches of the shared tree or SPT that were created with explicit join messages to be torn down when they are no longer needed.

Auto-RP

This proprietary feature eliminates the need to manually configure the RP information in every router and multilayer switch in the network. For Auto-RP to work, you configure a Cisco router or multilayer switch as the mapping agent. It uses IP multicast to learn which routers or switches in the network are possible candidate RPs to receive candidate RP announcements. Candidate RPs periodically send multicast RP-announce messages to a particular group or group range to announce their availability.

Mapping agents listen to these candidate RP announcements and use the information to create entries in their Group-to-RP mapping caches. Only one mapping cache entry is created for any Group-to-RP range received, even if multiple candidate RPs are sending RP announcements for the same range. As the RP-announce messages arrive, the mapping agent selects the router or switch with the highest IP address as the active RP and stores this RP address in the Group-to-RP mapping cache.

Mapping agents periodically multicast the contents of their Group-to-RP mapping cache. Thus, all routers and switches automatically discover which RP to use for the groups they support. If a router or switch fails to receive RP-discovery messages and the Group-to-RP mapping information expires, it switches to a statically configured RP that was defined with the **ip pim rp-address** global configuration command. If no statically configured RP exists, the router or switch changes the group to dense-mode operation.

Multiple RPs serve different group ranges or serve as hot backups of each other.

Bootstrap Router

PIMv2 BSR is another method to distribute group-to-RP mapping information to all PIM routers and multilayer switches in the network. It eliminates the need to manually configure RP information in every router and switch in the network. However, instead of using IP multicast to distribute group-to-RP mapping information, BSR uses hop-by-hop flooding of special BSR messages to distribute the mapping information.

The BSR is elected from a set of candidate routers and switches in the domain that have been configured to function as BSRs. The election mechanism is similar to the root-bridge election mechanism used in bridged LANs. The BSR election is based on the BSR priority of the device contained in the BSR messages that are sent hop-by-hop through the network. Each BSR device examines the message and forwards out all interfaces only the message that has either a higher BSR priority than its BSR priority or the same BSR priority, but with a higher BSR IP address. Using this method, the BSR is elected.

The elected BSR sends BSR messages with a TTL of 1. Neighboring PIMv2 routers or multilayer switches receive the BSR message and multicast it out all other interfaces (except the one on which it was received) with a TTL of 1. In this way, BSR messages travel hop-by-hop throughout the PIM domain. Because BSR messages contain the IP address of the current BSR, the flooding mechanism enables candidate RPs to automatically learn which device is the elected BSR.

Candidate RPs send candidate RP advertisements showing the group range for which they are responsible to the BSR, which stores this information in its local candidate-RP cache. The BSR periodically advertises the contents of this cache in BSR messages to all other PIM devices in the domain. These messages travel hop-by-hop through the network to all routers and switches, which store the RP information in the BSR message in their local RP cache. The routers and switches select the same RP for a given group because they all use a common RP hashing algorithm.

Multicast Forwarding and Reverse Path Check

With unicast routing, routers and multilayer switches forward traffic through the network along a single path from the source to the destination host whose IP address appears in the destination address field of the IP packet. Each router and switch along the way makes a unicast forwarding decision, using the destination IP address in the packet, by looking up the destination address in the unicast routing table and forwarding the packet through the specified interface to the next hop toward the destination.

With multicasting, the source is sending traffic to an arbitrary group of hosts represented by a multicast group address in the destination address field of the IP packet. To decide whether to forward or drop an incoming multicast packet, the router or multilayer switch uses a reverse path forwarding (RPF) check on the packet as follows and shown in Figure 3-2:

1. The router or multilayer switch examines the source address of the arriving multicast packet to decide whether the packet arrived on an interface that is on the reverse path back to the source.
2. If the packet arrives on the interface leading back to the source, the RPF check is successful and the packet is forwarded to all interfaces in the outgoing interface list (which might not be all interfaces on the router).
3. If the RPF check fails, the packet is discarded.

Some multicast routing protocols, such as DVMRP, maintain a separate multicast routing table and use it for the RPF check. However, PIM uses the unicast routing table to perform the RPF check.

Figure 3-2 shows port 2 receiving a multicast packet from source 151.10.3.21. Table 3-1 shows that the port on the reverse path to the source is port 1, not port 2. Because the RPF check fails, the multilayer switch discards the packet. Another multicast packet from source 151.10.3.21 is received on port 1, and the routing table shows this port is on the reverse path to the source. Because the RPF check passes, the switch forwards the packet to all port in the outgoing port list.

Figure 3-2 RPF Check

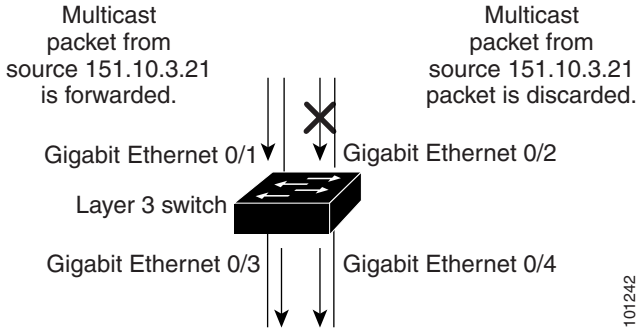


Table 3-1 Routing Table Example for an RPF Check

Network	Port
151.10.0.0/16	Gigabit Ethernet 0/1
198.14.32.0/32	Gigabit Ethernet 0/3
204.1.16.0/24	Gigabit Ethernet 0/4

PIM uses both source trees and RP-rooted shared trees to forward datagrams (described in the "PIM DM" section on page 3-4 and the "PIM SM" section on page 3-5); the RPF check is performed differently for each:

- If a PIM router or multilayer switch has a source-tree state (that is, an (S,G) entry is present in the multicast routing table), it performs the RPF check against the IP address of the source of the multicast packet.
- If a PIM router or multilayer switch has a shared-tree state (and no explicit source-tree state), it performs the RPF check on the rendezvous point (RP) address (which is known when members join the group).

Sparse-mode PIM uses the RPF lookup function to determine where it needs to send joins and prunes:

- (S,G) joins (which are source-tree states) are sent toward the source.
- (*,G) joins (which are shared-tree states) are sent toward the RP.

DVMRP and dense-mode PIM use only source trees and use RPF as previously described.

Understanding DVMRP

DVMRP is implemented in the equipment of many vendors and is based on the public-domain mrouted program. This protocol has been deployed in the MBONE and in other intradomain multicast networks.

Cisco routers and multilayer switches run PIM and can forward multicast packets to and receive from a DVMRP neighbor. It is also possible to propagate DVMRP routes into and through a PIM cloud. The software propagates DVMRP routes and builds a separate database for these routes on each router and multilayer switch, but PIM uses this routing information to make the packet-forwarding decision. The software does not implement the complete DVMRP. However, it supports dynamic discovery of DVMRP routers and can interoperate with them over traditional media (such as Ethernet and FDDI) or over DVMRP-specific tunnels.

DVMRP neighbors build a route table by periodically exchanging source network routing information in route-report messages. The routing information stored in the DVMRP routing table is separate from the unicast routing table and is used to build a source distribution tree and to perform multicast forward using RPF.

DVMRP is a dense-mode protocol and builds a parent-child database using a constrained multicast model to build a forwarding tree rooted at the source of the multicast packets. Multicast packets are initially flooded down this source tree. If redundant paths are on the source tree, packets are not forwarded along those paths. Forwarding occurs until prune messages are received on those parent-child links, which further constrain the broadcast of multicast packets.

Understanding CGMP

This software release provides CGMP-server support on your multilayer switches; no client-side functionality is provided. The multilayer switch serves as a CGMP server for devices that do not support IGMP snooping but have CGMP-client functionality.

CGMP is a protocol used on Cisco routers and multilayer switches connected to Layer 2 Catalyst switches to perform tasks similar to those performed by IGMP. CGMP permits Layer 2 group membership information to be communicated from the CGMP server to the switch. The switch can then learn on which ports multicast members reside instead of flooding multicast traffic to all switch ports. (IGMP snooping is another method to constrain the flooding of multicast packets. For more information, see Chapter 20, “Configuring IGMP Snooping and MVR.”)

CGMP is necessary because the Layer 2 switch cannot distinguish between IP multicast data packets and IGMP report messages, which are both at the MAC-level and are addressed to the same group address.

Configuring IP Multicast Routing

These sections describe how to configure IP multicast routing:

- Default Multicast Routing Configuration, page 3-9
- Multicast Routing Configuration Guidelines, page 3-9
- Configuring Basic Multicast Routing, page 3-10 (required)
- Configuring a Rendezvous Point, page 3-12 (required if the interface is in sparse-dense mode, and you want to treat the group as a sparse group)
- Using Auto-RP and a BSR, page 3-22 (required for non-Cisco PIMv2 devices to interoperate with Cisco PIM v1 devices)
- Monitoring the RP Mapping Information, page 3-23 (optional)
- Troubleshooting PIMv1 and PIMv2 Interoperability Problems, page 3-23 (optional)

Default Multicast Routing Configuration

Table 3-2 shows the default multicast routing configuration.

Table 3-2 Default Multicast Routing Configuration

Feature	Default Setting
Multicast routing	Disabled on all interfaces.
PIM version	Version 2
PIM mode	No mode is defined.
PIM RP address	None configured.
PIM domain border	Disabled.
PIM multicast boundary	None.
Candidate BSRs	Disabled.
Candidate RPs	Disabled.
Shortest-path tree threshold rate	0 kbps.
PIM router query message interval	30 seconds.

Multicast Routing Configuration Guidelines

To avoid misconfiguring multicast routing on your multilayer switch, review the information in these sections:

- PIMv1 and PIMv2 Interoperability, page 3-9
- Auto-RP and BSR Configuration Guidelines, page 3-10

PIMv1 and PIMv2 Interoperability

The Cisco PIMv2 implementation allows interoperability and transition between Version 1 and Version 2, although there might be some minor problems.

You can upgrade to PIMv2 incrementally. PIM Versions 1 and 2 can be configured on different routers and multilayer switches within one network. Internally, all routers and multilayer switches on a shared media network must run the same PIM version. Therefore, if a PIMv2 device detects a PIMv1 device, the Version 2 device downgrades itself to Version 1 until all Version 1 devices have been shut down or upgraded.

PIMv2 uses the BSR to discover and announce RP-set information for each group prefix to all the routers and multilayer switches in a PIM domain. PIMv1, together with the Auto-RP feature, can perform the same tasks as the PIMv2 BSR. However, Auto-RP is a standalone protocol, separate from PIMv1, and is a proprietary Cisco protocol. PIMv2 is a standards track protocol in the IETF. We recommend that you use PIMv2. The BSR mechanism interoperates with Auto-RP on Cisco routers and multilayer switches. For more information, see the "Auto-RP and BSR Configuration Guidelines" section on page 3-10.

When PIMv2 devices interoperate with PIMv1 devices, Auto-RP should have already been deployed. A PIMv2 BSR that is also an Auto-RP mapping agent automatically advertises the RP elected by Auto-RP. That is, Auto-RP sets its single RP on every router or multilayer switch in the group. Not all routers and switches in the domain use the PIMv2 hash function to select multiple RPs.

Dense-mode groups in a mixed PIMv1 and PIMv2 region need no special configuration; they automatically interoperate.

Sparse-mode groups in a mixed PIMv1 and PIMv2 region are possible because the Auto-RP feature in PIMv1 interoperates with the PIMv2 RP feature. Although all PIMv2 devices can also use PIMv1, we recommend that the RPs be upgraded to PIMv2. To ease the transition to PIMv2, we have these recommendations:

- Use Auto-RP throughout the region.
- Configure sparse-dense mode throughout the region.

If Auto-RP is not already configured in the PIMv1 regions, configure Auto-RP. For more information, see the "Configuring Auto-RP" section on page 3-14.

Auto-RP and BSR Configuration Guidelines

There are two approaches to using PIMv2. You can use Version 2 exclusively in your network or migrate to Version 2 by employing a mixed PIM version environment.

- If your network is all Cisco routers and multilayer switches, you can use either Auto-RP or BSR.
- If you have non-Cisco routers in your network, you must use BSR.
- If you have Cisco PIMv1 and PIMv2 routers and multilayer switches and non-Cisco routers, you must use both Auto-RP and BSR. If your network includes routers from other vendors, configure the Auto-RP mapping agent and the BSR on a Cisco PIMv2 device. Ensure that no PIMv1 device is located in the path a between the BSR and a non-Cisco PIMv2 device.
- Because bootstrap messages are sent hop-by-hop, a PIMv1 device prevents these messages from reaching all routers and multilayer switches in your network. Therefore, if your network has a PIMv1 device in it and only Cisco routers and multilayer switches, it is best to use Auto-RP.
- If you have a network that includes non-Cisco routers, configure the Auto-RP mapping agent and the BSR on a Cisco PIMv2 router or multilayer switch. Ensure that no PIMv1 device is on the path between the BSR and a non-Cisco PIMv2 router.
- If you have non-Cisco PIMv2 routers that need to interoperate with Cisco PIMv1 routers and multilayer switches, both Auto-RP and a BSR are required. We recommend that a Cisco PIMv2 device be both the Auto-RP mapping agent and the BSR. For more information, see the "Using Auto-RP and a BSR" section on page 3-22.

Configuring Basic Multicast Routing

You must enable IP multicast routing and configure the PIM version and the PIM mode. Then the software can forward multicast packets, and the switch can populate its multicast routing table.

You can configure an interface to be in PIM dense mode, sparse mode, or sparse-dense mode. The switch populates its multicast routing table and forwards multicast packets it receives from its directly connected LANs according to the mode setting. You must enable PIM in one of these modes for an interface to perform IP multicast routing. Enabling PIM on an interface also enables IGMP operation on that interface.

In populating the multicast routing table, dense-mode interfaces are always added to the table. Sparse-mode interfaces are added to the table only when periodic join messages are received from downstream devices or when there is a directly connected member on the interface. When forwarding from a LAN, sparse-mode operation occurs if there is an RP known for the group. If so, the packets are

encapsulated and sent toward the RP. When no RP is known, the packet is flooded in a dense-mode fashion. If the multicast traffic from a specific source is sufficient, the receiver's first-hop router might send join messages toward the source to build a source-based distribution tree.

By default, multicast routing is disabled, and there is no default mode setting. This procedure is required.

Beginning in privileged EXEC mode, follow these steps to enable IP multicasting, to configure a PIM version, and to configure a PIM mode. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip multicast-routing**	Enable IP multicast forwarding.
Step 3	**interface** *interface-id*	Specify the Layer 3 interface on which you want to enable multicast routing, and enter interface configuration mode. The specified interface must be one of the following: • A routed port: a physical port that has been configured as a Layer 3 port by entering the **no switchport** interface configuration command. • An SVI: a VLAN interface created by using the **interface vlan** *vlan-id* global configuration command. These ports must have IP addresses assigned to them. For more information, see the "Configuring Layer 3 Interfaces" section on page 9-19.
Step 4	**ip pim version** [**1** | **2**]	Configure the PIM version on the interface. By default, Version 2 is enabled and is the recommended setting. An interface in PIMv2 mode automatically downgrades to PIMv1 mode if that interface has a PIMv1 neighbor. The interface returns to Version 2 mode after all Version 1 neighbors are shut down or upgraded. For more information, see the "PIMv1 and PIMv2 Interoperability" section on page 3-9.
Step 5	**ip pim** {**dense-mode** | **sparse-mode** | **sparse-dense-mode**}	Enable a PIM mode on the interface. By default, no mode is configured. The keywords have these meanings: • **dense-mode**—Enables dense mode of operation. • **sparse-mode**—Enables sparse mode of operation. If you configure sparse-mode, you must also configure an RP. For more information, see the "Configuring a Rendezvous Point" section on page 3-12. • **sparse-dense-mode**—Causes the interface to be treated in the mode in which the group belongs. Sparse-dense-mode is the recommended setting.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable multicasting, use the **no ip multicast-routing** global configuration command. To return to the default PIM version, use the **no ip pim version** interface configuration command. To disable PIM on an interface, use the **no ip pim** interface configuration command.

Configuring a Rendezvous Point

You must have an RP if the interface is in sparse-dense mode and if you want to treat the group as a sparse group. You can use several methods, as described in these sections:

- Manually Assigning an RP to Multicast Groups, page 3-12
- Configuring Auto-RP, page 3-14 (a standalone, Cisco-proprietary protocol separate from PIMv1)
- Configuring PIMv2 BSR, page 3-18 (a standards track protocol in the Internet Engineering Task Force (IETF)

You can use Auto-RP, BSR, or a combination of both, depending on the PIM version you are running and the types of routers in your network. For more information, see the "PIMv1 and PIMv2 Interoperability" section on page 3-9 and the "Auto-RP and BSR Configuration Guidelines" section on page 3-10.

Manually Assigning an RP to Multicast Groups

This section explains how to manually configure an RP. If the RP for a group is learned through a dynamic mechanism (such as Auto-RP or BSR), you need not perform this task for that RP.

Senders of multicast traffic announce their existence through register messages received from the source's first-hop router (designated router) and forwarded to the RP. Receivers of multicast packets use RPs to join a multicast group by using explicit join messages. RPs are not members of the multicast group; rather, they serve as a *meeting place* for multicast sources and group members.

You can configure a single RP for multiple groups defined by an access list. If there is no RP configured for a group, the multilayer switch treats the group as dense and uses the dense-mode PIM techniques.

Beginning in privileged EXEC mode, follow these steps to manually configure the address of the RP. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip pim rp-address** *ip-address* [*access-list-number*] [**override**]	Configure the address of a PIM RP. By default, no PIM RP address is configured. You must configure the IP address of RPs on all routers and multilayer switches (including the RP). If there is no RP configured for a group, the multilayer switch treats the group as dense, using the dense-mode PIM techniques. A PIM device can be an RP for more than one group. Only one RP address can be used at a time within a PIM domain. The access-list conditions specify for which groups the device is an RP. • For *ip-address*, enter the unicast address of the RP in dotted-decimal notation. • (Optional) For *access-list-number*, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. • (Optional) The **override** keyword means that if there is a conflict between the RP configured with this command and one learned by Auto-RP or BSR, the RP configured with this command prevails.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the multicast group address for which the RP should be used. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an RP address, use the **no ip pim rp-address** *ip-address* [*access-list-number*] [**override**] global configuration command.

This example shows how to configure the address of the RP to 147.106.6.22 for multicast group 225.2.2.2 only:

```
Switch(config)# access-list 1 permit 225.2.2.2 0.0.0.0
Switch(config)# ip pim rp-address 147.106.6.22 1
```

Configuring Auto-RP

Auto-RP uses IP multicast to automate the distribution of group-to-RP mappings to all Cisco routers and multilayer switches in a PIM network. It has these benefits:

- It is easy to use multiple RPs within a network to serve different group ranges.
- It allows load splitting among different RPs and arrangement of RPs according to the location of group participants.
- It avoids inconsistent, manual RP configurations on every router and multilayer switch in a PIM network, which can cause connectivity problems.

Note If you configure PIM in sparse mode or sparse-dense mode and do not configure Auto-RP, you must manually configure an RP as described in the "Manually Assigning an RP to Multicast Groups" section on page 3-12.

Note If routed interfaces are configured in sparse mode, Auto-RP can still be used if all devices are configured with a manual RP address for the Auto-RP groups.

These sections describe how to configure Auto-RP:

- Setting up Auto-RP in a New Internetwork, page 3-14
- Adding Auto-RP to an Existing Sparse-Mode Cloud, page 3-14
- Preventing Join Messages to False RPs, page 3-16
- Filtering Incoming RP Announcement Messages, page 3-16

For overview information, see the "Auto-RP" section on page 3-5.

Setting up Auto-RP in a New Internetwork

If you are setting up Auto-RP in a new internetwork, you do not need a default RP because you configure all the interfaces for sparse-dense mode. Follow the process described in the section "Adding Auto-RP to an Existing Sparse-Mode Cloud" section on page 3-14. However, skip Step 3 to configure a PIM router as the RP for the local group.

Adding Auto-RP to an Existing Sparse-Mode Cloud

This section contains some suggestions for the initial deployment of Auto-RP into an existing sparse-mode cloud to minimize disruption of the existing multicast infrastructure.

Beginning in privileged EXEC mode, follow these steps to deploy Auto-RP in an existing sparse-mode cloud. This procedure is optional.

	Command	Purpose
Step 1	**show running-config**	Verify that a default RP is already configured on all PIM devices and the RP in the sparse-mode network. It was previously configured with the **ip pim rp-address** global configuration command. This step is not required for spare-dense-mode environments. The selected RP should have good connectivity and be available across the network. Use this RP for the global groups (for example 224.x.x.x and other global groups). Do not reconfigure the group address range that this RP serves. RPs dynamically discovered through Auto-RP take precedence over statically configured RPs. Assume that it is desirable to use a second RP for the local groups.
Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**ip pim send-rp-announce** *interface-id* **scope** *ttl* **group-list** *access-list-number* **interval** *seconds*	Configure another PIM device to be the candidate RP for local groups. • For *interface-id*, enter the interface type and number that identifies the RP address. Valid interfaces include physical ports, port channels, and VLANs. • For **scope** *ttl*, specify the time-to-live value in hops. Enter a hop count that is high enough so that the RP-announce messages reach all mapping agents in the network. There is no default setting. The range is 1 to 255. • For **group-list** *access-list-number*, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. • For **interval** *seconds*, specify how often the announcement messages must be sent. The default is 60 seconds. The range is 1 to 16383.
Step 4	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 3. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the multicast group address range for which the RP should be used. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.

	Command	Purpose
Step 5	**ip pim send-rp-discovery scope** *ttl*	Find a multilayer switch whose connectivity is not likely to be interrupted, and assign it the role of RP-mapping agent. For **scope** *ttl*, specify the time-to-live value in hops to limit the RP discovery packets. All devices within the hop count from the source device receive the Auto-RP discovery messages. These messages tell other devices which group-to-RP mapping to use to avoid conflicts (such as overlapping group-to-RP ranges). There is no default setting. The range is 1 to 255.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the PIM device configured as the candidate RP, use the **no ip pim send-rp-announce** *interface-id* global configuration command. To remove the multilayer switch as the RP-mapping agent, use the **no ip pim send-rp-discovery** global configuration command.

This example shows how to send RP announcements out all PIM-enabled interfaces for a maximum of 31 hops. The IP address of Gigabit Ethernet interface 0/1 is the RP. Access list 5 describes the group for which this multilayer switch serves as RP:

```
Switch(config)# ip pim send-rp-announce gigabitethernet0/1 scope 31 group-list 5
Switch(config)# access-list 5 permit 224.0.0.0 15.255.255.255
```

Preventing Join Messages to False RPs

Find whether the **ip pim accept-rp** command was previously configured throughout the network by using the **show running-config** privileged EXEC command. If the **ip pim accept-rp** command is not configured on any device, this problem can be addressed later. In those routers or multilayer switches already configured with the **ip pim accept-rp** command, you must enter the command again to accept the newly advertised RP.

To accept all RPs advertised with Auto-RP and reject all other RPs by default, use the **ip pim accept-rp auto-rp** global configuration command.

If all interfaces are in sparse mode, use a default-configured RP to support the two well-known groups 224.0.1.39 and 224.0.1.40. Auto-RP uses these two well-known groups to collect and distribute RP-mapping information. When this is the case and the **ip pim accept-rp auto-rp** command is configured, another **ip pim accept-rp** command accepting the RP must be configured as follows:

```
Switch(config)# ip pim accept-rp 172.10.20.1 1
Switch(config)# access-list 1 permit 224.0.1.39
Switch(config)# access-list 1 permit 224.0.1.40
```

Filtering Incoming RP Announcement Messages

You can add configuration commands to the mapping agents to prevent a maliciously configured router from masquerading as a candidate RP and causing problems.

Beginning in privileged EXEC mode, follow these steps to filter incoming RP announcement messages. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip pim rp-announce-filter rp-list** *access-list-number* **group-list** *access-list-number*	Filter incoming RP announcement messages. Enter this command on each mapping agent in the network. Without this command, all incoming RP-announce messages are accepted by default. For **rp-list** *access-list-number*, configure an access list of candidate RP addresses that, if permitted, is accepted for the group ranges supplied in the **group-list** *access-list-number* variable. If this variable is omitted, the filter applies to all multicast groups. If more than one mapping agent is used, the filters must be consistent across all mapping agents to ensure that no conflicts occur in the Group-to-RP mapping information.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • Create an access list that specifies from which routers and multilayer switches the mapping agent accepts candidate RP announcements (rp-list ACL). • Create an access list that specifies the range of multicast groups from which to accept or deny (group-list ACL). • For *source*, enter the multicast group address range for which the RP should be used. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a filter on incoming RP announcement messages, use the **no ip pim rp-announce-filter rp-list** *access-list-number* **group-list** *access-list-number* global configuration command.

This example shows a sample configuration on an Auto-RP mapping agent that is used to prevent candidate RP announcements from being accepted from unauthorized candidate RPs:

```
Switch(config)# ip pim rp-announce-filter rp-list 10 group-list 20
Switch(config)# access-list 10 permit host 172.16.5.1
Switch(config)# access-list 10 permit host 172.16.2.1
Switch(config)# access-list 20 deny 239.0.0.0 0.0.255.255
Switch(config)# access-list 20 permit 224.0.0.0 15.255.255.255
```

In this example, the mapping agent accepts candidate RP announcements from only two devices, 172.16.5.1 and 172.16.2.1. The mapping agent accepts candidate RP announcements from these two devices only for multicast groups that fall in the group range of 224.0.0.0 to 239.255.255.255. The mapping agent does not accept candidate RP announcements from any other devices in the network. Furthermore, the mapping agent does not accept candidate RP announcements from 172.16.5.1 or 172.16.2.1 if the announcements are for any groups in the 239.0.0.0 through 239.255.255.255 range. This range is the administratively scoped address range.

Configuring PIMv2 BSR

These sections describe how to set up BSR in your PIMv2 network:

These sections describe how to set up BSR in your PIMv2 network:

- Defining the PIM Domain Border, page 3-18 (optional)
- Defining the IP Multicast Boundary, page 3-19 (optional)
- Configuring Candidate BSRs, page 3-20 (optional)
- Configuring Candidate RPs, page 3-21 (optional)

For overview information, see the "Bootstrap Router" section on page 3-5.

Defining the PIM Domain Border

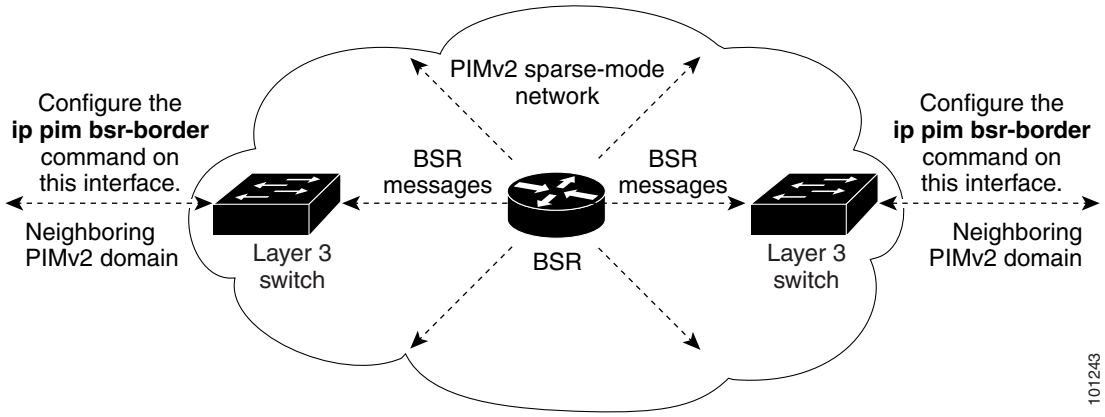
As IP multicast becomes more widespread, the chances of one PIMv2 domain bordering another PIMv2 domain is increasing. Because these two domains probably do not share the same set of RPs, BSR, candidate RPs, and candidate BSRs, you need to constrain PIMv2 BSR messages from flowing into or out of the domain. Allowing these messages to leak across the domain borders could adversely affect the normal BSR election mechanism and elect a single BSR across all bordering domains and co-mingle candidate RP advertisements, resulting in the election of RPs in the wrong domain.

Beginning in privileged EXEC mode, follow these steps to define the PIM domain border. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**ip pim bsr-border**	Define a PIM bootstrap message boundary for the PIM domain. Enter this command on each interface that connects to other bordering PIM domains. This command instructs the multilayer switch to neither send or receive PIMv2 BSR messages on this interface as shown in Figure 3-3.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the PIM border, use the **no ip pim bsr-border** interface configuration command.

Figure 3-3 Constraining PIMv2 BSR Messages



Defining the IP Multicast Boundary

You define a multicast boundary to prevent Auto-RP messages from entering the PIM domain. You create an access list to deny packets destined for 224.0.1.39 and 224.0.1.40, which carry Auto-RP information.

Beginning in privileged EXEC mode, follow these steps to define a multicast boundary. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list** *access-list-number* **deny** *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. • For *access-list-number*, the range is 1 to 99. • The **deny** keyword denies access if the conditions are matched. • For *source*, enter multicast addresses 224.0.1.39 and 224.0.1.40, which carry Auto-RP information. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 3	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 4	**ip multicast boundary** *access-list-number*	Configure the boundary, specifying the access list you created in Step 2.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the boundary, use the **no ip multicast boundary** interface configuration command.

This example shows a portion of an IP multicast boundary configuration that denies Auto-RP information:

```
Switch(config)# access-list 1 deny 224.0.1.39
Switch(config)# access-list 1 deny 224.0.1.40
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip multicast boundary 1
```

Configuring Candidate BSRs

You can configure one or more candidate BSRs. The devices serving as candidate BSRs should have good connectivity to other devices and be in the backbone portion of the network.

Beginning in privileged EXEC mode, follow these steps to configure your switch as a candidate BSR. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip pim bsr-candidate** *interface-id hash-mask-length* [*priority*]	Configure your multilayer switch to be a candidate BSR. • For *interface-id,* specify the interface on this switch from which the BSR address is derived to make it a candidate. This interface must be enabled with PIM. Valid interfaces include physical ports, port channels, and VLANs. • For *hash-mask-length*, specify the mask length (32 bits maximum) that is to be ANDed with the group address before the hash function is called. All groups with the same seed hash correspond to the same RP. For example, if this value is 24, only the first 24 bits of the group addresses matter. • (Optional) For *priority*, enter a number from 0 to 255. The BSR with the larger priority is preferred. If the priority values are the same, the device with the highest IP address is selected as the BSR. The default is 0.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove this device as a candidate BSR, use the **no ip pim bsr-candidate** global configuration command.

This example shows how to configure a candidate BSR, which uses the IP address 172.21.24.18 on a port as the advertised BSR address, uses 30 bits as the hash-mask-length, and has a priority of 10.

```
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# ip address 172.21.24.18 255.255.255.0
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# ip pim bsr-candidate gigabitethernet0/2 30 10
```

Configuring Candidate RPs

You can configure one or more candidate RPs. Similar to BSRs, the RPs should also have good connectivity to other devices and be in the backbone portion of the network. An RP can serve the entire IP multicast address space or a portion of it. Candidate RPs send candidate RP advertisements to the BSR. When deciding which devices should be RPs, consider these options:

- In a network of Cisco routers and multilayer switches where only Auto-RP is used, any device can be configured as an RP.
- In a network that includes only Cisco PIMv2 routers and multilayer switches and with routers from other vendors, any device can be used as an RP.
- In a network of Cisco PIMv1 routers, Cisco PIMv2 routers, and routers from other vendors, configure only Cisco PIMv2 routers and multilayer switches as RPs.

Beginning in privileged EXEC mode, follow these steps to configure your switch to advertise itself as a PIMv2 candidate RP to the BSR. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip pim rp-candidate** *interface-id* [**group-list** *access-list-number*]	Configure your multilayer switch to be a candidate RP. • For *interface-id,* specify the interface whose associated IP address is advertised as a candidate RP address. Valid interfaces include physical ports, port channels, and VLANs. • (Optional) For **group-list** *access-list-number*, enter an IP standard access list number from 1 to 99. If no group-list is specified, the multilayer switch is a candidate RP for all groups.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the number of the network or host from which the packet is being sent. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove this device as a candidate RP, use the **no ip pim rp-candidate** *interface-id* global configuration command.

This example shows how to configure the switch to advertise itself as a candidate RP to the BSR in its PIM domain. Standard access list number 4 specifies the group prefix associated with the RP that has the address identified by a port. That RP is responsible for the groups with the prefix 239.

```
Switch(config)# ip pim rp-candidate gigabitethernet0/2 group-list 4
Switch(config)# access-list 4 permit 239.0.0.0 0.255.255.255
```

Using Auto-RP and a BSR

If there are only Cisco devices in you network (no routers from other vendors), there is no need to configure a BSR. Configure Auto-RP in a network that is running both PIMv1 and PIMv2.

If you have non-Cisco PIMv2 routers that need to interoperate with Cisco PIMv1 routers and multilayer switches, both Auto-RP and a BSR are required. We recommend that a Cisco PIMv2 router or multilayer switch be both the Auto-RP mapping agent and the BSR.

If you must have one or more BSRs, we have these recommendations:

- Configure the candidate BSRs as the RP-mapping agents for Auto-RP. For more information, see the "Configuring Auto-RP" section on page 3-14 and the "Configuring Candidate BSRs" section on page 3-20.
- For group prefixes advertised through Auto-RP, the PIMv2 BSR mechanism should not advertise a subrange of these group prefixes served by a different set of RPs. In a mixed PIMv1 and PIMv2 domain, have backup RPs serve the same group prefixes. This prevents the PIMv2 DRs from selecting a different RP from those PIMv1 DRs, due to the longest match lookup in the RP-mapping database.

Beginning in privileged EXEC mode, follow these steps to verify the consistency of group-to-RP mappings. This procedure is optional.

	Command	Purpose
Step 1	**show ip pim rp** [[*group-name* | *group-address*] | **mapping**]	On any Cisco device, display the available RP mappings. • (Optional) For *group-name*, specify the name of the group about which to display RPs. • (Optional) For *group-address*, specify the address of the group about which to display RPs. • (Optional) Use the **mapping** keyword to display all group-to-RP mappings of which the Cisco device is aware (either configured or learned from Auto-RP).
Step 2	**show ip pim rp-hash** *group*	On a PIMv2 router or multilayer switch, confirm that the same RP is the one that a PIMv1 system chooses. For *group*, enter the group address for which to display RP information.

Monitoring the RP Mapping Information

To monitor the RP mapping information, use these commands in privileged EXEC mode:

- **show ip pim bsr** displays information about the elected BSR.
- **show ip pim rp-hash** *group* displays the RP that was selected for the specified group.
- **show ip pim rp** [*group-name* | *group-address* | **mapping**] displays how the multilayer switch learns of the RP (through the BSR or the Auto-RP mechanism).

Troubleshooting PIMv1 and PIMv2 Interoperability Problems

When debugging interoperability problems between PIMv1 and PIMv2, check these in the order shown:

1. Verify RP mapping with the **show ip pim rp-hash** privileged EXEC command, making sure that all systems agree on the same RP for the same group.
2. Verify interoperability between different versions of DRs and RPs. Make sure the RPs are interacting with the DRs properly (by responding with register-stops and forwarding decapsulated data packets from registers).

Configuring Advanced PIM Features

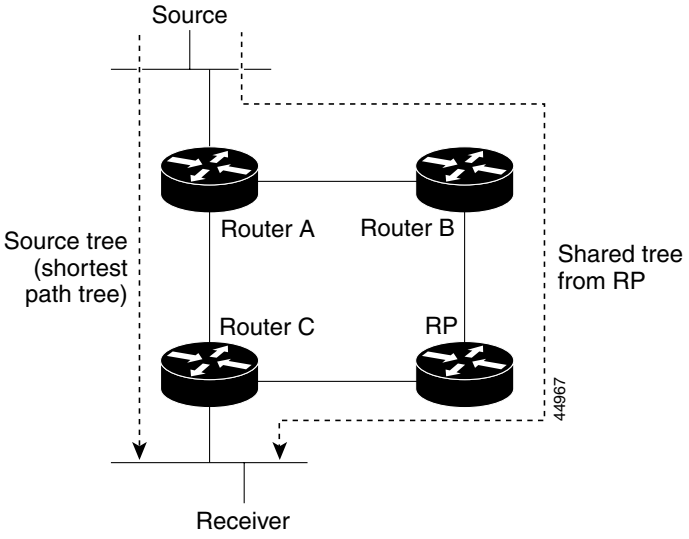
These sections describe the optional advanced PIM features:

- Understanding PIM Shared Tree and Source Tree, page 3-23
- Delaying the Use of PIM Shortest-Path Tree, page 3-25 (optional)
- Modifying the PIM Router-Query Message Interval, page 3-26 (optional)

Understanding PIM Shared Tree and Source Tree

By default, members of a group receive data from senders to the group across a single data-distribution tree rooted at the RP. Figure 3-4 shows this type of shared-distribution tree. Data from senders is delivered to the RP for distribution to group members joined to the shared tree.

Figure 3-4 Shared Tree and Source Tree (Shortest-Path Tree)



If the data rate warrants, leaf routers (routers without any downstream connections) on the shared tree can use the data distribution tree rooted at the source. This type of distribution tree is called a shortest-path tree or source tree. By default, the software switches to a source tree upon receiving the first data packet from a source.

This process describes the move from a shared tree to a source tree:

1. A receiver joins a group; leaf Router C sends a join message toward the RP.
2. The RP puts a link to Router C in its outgoing interface list.
3. A source sends data; Router A encapsulates the data in a register message and sends it to the RP.
4. The RP forwards the data down the shared tree to Router C and sends a join message toward the source. At this point, data might arrive twice at Router C, once encapsulated and once natively.
5. When data arrives natively (unencapsulated) at the RP, it sends a register-stop message to Router A.
6. By default, reception of the first data packet prompts Router C to send a join message toward the source.
7. When Router C receives data on (S,G), it sends a prune message for the source up the shared tree.
8. The RP deletes the link to Router C from the outgoing interface of (S,G). The RP triggers a prune message toward the source.

Join and prune messages are sent for sources and RPs. They are sent hop-by-hop and are processed by each PIM device along the path to the source or RP. Register and register-stop messages are not sent hop-by-hop. They are sent by the designated router that is directly connected to a source and are received by the RP for the group.

Multiple sources sending to groups use the shared tree.

You can configure the PIM device to stay on the shared tree. For more information, see the "Delaying the Use of PIM Shortest-Path Tree" section on page 3-25.

Delaying the Use of PIM Shortest-Path Tree

The change from shared to source tree happens when the first data packet arrives at the last-hop router (Router C in Figure 3-4). This change occurs because the **ip pim spt-threshold** interface configuration command controls that timing; its default setting is 0 kbps.

The shortest-path tree requires more memory than the shared tree but reduces delay. You might want to postpone its use. Instead of allowing the leaf router to immediately move to the shortest-path tree, you can specify that the traffic must first reach a threshold.

You can configure when a PIM leaf router should join the shortest-path tree for a specified group. If a source sends at a rate greater than or equal to the specified kbps rate, the multilayer switch triggers a PIM join message toward the source to construct a source tree (shortest-path tree). If the traffic rate from the source drops below the threshold value, the leaf router switches back to the shared tree and sends a prune message toward the source.

You can specify to which groups the shortest-path tree threshold applies by using a group list (a standard access list). If a value of 0 is specified or if the group list is not used, the threshold applies to all groups.

Beginning in privileged EXEC mode, follow these steps to configure a traffic rate threshold that must be reached before multicast routing is switched from the source tree to the shortest-path tree. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list. • For *access-list-number*, the range is 1 to 99. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, specify the multicast group to which the threshold will apply. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 3	**ip pim spt-threshold** {*kbps* | **infinity**} [**group-list** *access-list-number*]	Specify the threshold that must be reached before moving to shortest-path tree (spt). • For *kbps*, specify the traffic rate in kilobits per second. The default is 0 kbps. The range is 0 to 4294967. • Specify **infinity** if you want all sources for the specified group to use the shared tree, never switching to the source tree. • (Optional) For **group-list** *access-list-number*, specify the access list created in Step 2. If the value is 0 or if the group-list is not used, the threshold applies to all groups.
Step 4	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default threshold, use the **no ip pim spt-threshold** {*kbps* | **infinity**} global configuration command.

Modifying the PIM Router-Query Message Interval

PIM routers and multilayer switches send PIM router-query messages to find which device will be the DR for each LAN segment (subnet). The DR is responsible for sending IGMP host-query messages to all hosts on the directly connected LAN.

With PIM DM operation, the DR has meaning only if IGMPv1 is in use. IGMPv1 does not have an IGMP querier election process, so the elected DR functions as the IGMP querier. With PIM SM operation, the DR is the device that is directly connected to the multicast source. It sends PIM register messages to notify the RP that multicast traffic from a source needs to be forwarded down the shared tree. In this case, the DR is the device with the highest IP address.

Beginning in privileged EXEC mode, follow these steps to modify the router-query message interval. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**ip pim query-interval** *seconds*	Configure the frequency at which the multilayer switch sends PIM router-query messages. The default is 30 seconds. The range is 1 to 65535.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip igmp interface** [*interface-id*]	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default interval, use the **no ip pim query-interval** [*seconds*] interface configuration command.

Configuring Optional IGMP Features

These sections describe how to configure optional IGMP features:

- Default IGMP Configuration, page 3-27
- Configuring the Multilayer Switch as a Member of a Group, page 3-27 (optional)
- Controlling Access to IP Multicast Groups, page 3-28 (optional)
- Changing the IGMP Version, page 3-29 (optional)
- Modifying the IGMP Host-Query Message Interval, page 3-29 (optional)

- Changing the IGMP Query Timeout for IGMPv2, page 3-30 (optional)
- Changing the Maximum Query Response Time for IGMPv2, page 3-31 (optional)
- Configuring the Multilayer Switch as a Statically Connected Member, page 3-31 (optional)

Default IGMP Configuration

Table 3-3 shows the default IGMP configuration.

Table 3-3 Default IGMP Configuration

Feature	Default Setting
Multilayer switch as a member of a multicast group	No group memberships are defined.
Access to multicast groups	All groups are allowed on an interface.
IGMP version	Version 2 on all interfaces.
IGMP host-query message interval	60 seconds on all interfaces.
IGMP query timeout	60 seconds on all interfaces.
IGMP maximum query response time	10 seconds on all interfaces.
Multilayer switch as a statically connected member	Disabled.

Configuring the Multilayer Switch as a Member of a Group

You can configure the switch as a member of a multicast group and discover multicast reachability in a network. If all the multicast-capable routers and multilayer switches that you administer are members of a multicast group, pinging that group causes all these devices to respond. The devices respond to ICMP echo-request packets addressed to a group of which they are members. Another example is the multicast trace-route tools provided in the software.

Caution Performing this procedure might impact the CPU performance because the CPU will receive all data traffic for the group address.

Beginning in privileged EXEC mode, follow these steps to configure the switch to be a member of a group. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**ip igmp join-group** *group-address*	Configure the switch to join a multicast group. By default, no group memberships are defined. For *group-address*, specify the multicast IP address in dotted decimal notation.
Step 4	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 5	**show ip igmp interface** [*interface-id*]	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To cancel membership in a group, use the **no ip igmp join-group** *group-address* interface configuration command.

This example shows how to allow the switch to join multicast group 255.2.2.2:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip igmp join-group 255.2.2.2
```

Controlling Access to IP Multicast Groups

The switch sends IGMP host-query messages to find which multicast groups have members on attached local networks. The switch then forwards to these group members all packets addressed to the multicast group. You can place a filter on each interface to restrict the multicast groups that hosts on the subnet serviced by the interface can join.

Beginning in privileged EXEC mode, follow these steps to filter multicast groups allowed on an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**ip igmp access-group** *access-list-number*	Specify the multicast groups that hosts on the subnet serviced by an interface can join. By default, all groups are allowed on an interface. For *access-list-number*, specify an IP standard access list number. The range is 1 to 99.
Step 4	**exit**	Return to global configuration mode.
Step 5	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list. • For *access-list-number*, specify the access list created in Step 3. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, specify the multicast group that hosts on the subnet can join. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 6	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 7	**show ip igmp interface** [*interface-id*]	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable groups on an interface, use the **no ip igmp access-group** *access-list-number* interface configuration command.

This example shows how to configure hosts attached to an interface as able to join only group 255.2.2.2:

```
Switch(config)# access-list 1 255.2.2.2 0.0.0.0
Switch(config-if)# interface gigabitethernet0/1
Switch(config-if)# ip igmp access-group 1
```

Changing the IGMP Version

By default, the multilayer switch uses IGMP Version 2, which allows features such as the IGMP query timeout and the maximum query response time.

All systems on the subnet must support the same version. The switch does not automatically detect Version 1 systems and switch to Version 1. You can mix Version 1 and Version 2 hosts on the subnet because Version 2 routers or switches always work correctly with IGMPv1 hosts.

Configure the switch for Version 1 if your hosts do not support Version 2.

Beginning in privileged EXEC mode, follow these steps to change the IGMP version. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**ip igmp version** {**1** | **2**}	Specify the IGMP version that the switch uses. **Note** If you change to Version 1, you cannot configure the **ip igmp query-interval** or the **ip igmp query-max-response-time** interface configuration commands.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip igmp interface** [*interface-id*]	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip igmp version** interface configuration command.

Modifying the IGMP Host-Query Message Interval

The switch periodically sends IGMP host-query messages to discover which multicast groups are present on attached networks. These messages are sent to the all-hosts multicast group (224.0.0.1) with a time-to-live (TTL) of 1. The switch sends host-query messages to refresh its knowledge of memberships present on the network. If, after some number of queries, the software discovers that no local hosts are members of a multicast group, the software stops forwarding multicast packets to the local network from remote origins for that group and sends a prune message upstream toward the source.

The switch elects a PIM designated router (DR) for the LAN (subnet). The DR is the router or multilayer switch with the highest IP address for IGMPv2. For IGMPv1, the DR is elected according to the multicast routing protocol that runs on the LAN. The designated router is responsible for sending IGMP host-query messages to all hosts on the LAN. In sparse mode, the designated router also sends PIM register and PIM join messages toward the RP router.

Beginning in privileged EXEC mode, follow these steps to modify the host-query interval. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**ip igmp query-interval** *seconds*	Configure the frequency at which the designated router sends IGMP host-query messages. By default, the designated router sends IGMP host-query messages every 60 seconds to keep the IGMP overhead very low on hosts and networks. The range is 1 to 18000.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip igmp interface** [*interface-id*]	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default frequency, use the **no ip igmp query-interval** interface configuration command.

Changing the IGMP Query Timeout for IGMPv2

If you are using IGMPv2, you can specify the period of time before the switch takes over as the querier for the interface. By default, the switch waits twice the query interval controlled by the **ip igmp query-interval** interface configuration command. After that time, if the switch has received no queries, it becomes the querier.

You can configure the query interval by entering the **show ip igmp interface** *interface-id* privileged EXEC command.

Beginning in privileged EXEC mode, follow these steps to change the IGMP query timeout. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**ip igmp querier-timeout** *seconds*	Specify the IGMP query timeout. The default is 60 seconds (twice the query interval). The range is 60 to 300.
Step 4	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 5	**show ip igmp interface** [*interface-id*]	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default timeout value, use the **no ip igmp query-timeout** interface configuration command.

Changing the Maximum Query Response Time for IGMPv2

If you are using IGMPv2, you can change the maximum query response time advertised in IGMP queries. The maximum query response time enables the switch to quickly detect that there are no more directly connected group members on a LAN. Decreasing the value enables the switch to prune groups faster.

Beginning in privileged EXEC mode, follow these steps to change the maximum query response time. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**ip igmp query-max-response-time** *seconds*	Change the maximum query response time advertised in IGMP queries. The default is 10 seconds. The range is 1 to 25.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip igmp interface** [*interface-id*]	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default query-response time, use the **no ip igmp query-max-response-time** interface configuration command.

Configuring the Multilayer Switch as a Statically Connected Member

Sometimes there is either no group member on a network segment or a host cannot report its group membership by using IGMP. However, you might want multicast traffic to go to that network segment. These are ways to pull multicast traffic down to a network segment:

- Use the **ip igmp join-group** interface configuration command. With this method, the switch accepts the multicast packets in addition to forwarding them. Accepting the multicast packets prevents the switch from fast switching.
- Use the **ip igmp static-group** interface configuration command. With this method, the switch does not accept the packets itself, but only forwards them. This method enables fast switching. The outgoing interface appears in the IGMP cache, but the switch itself is not a member, as evidenced by lack of an *L* (local) flag in the multicast route entry.

Beginning in privileged EXEC mode, follow these steps to configure the switch itself to be a statically connected member of a group (and enable fast switching). This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**ip igmp static-group** *group-address*	Configure the switch as a statically connected member of a group. By default, this feature is disabled.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show ip igmp interface** [*interface-id*]	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the switch as a member of the group, use the **no ip igmp static-group** interface configuration command.

Configuring Optional Multicast Routing Features

This section describes how to configure optional multicast routing features, which are grouped as follows:

- Features for Layer 2 connectivity and MBONE multimedia conference session and set up:
  - Enabling CGMP Server Support, page 3-32 (optional)
  - Configuring sdr Listener Support, page 3-33 (optional)
- Features that control bandwidth utilization:
  - Configuring the TTL Threshold, page 3-34 (optional)
  - Configuring an IP Multicast Boundary, page 3-36 (optional)

Enabling CGMP Server Support

The switch serves as a CGMP server for devices that do not support IGMP snooping but have CGMP client functionality. CGMP is a protocol used on Cisco routers and multilayer switches connected to Layer 2 Catalyst switches to perform tasks similar to those performed by IGMP. CGMP is necessary because the Layer 2 switch cannot distinguish between IP multicast data packets and IGMP report messages, which are both at the MAC-level and are addressed to the same group address.

Beginning in privileged EXEC mode, follow these steps to enable the CGMP server on the switch interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface that is connected to the Layer 2 Catalyst switch, and enter interface configuration mode.

	Command	Purpose
Step 3	**ip cgmp** [**proxy**]	Enable CGMP on the interface. By default, CGMP is disabled on all interfaces. Enabling CGMP triggers a CGMP join message. Enable CGMP only on Layer 3 interfaces connected to Layer 2 Catalyst switches. (Optional) When you enter the **proxy** keyword, the CGMP proxy function is enabled. The proxy router advertises the existence of non-CGMP-capable routers by sending a CGMP join message with the non-CGMP-capable router MAC address and a group address of 0000.0000.0000. **Note** To perform CGMP proxy, the multilayer switch must be the IGMP querier. If you configure the **ip cgmp proxy** command, you must manipulate the IP addresses so that the switch is the IGMP querier, which might be the highest or lowest IP address, depending on which version of IGMP is running on the network. An IGMP Version 2 querier is selected based on the lowest IP address on the interface. An IGMP Version 1 querier is selected based on the multicast routing protocol used on the interface.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.
Step 7		Verify the Layer 2 Catalyst switch CGMP-client configuration. For more information, see the documentation that shipped with the product.

To disable CGMP on the interface, use the **no ip cgmp** interface configuration command.

When multiple Cisco CGMP-capable devices are connected to a switched network and the **ip cgmp proxy** command is needed, we recommend that all devices be configured with the same CGMP option and have precedence for becoming the IGMP querier over non-Cisco routers.

Configuring sdr Listener Support

The MBONE is the small subset of Internet routers and hosts that are interconnected and capable of forwarding IP multicast traffic. Other interesting multimedia content is often broadcast over the MBONE. Before you can join a multimedia session, you need to know what multicast group address and port are being used for the session, when the session is going to be active, and what sort of applications (audio, video, and so forth) are required on your workstation. The MBONE Session Directory Version 2 (sdr) tool provides this information. This freeware application can be downloaded from several sites on the World Wide Web, one of which is http://www.video.ja.net/mice/index.html.

SDR is a multicast application that listens to a well-known multicast group address and port for Session Announcement Protocol (SAP) multicast packets from SAP clients, which announce their conference sessions. These SAP packets contain a session description, the time the session is active, its IP multicast group addresses, media format, contact person, and other information about the advertised multimedia session. The information in the SAP packet is displayed in the SDR Session Announcement window.

Enabling sdr Listener Support

By default, the multilayer switch does not listen to session directory advertisements.

Beginning in privileged EXEC mode, follow these steps to enable the switch to join the default session directory group (224.2.127.254) on the interface and listen to session directory advertisements. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be enabled for sdr, and enter interface configuration mode.
Step 3	**ip sdr listen**	Enable sdr listener support.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable sdr support, use the **no ip sdr listen** interface configuration command.

Limiting How Long an sdr Cache Entry Exists

By default, entries are never deleted from the sdr cache. You can limit how long the entry remains active so that if a source stops advertising SAP information, old advertisements are not needlessly kept.

Beginning in privileged EXEC mode, follow these steps to limit how long an sdr cache entry stays active in the cache. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip sdr cache-timeout** *minutes*	Limit how long an sdr cache entry stays active in the cache. By default, entries are never deleted from the cache. For *minutes*, specify a number from 1 to 4294967295.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip sdr cache-timeout** global configuration command. To delete the entire cache, use the **clear ip sdr** privileged EXEC command.

To display the session directory cache, use the **show ip sdr** privileged EXEC command.

Configuring the TTL Threshold

Each time an IP multicast packet is forwarded by the multilayer switch, the time-to-live (TTL) value in the IP header is decremented by one. If the packet TTL decrements to zero, the switch drops the packet. TTL thresholds can be applied to individual interfaces of the multilayer switch to prevent multicast

packets with a TTL less than the TTL threshold from being forwarded out the interface. TTL thresholds provide a simple method to prevent the forwarding of multicast traffic beyond the boundary of a site or region, based on the TTL field in a multicast packet. This is known as TTL scoping.

Figure 3-5 shows a multicast packet arriving on Gigabit Ethernet interface 0/2 with a TTL value of 24. Assuming that the RPF check succeeds and that Gigabit Ethernet interfaces 0/1, 0/3, and 0/4 are all in the outgoing interface list, the packet would normally be forwarded out these interfaces. Because some TTL thresholds have been applied to these interfaces, the multilayer switch makes sure that the packet TTL value, which is decremented by 1 to 23, is greater than or equal to the interface TTL threshold before forwarding the packet out the interface. In this example, the packet is forwarded out interfaces 0/1 and 0/4, but not interface 0/3.

Figure 3-5 TTL Thresholds

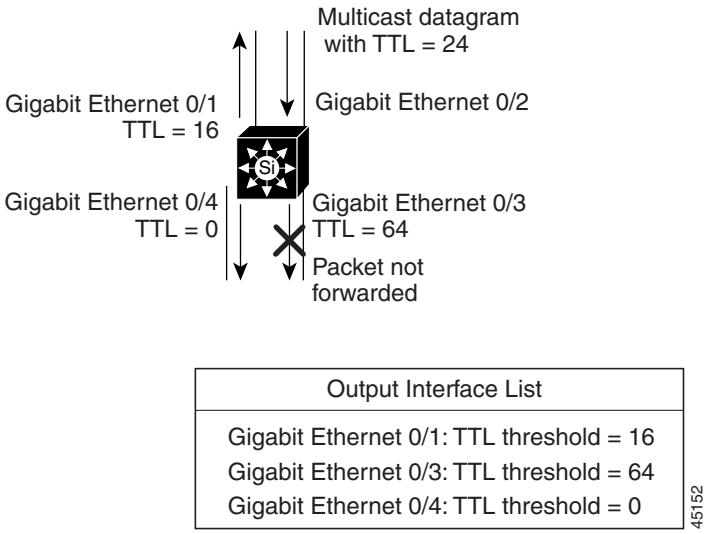
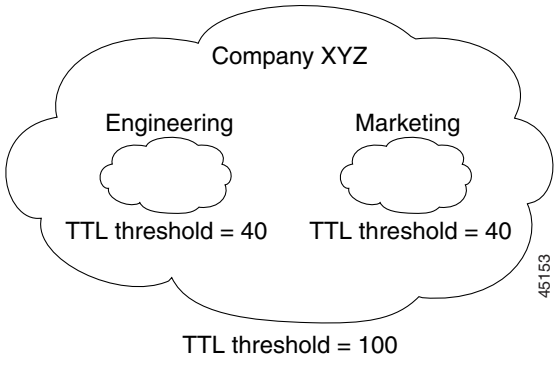


Figure 3-6 shows an example of TTL threshold boundaries being used to limit the forwarding of multicast traffic. Company XYZ has set a TTL threshold of 100 on all routed interfaces at the perimeter of its network. Multicast applications that constrain traffic to within the company's network need to send multicast packets with an initial TTL value set to 99. The engineering and marketing departments have set a TTL threshold of 40 at the perimeter of their networks; therefore, multicast applications running on these networks can prevent their multicast transmissions from leaving their respective networks.

Figure 3-6 TTL Boundaries



Beginning in privileged EXEC mode, follow these steps to change the default TTL threshold value. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 3	**ip multicast ttl-threshold** *ttl-value*	Configure the TTL threshold of packets being forwarded out an interface. The default TTL value is 0 hops, which means that all multicast packets are forwarded out the interface. The range is 0 to 255. Only multicast packets with a TTL value greater than the threshold are forwarded out the interface. You should configure the TTL threshold only on routed interfaces at the perimeter of the network.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

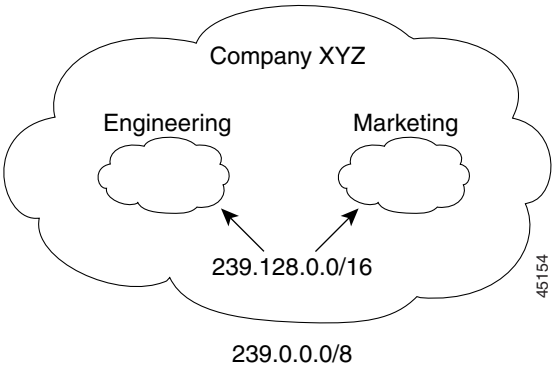
To return to the default TTL setting, use the **no ip multicast ttl-threshold** interface configuration command.

Configuring an IP Multicast Boundary

Administratively-scoped boundaries can be used to limit the forwarding of multicast traffic outside of a domain or subdomain. This approach uses a special range of multicast addresses, called *administratively-scoped addresses*, as the boundary mechanism. If you configure an administratively-scoped boundary on a routed interface, multicast traffic whose multicast group addresses fall in this range can not enter or exit this interface, thereby providing a firewall for multicast traffic in this address range.

Figure 3-7 shows that Company XYZ has an administratively-scoped boundary set for the multicast address range 239.0.0.0/8 on all routed interfaces at the perimeter of its network. This boundary prevents any multicast traffic in the range 239.0.0.0 through 239.255.255.255 from entering or leaving the network. Similarly, the engineering and marketing departments have an administratively-scoped boundary of 239.128.0.0/16 around the perimeter of their networks. This boundary prevents multicast traffic in the range of 239.128.0.0 through 239.128.255.255 from entering or leaving their respective networks.

Figure 3-7 Administratively-Scoped Boundaries



You can define an administratively-scoped boundary on a routed interface for multicast group addresses. A standard access list defines the range of addresses affected. When a boundary is defined, no multicast data packets are allowed to flow across the boundary from either direction. The boundary allows the same multicast group address to be reused in different administrative domains.

The IANA has designated the multicast address range 239.0.0.0 to 239.255.255.255 as the administratively-scoped addresses. This range of addresses can then be reused in domains administered by different organizations. The addresses would be considered local, not globally unique.

Beginning in privileged EXEC mode, follow these steps to set up an administratively-scoped boundary. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. • For *access-list-number*, the range is 1 to 99. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the number of the network or host from which the packet is being sent. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 3	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
Step 4	**ip multicast boundary** *access-list-number*	Configure the boundary, specifying the access list you created in Step 2.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the boundary, use the **no ip multicast boundary** interface configuration command.

This example shows how to set up a boundary for all administratively-scoped addresses:

```
Switch(config)# access-list 1 deny 239.0.0.0 0.255.255.255
Switch(config)# access-list 1 permit 224.0.0.0 15.255.255.255
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip multicast boundary 1
```

Configuring Basic DVMRP Interoperability Features

These sections describe how to perform basic configuration tasks on your multilayer switch to interoperate with DVMRP devices:

- Configuring DVMRP Interoperability, page 3-38 (optional)
- Configuring a DVMRP Tunnel, page 3-40 (optional)
- Advertising Network 0.0.0.0 to DVMRP Neighbors, page 3-42 (optional)
- Responding to mrinfo Requests, page 3-43 (optional)

For more advanced DVMRP features, see the "Configuring Advanced DVMRP Interoperability Features" section on page 3-43.

Configuring DVMRP Interoperability

Cisco multicast routers and multilayer switches using PIM can interoperate with non-Cisco multicast routers that use the DVMRP.

PIM devices dynamically discover DVMRP multicast routers on attached networks by listening to DVMR probe messages. When a DVMRP neighbor has been discovered, the PIM device periodically sends DVMRP report messages advertising the unicast sources reachable in the PIM domain. By default, directly connected subnets and networks are advertised. The device forwards multicast packets that have been forwarded by DVMRP routers and, in turn, forwards multicast packets to DVMRP routers.

You can configure an access list on the PIM routed interface connected to the MBONE to limit the number of unicast routes that are advertised in DVMRP route reports. Otherwise, all routes in the unicast routing table are advertised.

Note The mrouted protocol is a public-domain implementation of DVMRP. You must use mrouted Version 3.8 (which implements a nonpruning version of DVMRP) when Cisco routers and multilayer switches are directly connected to DVMRP routers or interoperate with DVMRP routers over an MBONE tunnel. DVMRP advertisements produced by the Cisco IOS software can cause older versions of the mrouted protocol to corrupt their routing tables and those of their neighbors.

You can configure what sources are advertised and what metrics are used by configuring the **ip dvmrp metric** interface configuration command. You can also direct all sources learned through a particular unicast routing process to be advertised into DVMRP.

Beginning in privileged EXEC mode, follow these steps to configure the sources that are advertised and the metrics that are used when DVMRP route-report messages are sent. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. • For *access-list-number*, the range is 1 to 99. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the number of the network or host from which the packet is being sent. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 3	**interface** *interface-id*	Specify the interface connected to the MBONE and enabled for multicast routing, and enter interface configuration mode.
Step 4	**ip dvmrp metric** *metric* [**list** *access-list-number*] [[*protocol process-id*] | [**dvmrp**]]	Configure the metric associated with a set of destinations for DVMRP reports. • For *metric*, the range is 0 to 32. A value of 0 means that the route is not advertised. A value of 32 is equivalent to infinity (unreachable). • (Optional) For **list** *access-list-number*, enter the access list number created in Step 2. If specified, only the multicast destinations that match the access list are reported with the configured metric. • (Optional) For *protocol process-id*, enter the name of the unicast routing protocol, such as **eigrp**, **igrp**, **ospf**, **rip**, **static**, or **dvmrp**, and the process ID number of the routing protocol. If specified, only routes learned by the specified routing protocol are advertised in DVMRP report messages. • (Optional) If specified, the **dvmrp** keyword allows routes from the DVMRP routing table to be advertised with the configured *metric* or filtered.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the metric or route map, use the **no ip dvmrp metric** *metric* [**list** *access-list-number*] [[*protocol process-id*] | [**dvmrp**]] or the **no ip dvmrp metric** *metric* **route-map** *map-name* interface configuration command.

A more sophisticated way to achieve the same results as the preceding command is to use a route map (**ip dvmrp metric** *metric* **route-map** *map-name* interface configuration command) instead of an access list. You subject unicast routes to route-map conditions before they are injected into DVMRP.

This example shows how to configure DVMRP interoperability when the PIM device and the DVMRP router are on the same network segment. In this example, access list 1 advertises the networks (198.92.35.0, 198.92.36.0, 198.92.37.0, 131.108.0.0, and 150.136.0.0) to the DVMRP router, and access list 2 prevents all other networks from being advertised (**ip dvmrp metric 0** interface configuration command).

```
Switch(config-if)# interface gigabitethernet0/1
Switch(config-if)# ip address 131.119.244.244 255.255.255.0
Switch(config-if)# ip pim dense-mode
Switch(config-if)# ip dvmrp metric 1 list 1
Switch(config-if)# ip dvmrp metric 0 list 2
Switch(config-if)# exit
Switch(config)# access-list 1 permit 198.92.35.0 0.0.0.255
Switch(config)# access-list 1 permit 198.92.36.0 0.0.0.255
Switch(config)# access-list 1 permit 198.92.37.0 0.0.0.255
Switch(config)# access-list 1 permit 131.108.0.0 0.0.255.255
Switch(config)# access-list 1 permit 150.136.0.0 0.0.255.255
Switch(config)# access-list 1 deny   0.0.0.0 255.255.255.255
Switch(config)# access-list 2 permit 0.0.0.0 255.255.255.255
```

Configuring a DVMRP Tunnel

The software supports DVMRP tunnels to the MBONE. You can configure a DVMRP tunnel on a router or multilayer switch if the other end is running DVMRP. The software then sends and receives multicast packets through the tunnel. This strategy allows a PIM domain to connect to the DVMRP router when all routers on the path do not support multicast routing. You cannot configure a DVMRP tunnel between two routers.

When a Cisco router or multilayer switch runs DVMRP through a tunnel, it advertises sources in DVMRP report messages, much as it does on real networks. The software also caches DVMRP report messages it receives and uses them in its Reverse Path Forwarding (RPF) calculation. This behavior allows the software to forward multicast packets received through the tunnel.

When you configure a DVMRP tunnel, you should assign an IP address to a tunnel in these cases:

- To send IP packets through the tunnel
- To configure the software to perform DVMRP summarization

The software does not advertise subnets through the tunnel if the tunnel has a different network number from the subnet. In this case, the software advertises only the network number through the tunnel.

Beginning in privileged EXEC mode, follow these steps to configure a DVMRP tunnel. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. • For *access-list-number*, the range is 1 to 99. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the number of the network or host from which the packet is being sent. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 3	**interface tunnel** *number*	Specify a tunnel interface, and enter interface configuration mode.
Step 4	**tunnel source** *ip-address*	Specify the source address of the tunnel interface. Enter the IP address of the interface on the multilayer switch.
Step 5	**tunnel destination** *ip-address*	Specify the destination address of the tunnel interface. Enter the IP address of the mrouted router.
Step 6	**tunnel mode dvmrp**	Configure the encapsulation mode for the tunnel to DVMRP.
Step 7	**ip address** *address mask* or **ip unnumbered** *type number*	Assign an IP address to the interface. or Configure the interface as unnumbered.
Step 8	**ip pim** [**dense-mode** | **sparse-mode**]	Configure the PIM mode on the interface.
Step 9	**ip dvmrp accept-filter** *access-list-number* [*distance*] **neighbor-list** *access-list-number*	Configure an acceptance filter for incoming DVMRP reports. By default, all destination reports are accepted with a distance of 0. Reports from all neighbors are accepted. • For *access-list-number*, specify the access list number created in Step 2. Any sources that match the access list are stored in the DVMRP routing table with distance. • (Optional) For *distance*, enter the administrative distance to the destination. By default, the administrative distance for DVMRP routes is 0 and take precedence over unicast routing table routes. If you have two paths to a source, one through unicast routing (using PIM as the multicast routing protocol) and another using DVMRP, and if you want to use the PIM path, increase the administrative distance for DVMRP routes. The range is 1 to 255. • For **neighbor-list** *access-list-number*, enter the number of the neighbor list created in Step 2. DVMRP reports are accepted only by those neighbors on the list.
Step 10	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 11	**show running-config**	Verify your entries.
Step 12	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the filter, use the **no ip dvmrp accept-filter** *access-list-number* [*distance*] **neighbor-list** *access-list-number* interface configuration command.

This example shows how to configure a DVMRP tunnel. In this configuration, the IP address of the tunnel on the Cisco multilayer switch is assigned *unnumbered*, which causes the tunnel to appear to have the same IP address as port 1. The tunnel endpoint source address is 172.16.2.1, and the tunnel endpoint address of the remote DVMRP router to which the tunnel is connected is 192.168.1.10. Any packets sent through the tunnel are encapsulated in an outer IP header. The Cisco multilayer switch is configured to accept incoming DVMRP reports with a distance of 100 from 198.92.37.0 through 198.92.37.255.

```
Switch(config)# ip multicast-routing
Switch(config)# interface tunnel 0
Switch(config-if)# ip unnumbered gigabitethernet 0/1
Switch(config-if)# ip pim dense-mode
Switch(config-if)# tunnel source gigabitethernet 0/1
Switch(config-if)# tunnel destination 192.168.1.10
Switch(config-if)# tunnel mode dvmrp
Switch(config-if)# ip dvmrp accept-filter 1 100
Switch(config-if)# interface gigabitethernet 0/1
Switch(config-if)# ip address 172.16.2.1 255.255.255.0
Switch(config-if)# ip pim dense-mode
Switch(config)# exit
Switch(config)# access-list 1 permit 198.92.37.0 0.0.0.255
```

Advertising Network 0.0.0.0 to DVMRP Neighbors

If your multilayer switch is a neighbor of an mrouted Version 3.6 device, you can configure the software to advertise network 0.0.0.0 (the default route) to the DVMRP neighbor. The DVMRP default route computes the RPF information for any multicast sources that do not match a more specific route.

Do not advertise the DVMRP default into the MBONE.

Beginning in privileged EXEC mode, follow these steps to advertise network 0.0.0.0 to DVMRP neighbors on an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface that is connected to the DVMRP router, and enter interface configuration mode.
Step 3	**ip dvmrp default-information** {**originate** | **only**}	Advertise network 0.0.0.0 to DVMRP neighbors. Use this command only when the multilayer switch is a neighbor of mrouted Version 3.6 machines. The keywords have these meanings: • **originate**—Specifies that other routes more specific than 0.0.0.0 can also be advertised. • **only**—Specifies that no DVMRP routes other than 0.0.0.0 are advertised.

	Command	Purpose
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To prevent the default route advertisement, use the **no ip dvmrp default-information** interface configuration command.

Responding to mrinfo Requests

The software answers mrinfo requests sent by mrouted systems and Cisco routers and multilayer switches. The software returns information about neighbors through DVMRP tunnels and all the routed interfaces. This information includes the metric (always set to 1), the configured TTL threshold, the status of the interface, and various flags. You can also use the **mrinfo** privileged EXEC command to query the router or switch itself, as in this example:

```
Switch# mrinfo
  171.69.214.27 (mm1-7kd.cisco.com) [version cisco 11.1] [flags: PMS]:
  171.69.214.27 -> 171.69.214.26 (mm1-r7kb.cisco.com) [1/0/pim/querier]
  171.69.214.27 -> 171.69.214.25 (mm1-45a.cisco.com) [1/0/pim/querier]
  171.69.214.33 -> 171.69.214.34 (mm1-45c.cisco.com) [1/0/pim]
  171.69.214.137 -> 0.0.0.0 [1/0/pim/querier/down/leaf]
  171.69.214.203 -> 0.0.0.0 [1/0/pim/querier/down/leaf]
  171.69.214.18 -> 171.69.214.20 (mm1-45e.cisco.com) [1/0/pim]
  171.69.214.18 -> 171.69.214.19 (mm1-45c.cisco.com) [1/0/pim]
  171.69.214.18 -> 171.69.214.17 (mm1-45a.cisco.com) [1/0/pim]
```

Configuring Advanced DVMRP Interoperability Features

Cisco routers and multilayer switches run PIM to forward multicast packets to receivers and receive multicast packets from senders. It is also possible to propagate DVMRP routes into and through a PIM cloud. PIM uses this information; however, Cisco routers and multilayer switches do not implement DVMRP to forward multicast packets.

These sections describe how to perform advanced optional configuration tasks on your multilayer switch to interoperate with DVMRP devices:

- Enabling DVMRP Unicast Routing, page 3-43 (optional)
- Rejecting a DVMRP Nonpruning Neighbor, page 3-44 (optional)
- Controlling Route Exchanges, page 3-46 (optional)

For information on basic DVMRP features, see the "Configuring Basic DVMRP Interoperability Features" section on page 3-38.

Enabling DVMRP Unicast Routing

Because multicast routing and unicast routing require separate topologies, PIM must follow the multicast topology to build loopless distribution trees. Using DVMRP unicast routing, Cisco routers, multilayer switches, and mrouted-based machines exchange DVMRP unicast routes, to which PIM can then reverse-path forward.

Cisco devices do not perform DVMRP multicast routing among each other, but they can exchange DVMRP routes. The DVMRP routes provide a multicast topology that might differ from the unicast topology. This allows PIM to run over the multicast topology, thereby allowing sparse-mode PIM over the MBONE topology.

When DVMRP unicast routing is enabled, the router or switch caches routes learned in DVMRP report messages in a DVMRP routing table. When PIM is running, these routes might be preferred over routes in the unicast routing table, allowing PIM to run on the MBONE topology when it is different from the unicast topology.

DVMRP unicast routing can run on all interfaces. For DVMRP tunnels, it uses DVMRP multicast routing. This feature does not enable DVMRP multicast routing among Cisco routers and multilayer switches. However, if there is a DVMRP-capable multicast router, the Cisco device can do PIM/DVMRP multicast routing.

Beginning in privileged EXEC mode, follow these steps to enable DVMRP unicast routing. This procedure is optional.

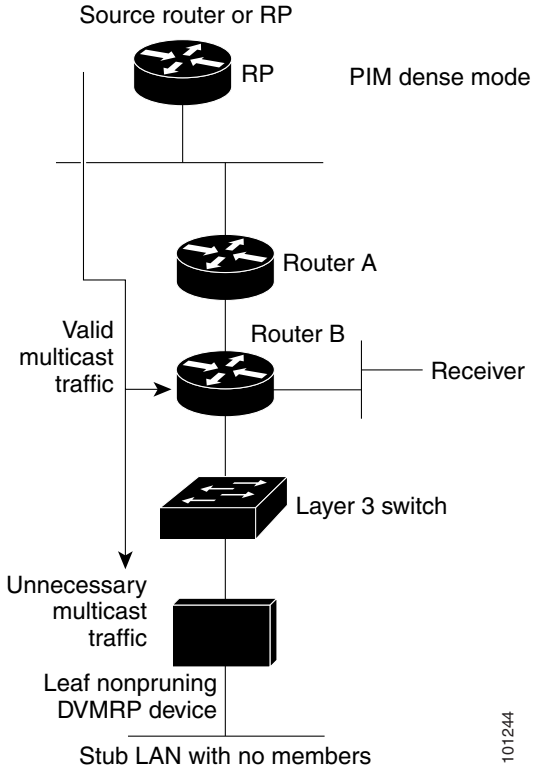
	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface that is connected to the DVMRP router, and enter interface configuration mode.
Step 3	**ip dvmrp unicast-routing**	Enable DVMRP unicast routing (to send and receive DVMRP routes). This feature is disabled by default.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable this feature, use the **no ip dvmrp unicast-routing** interface configuration command.

Rejecting a DVMRP Nonpruning Neighbor

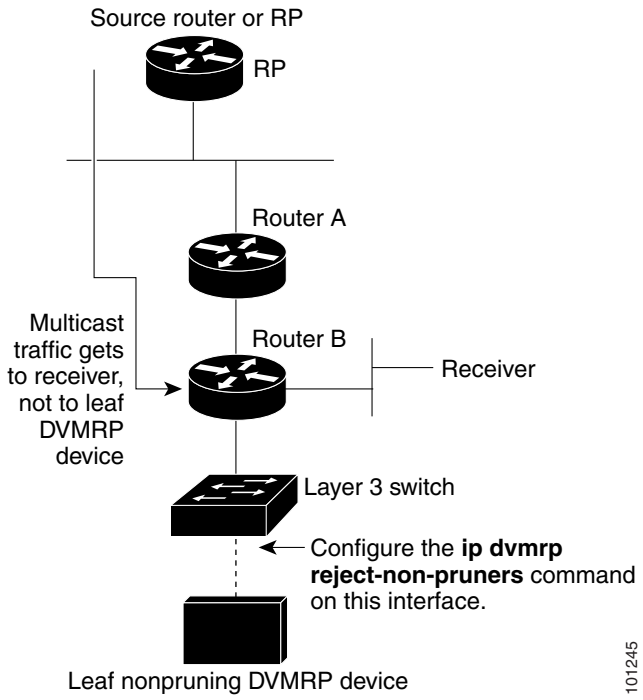
By default, Cisco devices accept all DVMRP neighbors as peers, regardless of their DVMRP capability. However, some non-Cisco devices run old versions of DVMRP that cannot prune, so they continuously receive forwarded packets, wasting bandwidth. Figure 3-8 shows this scenario.

Figure 3-8 Leaf Nonpruning DVMRP Neighbor



You can prevent the multilayer switch from peering (communicating) with a DVMRP neighbor if that neighbor does not support DVMRP pruning or grafting. To do so, configure the multilayer switch (which is a neighbor to the leaf, nonpruning DVMRP machine) with the **ip dvmrp reject-non-pruners** interface configuration command on the interface connected to the nonpruning machine as shown in Figure 3-9. In this case, when the multilayer switch receives DVMRP probe or report message without the prune-capable flag set, the switch logs a syslog message and discards the message.

Figure 3-9 Router Rejects Nonpruning DVMRP Neighbor



Note that the **ip dvmrp reject-non-pruners** interface configuration command prevents peering with neighbors only. If there are any nonpruning routers multiple hops away (downstream toward potential receivers) that are not rejected, a nonpruning DVMRP network might still exist.

Beginning in privileged EXEC mode, follow these steps to prevent peering with nonpruning DVMRP neighbors. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface connected to the nonpruning DVMRP neighbor, and enter interface configuration mode.
Step 3	**ip dvmrp reject-non-pruners**	Prevent peering with nonpruning DVMRP neighbors.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable this function, use the **no ip dvmrp reject-non-pruners** interface configuration command.

Controlling Route Exchanges

These sections describe how to tune the Cisco device advertisements of DVMRP routes:

- Limiting the Number of DVMRP Routes Advertised, page 3-47 (optional)
- Changing the DVMRP Route Threshold, page 3-47 (optional)

- Configuring a DVMRP Summary Address, page 3-48 (optional)
- Disabling DVMRP Autosummarization, page 3-50 (optional)
- Adding a Metric Offset to the DVMRP Route, page 3-50 (optional)

Limiting the Number of DVMRP Routes Advertised

By default, only 7000 DVMRP routes are advertised over an interface enabled to run DVMRP (that is, a DVMRP tunnel, an interface where a DVMRP neighbor has been discovered, or an interface configured to run the **ip dvmrp unicast-routing** interface configuration command).

Beginning in privileged EXEC mode, follow these steps to change the DVMRP route limit. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip dvmrp route-limit** *count*	Change the number of DVMRP routes advertised over an interface enabled for DVMRP. This command prevents misconfigured **ip dvmrp metric** interface configuration commands from causing massive route injection into the MBONE. By default, 7000 routes are advertised. The range is 0 to 4294967295.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To configure no route limit, use the **no ip dvmrp route-limit** global configuration command.

Changing the DVMRP Route Threshold

By default, 10,000 DVMRP routes can be received per interface within a 1-minute interval. When that rate is exceeded, a syslog message is issued, warning that there might be a route surge occurring. The warning is typically used to quickly detect when devices have been misconfigured to inject a large number of routes into the MBONE.

Beginning in privileged EXEC mode, follow these steps to change the threshold number of routes that trigger the warning. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip dvmrp routehog-notification** *route-count*	Configure the number of routes that trigger a syslog message. The default is 10,000 routes. The range is 1 to 4294967295.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default route count, use the **no ip dvmrp routehog-notification** global configuration command.

Use the **show ip igmp interface** privileged EXEC command to display a running count of routes. When the count is exceeded, *** *ALERT* *** is appended to the line.

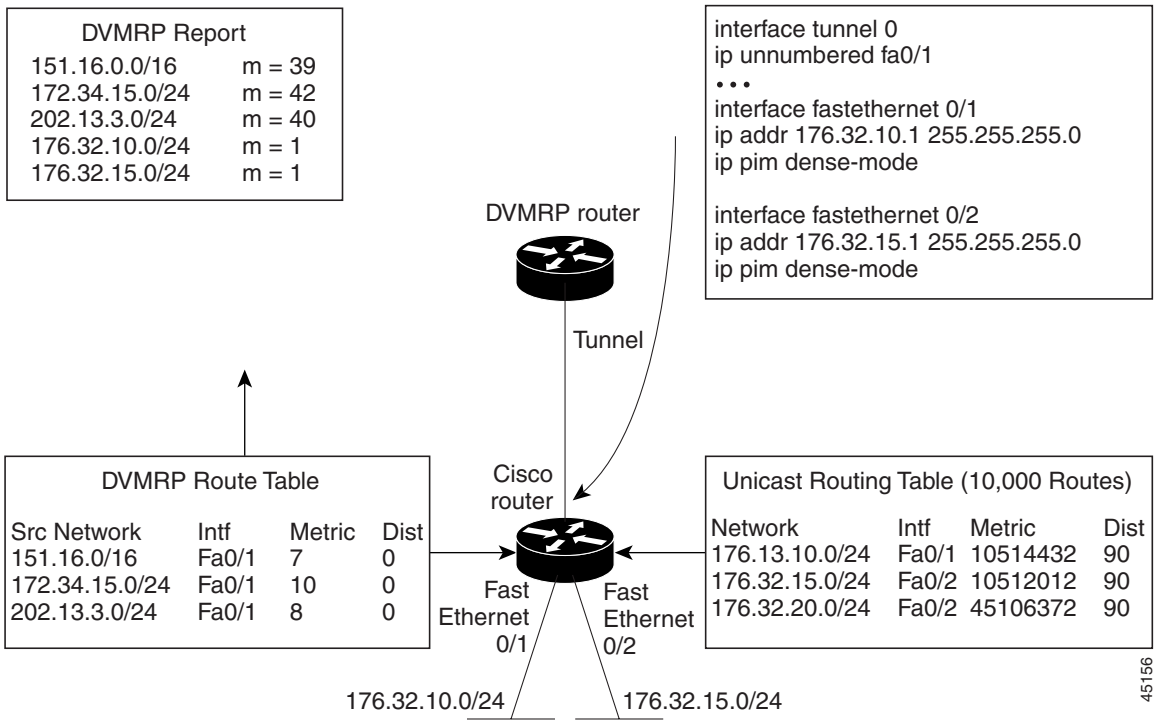
Configuring a DVMRP Summary Address

By default, a Cisco device advertises in DVMRP route-report messages only connected unicast routes (that is, only routes to subnets that are directly connected to the router) from its unicast routing table. These routes undergo normal DVMRP classful route summarization. This process depends on whether the route being advertised is in the same classful network as the interface over which it is being advertised.

Figure 3-10 shows an example of the default behavior. This example shows that the DVMRP report sent by the Cisco router contains the three original routes received from the DVMRP router that have been poison-reversed by adding 32 to the DVMRP metric. Listed after these routes are two routes that are advertisements for the two directly connected networks (176.32.10.0/24 and 176.32.15.0/24) that were taken from the unicast routing table. Because the DVMRP tunnel shares the same IP address as Fast Ethernet 0/1 and falls into the same Class B network as the two directly connected subnets, classful summarization of these routes was not performed. As a result, the DVMRP router is able to poison-reverse only these two routes to the directly connected subnets and is able to only RPF properly for multicast traffic sent by sources on these two Ethernet segments. Any other multicast source in the network behind the Cisco router that is not on these two Ethernet segments does not properly RPF-check on the DVMRP router and is discarded.

You can force the Cisco router to advertise the summary address (specified by the address and mask pair in the **ip dvmrp summary-address** *address mask* interface configuration command) in place of any route that falls in this address range. The summary address is sent in a DVMRP route report if the unicast routing table contains at least one route in this range; otherwise, the summary address is not advertised. In Figure 3-10, you configure the **ip dvmrp summary-address** command on the Cisco router tunnel interface. As a result, the Cisco router sends only a single summarized Class B advertisement for network 176.32.0.0.16 from the unicast routing table.

Figure 3-10 Only Connected Unicast Routes Are Advertised by Default



Beginning in privileged EXEC mode, follow these step to customize the summarization of DVMRP routes if the default classful autosummarization does not suit your needs. This procedure is optional.

Note At least one more-specific route must be present in the unicast routing table before a configured summary address is advertised.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface that is connected to the DVMRP router, and enter interface configuration command.
Step 3	**ip dvmrp summary-address** *address mask* [**metric** *value*]	Specify a DVMRP summary address. • For **summary-address** *address mask*, specify the summary IP address and mask that is advertised instead of the more specific route. • (Optional) For **metric** *value*, specify the metric that is advertised with the summary address. The default is 1. The range is 1 to 32.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the summary address, use the **no ip dvmrp summary-address** *address mask* [**metric** *value*] interface configuration command.

Disabling DVMRP Autosummarization

By default, the software automatically performs some level of DVMRP summarization. Disable this function if you want to advertise all routes, not just a summary. In some special cases, you can use the neighboring DVMRP router with all subnet information to better control the flow of multicast traffic in the DVMRP network. One such case might occur if the PIM network is connected to the DVMRP cloud at several points and more specific (unsummarized) routes are being injected into the DVMRP network to advertise better paths to individual subnets inside the PIM cloud.

If you configure the **ip dvmrp summary-address** interface configuration command and did not configure **no ip dvmrp auto-summary**, you get both custom and autosummaries.

Beginning in privileged EXEC mode, follow these steps to disable DVMRP autosummarization. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface connected to the DVMRP router, and enter interface configuration mode.
Step 3	**no ip dvmrp auto-summary**	Disable DVMRP autosummarization.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To re-enable auto summarization, use the **ip dvmrp auto-summary** interface configuration command.

Adding a Metric Offset to the DVMRP Route

By default, the multilayer switch increments by 1 the metric (hop count) of a DVMRP route advertised in incoming DVMRP reports. You can change the metric if you want to favor or not favor a certain route.

For example, a route is learned by multilayer switch A, and the same route is learned by multilayer switch B with a higher metric. If you want to use the path through switch B because it is a faster path, you can apply a metric offset to the route learned by switch A to make it larger than the metric learned by switch B, and you can choose the path through switch B.

Beginning in privileged EXEC mode, follow these steps to change the default metric. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.

	Command	Purpose
Step 3	**ip dvmrp metric-offset** [**in** | **out**] *increment*	Change the metric added to DVMRP routes advertised in incoming reports. The keywords have these meanings: • (Optional) **in**—Specifies that the increment value is added to incoming DVMRP reports and is reported in mrinfo replies. • (Optional) **out**—Specifies that the increment value is added to outgoing DVMRP reports for routes from the DVMRP routing table. If neither **in** nor **out** is specified, **in** is the default. For *increment*, specify the value that is added to the metric of a DVMRP router advertised in a report message. The range is 1 to 31. If the **ip dvmrp metric-offset** command is not configured on an interface, the default increment value for incoming routes is 1, and the default for outgoing routes is 0.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default value, use the **no ip dvmrp metric-offset** interface configuration command.

Monitoring and Maintaining IP Multicast Routing

These sections describe how to monitor and maintain IP multicast routing:

- Clearing Caches, Tables, and Databases, page 3-51
- Displaying System and Network Statistics, page 3-52
- Monitoring IP Multicast Routing, page 3-53

Clearing Caches, Tables, and Databases

You can remove all contents of a particular cache, table, or database. Clearing a cache, table, or database might be necessary when the contents of the particular structure are or suspected to be invalid.

You can use any of the privileged EXEC commands in Table 3-4 to clear IP multicast caches, tables, and databases:

Table 3-4 Commands for Clearing Caches, Tables, and Databases

Command	Purpose
clear ip cgmp	Clear all group entries the Catalyst switches have cached.
clear ip dvmrp route {* | *route*}	Delete routes from the DVMRP routing table.

Table 3-4 Commands for Clearing Caches, Tables, and Databases (continued)

Command	Purpose
clear ip igmp group [*group-name* | *group-address* | *interface*]	Delete entries from the IGMP cache.
clear ip mroute {* | *group* [*source*]}	Delete entries from the IP multicast routing table.
clear ip pim auto-rp *rp-address*	Clear the Auto-RP cache.
clear ip sdr [*group-address* | "*session-name*"]	Delete the Session Directory Protocol Version 2 cache or an sdr cache entry.

Displaying System and Network Statistics

You can display specific statistics such as the contents of IP routing tables, caches, and databases. You can display information to learn resource utilization and solve network problems. You can also display information about node reachability and discover the routing path your device's packets are taking through the network.

You can use any of the privileged EXEC commands in Table 3-5 to display various routing statistics:

Table 3-5 Commands for Displaying System and Network Statistics

Command	Purpose
ping [*group-name* | *group-address*]	Send an ICMP Echo Request to a multicast group address.
show ip dvmrp route [*ip-address*]	Display the entries in the DVMRP routing table.
show ip igmp groups [*group-name* | *group-address* | *type number*]	Display the multicast groups that are directly connected to the multilayer switch and that were learned through IGMP.
show ip igmp interface [*type number*]	Display multicast-related information about an interface.
show ip mcache [*group* [*source*]]	Display the contents of the IP fast-switching cache.switching, displaying
show ip mpacket [*source-address* | *name*] [*group-address* | *name*] [**detail**]	Display the contents of the circular cache-header buffer.
show ip mroute [*group-name* | *group-address*] [*source*] [**summary**] [**count**] [**active** *kbps*]	Display the contents of the IP multicast routing table.
show ip pim interface [*type number*] [**count**]	Display information about interfaces configured for PIM.
show ip pim neighbor [*type number*]	List the PIM neighbors discovered by the multilayer switch.
show ip pim rp [*group-name* | *group-address*]	Display the RP routers associated with a sparse-mode multicast group.

Table 3-5 Commands for Displaying System and Network Statistics (continued)

Command	Purpose
show ip rpf {*source-address* | *name*}	Display how the multilayer switch is doing Reverse-Path Forwarding (that is, from the unicast routing table, DVMRP routing table, or static mroutes).
show ip sdr [*group* | **"***session-name***"** | **detail**]	Display the Session Directory Protocol Version 2 cache.

Monitoring IP Multicast Routing

You can use the privileged EXEC commands in Table 3-6 to monitor IP multicast routers, packets, and paths:

Table 3-6 Commands for Monitoring IP Multicast Routing

Command	Purpose
mrinfo [*hostname* | *address*] [*source-address* | *interface*]	Query a multicast router or multilayer switch about which neighboring multicast devices are peering with it.
mstat *source* [*destination*] [*group*]	Display IP multicast packet rate and loss information.
mtrace *source* [*destination*] [*group*]	Trace the path from a source to a destination branch for a multicast distribution tree for a given group.

CHAPTER 4

Configuring MSDP

This chapter describes how to configure the Multicast Source Discovery Protocol (MSDP) on your Catalyst 3550 multilayer switch. The MSDP connects multiple Protocol-Independent Multicast sparse-mode (PIM-SM) domains.

MSDP is not fully supported in this Cisco IOS release because of a lack of support for Multicast Border Gateway Protocol (MBGP), which works closely with MSDP. However, it is possible to create default peers that MSDP can operate with if MBGP is not running.

To use this feature, you must have the IP services image (formerly known as the enhanced multilayer image [EMI]) installed on your switch.

Note For complete syntax and usage information for the commands used in this chapter, see the *Cisco IOS IP Command Reference, Volume 3 of 3: Multicast, Release 12.2*.

This chapter consists of these sections:

- Understanding MSDP, page 4-1
- Configuring MSDP, page 4-3
- Monitoring and Maintaining MSDP, page 4-17

Understanding MSDP

MSDP allows multicast sources for a group to be known to all rendezvous points (RPs) in different domains. Each PIM-SM domain uses its own RPs and does not depend on RPs in other domains. An RP runs MSDP over the TCP to discover multicast sources in other domains.

An RP in a PIM-SM domain has an MSDP peering relationship with MSDP-enabled devices in another domain. The peering relationship occurs over a TCP connection, primarily exchanging a list of sources sending to multicast groups. The TCP connections between RPs are achieved by the underlying routing system. The receiving RP uses the source lists to establish a source path.

The purpose of this topology is to have domains discover multicast sources in other domains. If the multicast sources are of interest to a domain that has receivers, multicast data is delivered over the normal, source-tree building mechanism in PIM-SM. MSDP is also used to announce sources sending to a group. These announcements must originate at the domain RP.

MSDP depends heavily on the Border Gateway Protocol (BGP) or MBGP for interdomain operation. We recommend that you run MSDP in RPs in your domain that are RPs for sources sending to global groups to be announced to the Internet.

MSDP Operation

Figure 4-1 shows MSDP operating between two MSDP peers. PIM uses MSDP as the standard mechanism to register a source with the RP of a domain. When MSDP is configured, this sequence occurs.

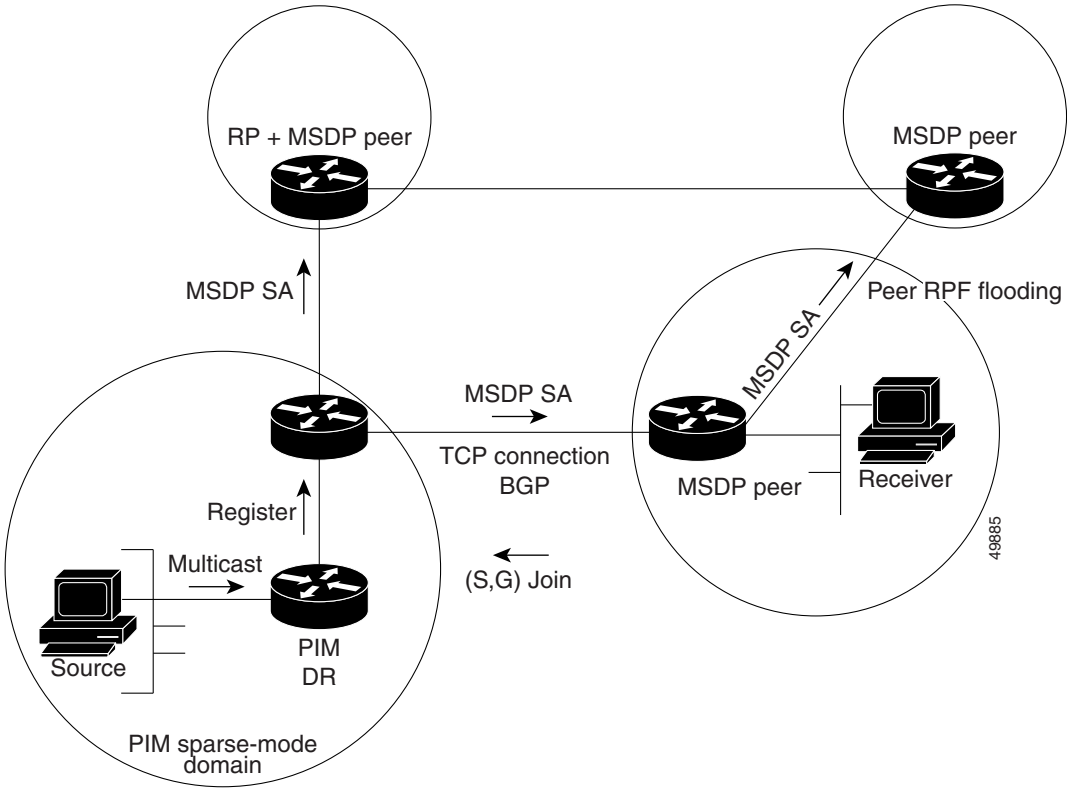
When a source sends its first multicast packet, the first-hop router (*designated router* or RP) directly connected to the source sends a PIM register message to the RP. The RP uses the register message to register the active source and to forward the multicast packet down the shared tree in the local domain. With MSDP configured, the RP also forwards a source-active (SA) message to all MSDP peers. The SA message identifies the source, the group the source is sending to, and the address of the RP or the originator ID (the IP address of the interface used as the RP address), if configured.

Each MSDP peer receives and forwards the SA message away from the originating RP to achieve peer-RPF flooding. The MSDP device examines the BGP or MBGP routing table to determine which peer is the next hop toward the originating RP of the SA message. Such a peer is called an *RPF peer* (reverse-path forwarding peer). The MSDP device forwards the message to all MSDP peers other than the RPF peer. For information on how to configure an MSDP peer when BGP and MBGP are not supported, see the "Configuring a Default MSDP Peer" section on page 4-4.

If the MSDP peer receives the same SA message from a non-RPF peer toward the originating RP, it drops the message. Otherwise, it forwards the message to all its MSDP peers.

When the RP for a domain receives the SA message from an MSDP peer, it determines if it has any join requests for the group the SA message describes. If the (*,G) entry exists with a nonempty outgoing interface list, the domain is interested in the group, and the RP triggers an (S,G) join toward the source. After the (S,G) join reaches the source's DR, a branch of the source tree has been built from the source to the RP in the remote domain. Multicast traffic can now flow from the source across the source tree to the RP and then down the shared tree in the remote domain to the receiver.

Figure 4-1 MSDP Running Between RP Peers



MSDP Benefits

MSDP has these benefits:

- It breaks up the shared multicast distribution tree. You can make the shared tree local to your domain. Your local members join the local tree, and join messages for the shared tree never need to leave your domain.
- PIM sparse-mode domains can rely only on their own RPs, decreasing reliance on RPs in another domain. This increases security because you can prevent your sources from being known outside your domain.
- Domains with only receivers can receive data without globally advertising group membership.
- Global source multicast routing table state is not required, saving memory.

Configuring MSDP

These sections describe how to configure MSDP:

- Default MSDP Configuration, page 4-4
- Configuring a Default MSDP Peer, page 4-4 (required)
- Caching Source-Active State, page 4-6 (optional)
- Requesting Source Information from an MSDP Peer, page 4-8 (optional)

- Controlling Source Information that Your Switch Originates, page 4-8 (optional)
- Controlling Source Information that Your Switch Forwards, page 4-11 (optional)
- Controlling Source Information that Your Switch Receives, page 4-13 (optional)
- Configuring an MSDP Mesh Group, page 4-15 (optional)
- Shutting Down an MSDP Peer, page 4-15 (optional)
- Including a Bordering PIM Dense-Mode Region in MSDP, page 4-16 (optional)
- Configuring an Originating Address other than the RP Address, page 4-17 (optional)

Default MSDP Configuration

MSDP is not enabled, and no default MSDP peer exists.

Configuring a Default MSDP Peer

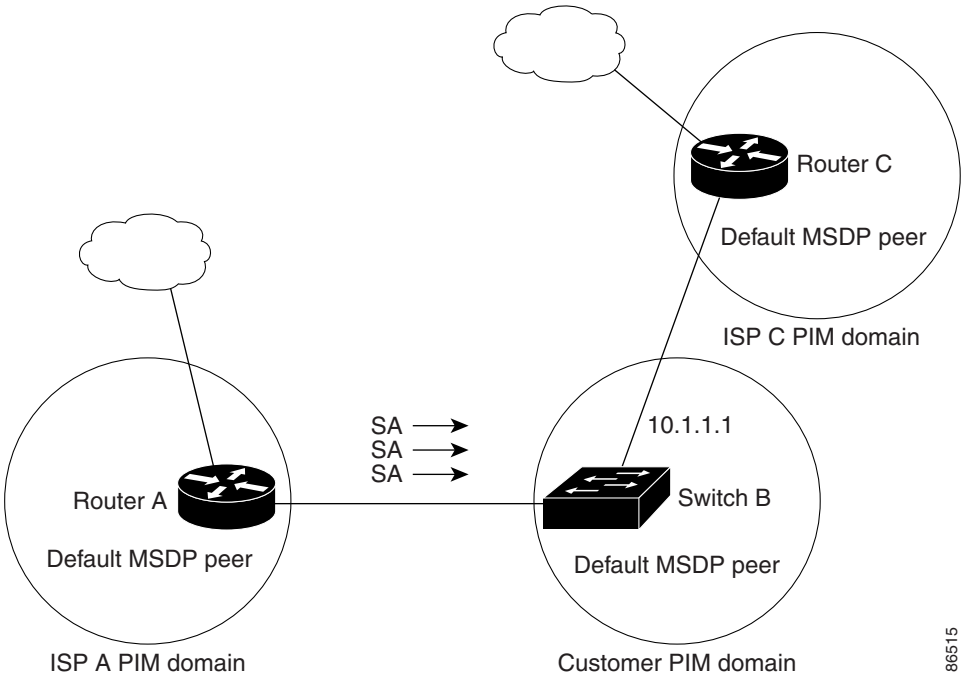
In this software release, because BGP and MBGP are not supported, you cannot configure an MSDP peer on the local multilayer switch by using the **ip msdp peer** global configuration command. Instead, you define a default MSDP peer (by using the **ip msdp default-peer** global configuration command) from which to accept all SA messages for the multilayer switch. The default MSDP peer must be a previously configured MSDP peer. Configure a default MSDP peer when the multilayer switch is not BGP- or MBGP-peering with an MSDP peer. If a single MSDP peer is configured, the multilayer switch always accepts all SA messages from that peer.

Figure 4-2 shows a network in which default MSDP peers might be used. In Figure 4-2, a customer who owns Multilayer Switch B is connected to the Internet through two Internet service providers (ISPs), one owning Router A and the other owning Router C. They are not running BGP or MBGP between them. To learn about sources in the ISP's domain or in other domains, multilayer Switch B at the customer site identifies Router A as its default MSDP peer. Multilayer Switch B advertises SA messages to both Router A and Router C but accepts SA messages only from Router A or only Router C. If Router A is first in the configuration file, it is used if it is running. If Router A is not running, only then does multilayer Switch B accept SA messages from Router C. This is the default behavior without a prefix list.

If you specify a prefix list, the peer is a default peer only for the prefixes in the list. You can have multiple active default peers when you have a prefix list associated with each. When you do not have any prefix lists, you can configure multiple default peers, but only the first one is the active default peer as long as the router has connectivity to this peer and the peer is alive. If the first configured peer fails or the connectivity to this peer fails, the second configured peer becomes the active default, and so on.

The ISP probably uses a prefix list to define which prefixes it accepts from the customer's router.

Figure 4-2 Default MSDP Peer Network



Beginning in privileged EXEC mode, follow these steps to specify a default MSDP peer. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp default-peer** *ip-address* | *name* [**prefix-list** *list*]	Define a default peer from which to accept all MSDP SA messages. • For *ip-address* | *name*, enter the IP address or Domain Name System (DNS) server name of the MSDP default peer. • (Optional) For **prefix-list** *list*, enter the list name that specifies the peer to be the default peer only for the listed prefixes. You can have multiple active default peers when you have a prefix list associated with each. When you enter multiple **ip msdp default-peer** commands with the **prefix-list** keyword, you use all the default peers at the same time for different RP prefixes. This syntax is typically used in a service provider cloud that connects stub site clouds. When you enter multiple **ip msdp default-peer** commands without the **prefix-list** keyword, a single active peer accepts all SA messages. If that peer fails, the next configured default peer accepts all SA messages. This syntax is typically used at a stub site.

	Command	Purpose
Step 3	**ip prefix-list** *name* [**description** *string*] | **seq** *number* {**permit** | **deny**} *network length*	(Optional) Create a prefix list using the name specified in Step 2. • (Optional) For **description** *string*, enter a description of up to 80 characters to describe this prefix list. • For **seq** *number,* enter the sequence number of the entry. The range is 1 to 4294967294. • The **deny** keyword denies access to matching conditions. • The **permit** keyword permits access to matching conditions. • For *network length*, specify the network number and length (in bits) of the network mask that is permitted or denied.
Step 4	**ip msdp description** {*peer-name* | *peer-address*} *text*	(Optional) Configure a description for the specified peer to make it easier to identify in a configuration or in **show** command output. By default, no description is associated with an MSDP peer.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the default peer, use the **no ip msdp default-peer** global configuration command.

This example shows a partial configuration of Router A and Router C in Figure 4-2. Each of these ISPs have more than one customer (like the customer in Figure 4-2) who use default peering (no BGP or MBGP). In that case, they might have similar configurations. That is, they accept SAs only from a default peer if the SA is permitted by the corresponding prefix list.

Router A

```
Router(config)# ip msdp default-peer 10.1.1.1
Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a
Router(config)# ip prefix-list site-b permit 10.0.0.0/8
```

Router C

```
Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a
Router(config)# ip prefix-list site-b permit 10.0.0.0/8
```

Caching Source-Active State

By default, the multilayer switch does not cache source/group pairs from received SA messages. When the switch forwards the MSDP SA information, it does not store it in memory. Therefore, if a member joins a group soon after a SA message is received by the local RP, that member needs to wait until the next SA message to hear about the source. This delay is known as join latency.

If you want to sacrifice some memory in exchange for reducing the latency of the source information, you can configure the switch to cache SA messages.

Beginning in privileged EXEC mode, follow these steps to enable the caching of source/group pairs. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp cache-sa-state** [**list** *access-list-number*]	Enable the caching of source/group pairs (create an SA state). Those pairs that pass the access list are cached. For **list** *access-list-number*, the range is 100 to 199.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*	Create an IP extended access list, repeating the command as many times as necessary. • For *access-list-number*, the range is 100 to 199. Enter the same number created in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *protocol*, enter **ip** as the protocol name. • For *source*, enter the number of the network or host from which the packet is being sent. • For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. • For *destination*, enter the number of the network or host to which the packet is being sent. • For *destination-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note An alternative to this command is the **ip msdp sa-request** global configuration command, which causes the multilayer switch to send an SA request message to the MSDP peer when a new member for a group becomes active. For more information, see the next section.

To return to the default setting (no SA state is created), use the **no ip msdp cache-sa-state** global configuration command.

This example shows how to enable the cache state for all sources in 171.69.0.0/16 sending to groups 224.2.0.0/16:

```
Switch(config)# ip msdp cache-sa-state 100
Switch(config)# access-list 100 permit ip 171.69.0.0 0.0.255.255 224.2.0.0 0.0.255.255
```

Requesting Source Information from an MSDP Peer

Local RPs can send SA requests and get immediate responses for all active sources for a given group. By default, the multilayer switch does not send any SA request messages to its MSDP peers when a new member joins a group and wants to receive multicast traffic. The new member waits to receive the next periodic SA message.

If you want a new member of a group to learn the active multicast sources in a connected PIM sparse-mode domain that are sending to a group, configure the switch to send SA request messages to the specified MSDP peer when a new member joins a group. The peer replies with the information in its SA cache. If the peer does not have a cache configured, this command has no result. Configuring this feature reduces join latency but sacrifices memory.

Beginning in privileged EXEC mode, follow these steps to configure the switch to send SA request messages to the MSDP peer when a new member joins a group and wants to receive multicast traffic. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp sa-request** {*ip-address* | *name*}	Configure the switch to send SA request messages to the specified MSDP peer. For *ip-address* | *name*, enter the IP address or name of the MSDP peer from which the local switch requests SA messages when a new member for a group becomes active. Repeat the command for each MSDP peer that you want to supply with SA messages.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip msdp sa-request** {*ip-address* | *name*} global configuration command.

This example shows how to configure the switch to send SA request messages to the MSDP peer at 171.69.1.1:

```
Switch(config)# ip msdp sa-request 171.69.1.1
```

Controlling Source Information that Your Switch Originates

You can control the multicast source information that originates with your switch:

- Sources you advertise (based on your sources)
- Receivers of source information (based on knowing the requestor)

For more information, see the "Redistributing Sources" section on page 4-9 and the "Filtering Source-Active Request Messages" section on page 4-10.

Redistributing Sources

SA messages are originated on RPs to which sources have registered. By default, any source that registers with an RP is advertised. The *A flag* is set in the RP when a source is registered, which means the source is advertised in an SA unless it is filtered.

Beginning in privileged EXEC mode, follow these steps to further restrict which registered sources are advertised. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp redistribute** [**list** *access-list-name*] [**asn** *aspath-access-list-number*] [**route-map** *map*]	Configure which (S,G) entries from the multicast routing table are advertised in SA messages. By default, only sources within the local domain are advertised. • (Optional) For **list** *access-list-name*, enter the name or number of an IP standard or extended access list. The range is 1 to 99 for standard access lists and 100 to 199 for extended lists. The access list controls which local sources are advertised and to which groups they send. • (Optional) For **asn** *aspath-access-list-number*, enter the IP standard or extended access list number in the range 1 to 199. This access list number must also be configured in the **ip as-path access-list** command. • (Optional) For **route-map** *map*, enter the IP standard or extended access list number in the range 1 to 199. This access list number must also be configured in the **ip as-path access-list** command. The access list or autonomous system path access list determines which (S,G) pairs are advertised.

	Command	Purpose
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] **or** **access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*	Create an IP standard access list, repeating the command as many times as necessary. or Create an IP extended access list, repeating the command as many times as necessary. • For *access-list-number*, the range is 1 to 99 for standard access lists and 100 to 199 for extended lists. Enter the same number created in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *protocol*, enter **ip** as the protocol name. • For *source*, enter the number of the network or host from which the packet is being sent. • For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. • For *destination*, enter the number of the network or host to which the packet is being sent. • For *destination-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the filter, use the **no ip msdp redistribute** global configuration command.

Filtering Source-Active Request Messages

By default, only multilayer switches that are caching SA information can respond to SA requests. By default, such a switch honors all SA request messages from its MSDP peers and supplies the IP addresses of the active sources.

However, you can configure the switch to ignore all SA requests from an MSDP peer. You can also honor only those SA request messages from a peer for groups described by a standard access list. If the groups in the access list pass, SA request messages are accepted. All other such messages from the peer for other groups are ignored.

Beginning in privileged EXEC mode, follow these steps to configure one of these options. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp filter-sa-request** *ip-address* | *name* or **ip msdp filter-sa-request** {*ip-address* | *name*} **list** *access-list-number*	Filter all SA request messages from the specified MSDP peer. or Filter SA request messages from the specified MSDP peer for groups that pass the standard access list. The access list describes a multicast group address. The range for the access-list-number is 1 to 99.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create an IP standard access list, repeating the command as many times as necessary. • For *access-list-number*, the range is 1 to 99. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the number of the network or host from which the packet is being sent. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip msdp filter-sa-request** {*ip-address* | *name*} global configuration command.

This example shows how to configure the switch to filter SA request messages from the MSDP peer at 171.69.2.2. SA request messages from sources on network 192.4.22.0 pass access list 1 and are accepted; all others are ignored.

```
Switch(config)# ip msdp filter sa-request 171.69.2.2 list 1
Switch(config)# access-list 1 permit 192.4.22.0 0.0.0.255
```

Controlling Source Information that Your Switch Forwards

By default, the multilayer switch forwards all SA messages it receives to all its MSDP peers. However, you can prevent outgoing messages from being forwarded to a peer by using a filter or by setting a time-to-live (TTL) value. These methods are described in the next sections.

Using a Filter

By creating a filter, you can perform one of these actions:

- Filter all source/group pairs
- Specify an IP extended access list to pass only certain source/group pairs
- Filter based on match criteria in a route map

Beginning in privileged EXEC mode, follow these steps to apply a filter:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp sa-filter out** *ip-address* | *name* or **ip msdp sa-filter out** {*ip-address* | *name*} **list** *access-list-number* or **ip msdp sa-filter out** {*ip-address* | *name*} **route-map** *map-tag*	Filter all SA messages to the specified MSDP peer. or To the specified peer, pass only those SA messages that pass the IP extended access list. The range for the extended *access-list-number* is 100 to 199. If both the **list** and the **route-map** keywords are used, all conditions must be true to pass any (S,G) pair in outgoing SA messages. or To the specified MSDP peer, pass only those SA messages that meet the match criteria in the route map *map-tag*. If all match criteria are true, a **permit** from the route map passes routes through the filter. A **deny** filters routes.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*	(Optional) Create an IP extended access list, repeating the command as many times as necessary. • For *access-list-number*, enter the number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *protocol*, enter **ip** as the protocol name. • For *source*, enter the number of the network or host from which the packet is being sent. • For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. • For *destination*, enter the number of the network or host to which the packet is being sent. • For *destination-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the filter, use the **no ip msdp sa-filter out** {*ip-address* | *name*} [**list** *access-list-number*] [**route-map** *map-tag*] global configuration command.

This example shows how to allow only (S,G) pairs that pass access list 100 to be forwarded in an SA message to the peer named *switch.cisco.com*:

```
Switch(config)# ip msdp peer switch.cisco.com connect-source gigabitethernet0/1
Switch(config)# ip msdp sa-filter out switch.cisco.com list 100
Switch(config)# access-list 100 permit ip 171.69.0.0 0.0.255.255 224.20 0 0.0.255.255
```

Using TTL to Limit the Multicast Data Sent in SA Messages

You can use a TTL value to control what data is encapsulated in the first SA message for every source. Only multicast packets with an IP-header TTL greater than or equal to the *ttl* argument are sent to the specified MSDP peer. For example, you can limit internal traffic to a TTL of 8. If you want other groups to go to external locations, you must send those packets with a TTL greater than 8.

Beginning in privileged EXEC mode, follow these steps to establish a TTL threshold. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp ttl-threshold** {*ip-address* | *name*} *ttl*	Limit which multicast data is encapsulated in the first SA message to the specified MSDP peer. • For *ip-address* | *name*, enter the IP address or name of the MSDP peer to which the TTL limitation applies. • For *ttl*, enter the TTL value. The default is 0, which means all multicast data packets are forwarded to the peer until the TTL is exhausted. The range is 0 to 255.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip msdp ttl-threshold** {*ip-address* | *name*} global configuration command.

Controlling Source Information that Your Switch Receives

By default, the multilayer switch receives all SA messages that its MSDP Reverse-Path Forwarding peers send to it. However, you can control the source information that you receive from MSDP peers by filtering incoming SA messages. In other words, you can configure the switch to not accept them.

You can perform one of these actions:

- Filter all incoming SA messages from an MSDP peer
- Specify an IP extended access list to pass certain source/group pairs
- Filter based on match criteria in a route map

Beginning in privileged EXEC mode, follow these steps to apply a filter. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp sa-filter in** *ip-address* | *name*	Filter all SA messages from the specified MSDP peer.
	or	or
	ip msdp sa-filter in {*ip-address* | *name*} **list** *access-list-number*	From the specified peer, pass only those SA messages that pass the IP extended access list. The range for the extended *access-list-number* is 100 to 199. If both the **list** and the **route-map** keywords are used, all conditions must be true to pass any (S,G) pair in incoming SA messages.
	or	or
	ip msdp sa-filter in {*ip-address* | *name*} **route-map** *map-tag*	From the specified MSDP peer, pass only those SA messages that meet the match criteria in the route map *map-tag*. If all match criteria are true, a **permit** from the route map passes routes through the filter. A **deny** will filter routes.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*	(Optional) Create an IP extended access list, repeating the command as many times as necessary. • For *access-list-number*, enter the number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *protocol*, enter **ip** as the protocol name. • For *source*, enter the number of the network or host from which the packet is being sent. • For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. • For *destination*, enter the number of the network or host to which the packet is being sent. • For *destination-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the filter, use the **no ip msdp sa-filter in** {*ip-address* | *name*} [**list** *access-list-number*] [**route-map** *map-tag*] global configuration command.

This example shows how to filter all SA messages from the peer named *switch.cisco.com*:

```
Switch(config)# ip msdp peer switch.cisco.com connect-source gigabitethernet0/1
Switch(config)# ip msdp sa-filter in switch.cisco.com
```

Configuring an MSDP Mesh Group

An MSDP mesh group is a group of MSDP speakers that have fully meshed MSDP connectivity among one another. Any SA messages received from a peer in a mesh group are not forwarded to other peers in the same mesh group. Thus, you reduce SA message flooding and simplify peer-RPF flooding. Use the **ip msdp mesh-group** global configuration command when there are multiple RPs within a domain. It is especially used to send SA messages across a domain. You can configure multiple mesh groups (with different names) in a single multilayer switch.

Beginning in privileged EXEC mode, follow these steps to create a mesh group. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp mesh-group** *name* {*ip-address* | *name*}	Configure an MSDP mesh group, and specify the MSDP peer belonging to that mesh group. By default, the MSDP peers do not belong to a mesh group. • For *name*, enter the name of the mesh group. • For *ip-address* | *name*, enter the IP address or name of the MSDP peer to be a member of the mesh group.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.
Step 6		Repeat this procedure on each MSDP peer in the group.

To remove an MSDP peer from a mesh group, use the **no ip msdp mesh-group** *name* {*ip-address* | *name*} global configuration command.

Shutting Down an MSDP Peer

If you want to configure many MSDP commands for the same peer and you do not want the peer to become active, you can shut down the peer, configure it, and later bring it up. When a peer is shut down, the TCP connection is terminated and is not restarted. You can also shut down an MSDP session without losing configuration information for the peer.

Beginning in privileged EXEC mode, follow these steps to shut down a peer. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp shutdown** {*peer-name* | *peer address*}	Administratively shut down the specified MSDP peer without losing configuration information. For *peer-name* | *peer address,* enter the IP address or name of the MSDP peer to shut down.
Step 3	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To bring the peer back up, use the **no ip msdp shutdown** {*peer-name* | *peer address*} global configuration command. The TCP connection is reestablished

Including a Bordering PIM Dense-Mode Region in MSDP

You can configure MSDP on a multilayer switch that borders a PIM sparse-mode region with a dense-mode region. By default, active sources in the dense-mode region do not participate in MSDP.

Note We do not recommend using the **ip msdp border sa-address** global configuration command. It is better to configure the border router in the sparse-mode domain to proxy-register sources in the dense-mode domain to the RP of the sparse-mode domain and have the sparse-mode domain use standard MSDP procedures to advertise these sources.

Beginning in privileged EXEC mode, follow these steps to configure the border router to send SA messages for sources active in the dense-mode region to the MSDP peers. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp border sa-address** *type number*	Configure the switch on the border between a dense-mode and sparse-mode region to send SA messages about active sources in the dense-mode region. For *type number*, specify the interface type and number from which the IP address is derived and used as the RP address in SA messages. The IP address of the interface is used as the Originator-ID, which is the RP field in the SA message.
Step 3	**ip msdp redistribute** [**list** *access-list-name*] [**asn** *aspath-access-list-number*] [**route-map** *map*]	Configure which (S,G) entries from the multicast routing table are advertised in SA messages. For more information, see the "Redistributing Sources" section on page 4-9.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note that the **ip msdp originator-id** global configuration command also identifies an interface type and number to be used as the RP address. If both the **ip msdp border sa-address** and the **ip msdp originator-id** global configuration commands are configured, the address derived from the **ip msdp originator-id** command specifies the RP address.

To return to the default setting (active sources in the dense-mode region do not participate in MSDP), use the **no ip msdp border sa-address** *type number* global configuration command.

Configuring an Originating Address other than the RP Address

You can allow an MSDP speaker that originates an SA message to use the IP address of the interface as the RP address in the SA message by changing the Originator ID. You might change the Originator ID in one of these cases:

- If you configure a logical RP on multiple multilayer switches in an MSDP mesh group.
- If you have a multilayer switch that borders a PIM sparse-mode domain and a dense-mode domain. If a switch borders a dense-mode domain for a site, and sparse-mode is being used externally, you might want dense-mode sources to be known to the outside world. Because this switch is not an RP, it would not have an RP address to use in an SA message. Therefore, this command provides the RP address by specifying the address of the interface.

Beginning in privileged EXEC mode, follow these steps to allow an MSDP speaker that originates an SA message to use the IP address on the interface as the RP address in the SA message. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip msdp originator-id** *type number*	Configures the RP address in SA messages to be the address of the originating device interface. For *type number*, specify the interface type and number on the local switch.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

If both the **ip msdp border sa-address** and the **ip msdp originator-id** global configuration commands are configured, the address derived from the **ip msdp originator-id** command specifies the address of the RP.

To prevent the RP address from being derived in this way, use the **no ip msdp originator-id** *type number* global configuration command.

Monitoring and Maintaining MSDP

To monitor MSDP SA messages, peers, state, or peer status, use one or more of the privileged EXEC commands in Table 4-1:

Table 4-1 Commands for Monitoring and Maintaining MSDP

Command	Purpose
debug ip msdp [*peer-address* | *name*] [**detail**] [**routes**]	Debugs an MSDP activity.
debug ip msdp resets	Debugs MSDP peer reset reasons.
show ip msdp count [*autonomous-system-number*]	Displays the number of sources and groups originated in SA messages from each autonomous system. The **ip msdp cache-sa-state** command must be configured for this command to produce any output.

Table 4-1 Commands for Monitoring and Maintaining MSDP (continued)

Command	Purpose
show ip msdp peer [*peer-address* | *name*]	Displays detailed information about an MSDP peer.
show ip msdp sa-cache [*group-address* | *source-address* | *group-name* | *source-name*] [*autonomous-system-number*]	Displays (S,G) state learned from MSDP peers.
show ip msdp summary	Displays MSDP peer status and SA message counts.

To clear MSDP connections, statistics, or SA cache entries, use the privileged EXEC commands in Table 4-2:

Table 4-2 Commands for Clearing MSDP Connections, Statistics, or SA Cache Entries

Command	Purpose
clear ip msdp peer *peer-address* | *name*	Clears the TCP connection to the specified MSDP peer, resetting all MSDP message counters.
clear ip msdp statistics [*peer-address* | *name*]	Clears statistics counters for one or all the MSDP peers without resetting the sessions.
clear ip msdp sa-cache [*group-address* | *name*]	Clears the SA cache entries for all entries, all sources for a specific group, or all entries for a specific source/group pair.

CHAPTER 5

Configuring Fallback Bridging

This chapter describes how to configure fallback bridging (VLAN bridging) on your Catalyst 3550 switch. With fallback bridging, you can forward non-IP packets that the switch does not route between VLAN bridge domains and routed ports.

To use this feature, you must have the IP services image, formerly known as the enhanced multilayer (EMI) image, installed on your switch.

Note For complete syntax and usage information for the commands used in this chapter, see the *Cisco IOS Bridging and IBM Networking Command Reference, Volume 1 of 2, Release 12.2*.

This chapter consists of these sections:

- Understanding Fallback Bridging, page 5-1
- Configuring Fallback Bridging, page 5-2
- Monitoring and Maintaining Fallback Bridging, page 5-12

Understanding Fallback Bridging

With fallback bridging, the switch bridges together two or more VLANs or routed ports, essentially connecting multiple VLANs within one bridge domain. Fallback bridging forwards traffic that the switch does not route and forwards traffic belonging to a nonroutable protocol such as DECnet.

Fallback bridging does not allow the spanning trees from the VLANs being bridged to collapse; each VLAN has its own spanning-tree instance and a separate spanning tree, called the VLAN-bridge spanning tree, which runs on top of the bridge group to prevent loops.

A VLAN bridge domain is represented with switch virtual interface (SVI). A set of SVIs and routed ports (which do not have any VLANs associated with them) can be configured (grouped together) to form a bridge group. Recall that an SVI represents a VLAN of switch ports as one interface to the routing or bridging function in the system. You associate only one SVI with a VLAN, and you configure an SVI for a VLAN only when you want to route between VLANs, to fallback-bridge nonroutable protocols between VLANs, or to provide IP host connectivity to the switch. A routed port is a physical port that acts like a port on a router, but it is not connected to a router. A routed port is not associated with a particular VLAN, does not support VLAN subinterfaces, but behaves like a normal routed interface. For more information about SVIs and routed ports, see Chapter 9, "Configuring Interface Characteristics."

A bridge group is an internal organization of network interfaces on a switch. Bridge groups cannot be used to identify traffic switched within the bridge group outside the switch on which they are defined. Bridge groups on the same switch function as distinct bridges; that is, bridged traffic and bridge protocol

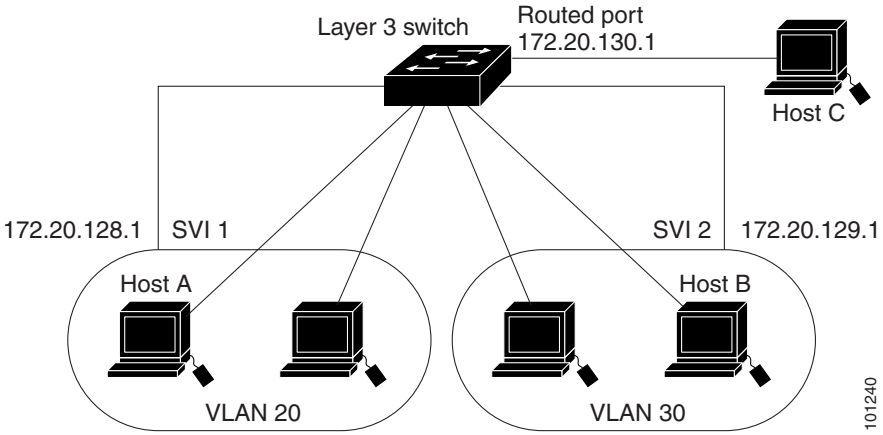
data units (BPDUs) are not exchanged between different bridge groups on a switch. An interface can be a member of only one bridge group. Use a bridge group for each separately bridged (topologically distinct) network connected to the switch.

These are the reasons for placing network interfaces into a bridge group:

- To bridge all nonrouted traffic among the network interfaces making up the bridge group. If the packet destination address is in the bridge table, the packet is forwarded on a single interface in the bridge group. If the packet destination address is not in the bridge table, the packet is flooded on all forwarding interfaces in the bridge group. The switch places source addresses in the bridge table as it learns them during the bridging process.
- To participate in the spanning-tree algorithm by receiving, and in some cases sending, BPDUs on the LANs to which they are attached. A separate spanning-tree process runs for each configured bridge group. Each bridge group participates in a separate spanning-tree instance. A bridge group establishes a spanning-tree instance based on the BPDUs it receives on only its member interfaces.

Figure 5-1 shows a fallback bridging network example. The switch has two interfaces configured as SVIs with different assigned IP addresses and attached to two different VLANs. Another interface is configured as a routed port with its own IP address. If all three of these ports are assigned to the same bridge group, non-IP protocol frames can be forwarded among the end stations connected to the switch even though they are on different networks and in different VLANs. IP addresses do not need to be assigned to routed ports or SVIs for fallback bridging to work.

Figure 5-1 Fallback Bridging Network Example



Configuring Fallback Bridging

These sections describe how to configure fallback bridging on your switch:

- Default Fallback Bridging Configuration, page 5-3
- Fallback Bridging Configuration Guidelines, page 5-3
- Creating a Bridge Group, page 5-3 (required)
- Preventing the Forwarding of Dynamically Learned Stations, page 5-5 (optional)
- Configuring the Bridge Table Aging Time, page 5-5 (optional)

- Filtering Frames by a Specific MAC Address, page 5-6 (optional)
- Adjusting Spanning-Tree Parameters, page 5-7 (optional)

Default Fallback Bridging Configuration

Table 5-1 shows the default fallback bridging configuration.

Table 5-1 Default Fallback Bridging Configuration

Feature	Default Setting
Bridge groups	None are defined or assigned to an interface. No VLAN-bridge STP is defined.
Switch forwards frames for stations that it has dynamically learned	Enabled.
Bridge table aging time for dynamic entries	300 seconds.
MAC-layer frame filtering	Disabled.
Spanning tree parameters:	
• Switch priority	• 32768.
• Interface priority	• 128.
• Interface path cost	• 10 Mbps: 100. 100 Mbps: 19. 1000 Mbps: 4.
• Hello BPDU interval	• 2 seconds.
• Forward-delay interval	• 20 seconds.
• Maximum idle interval	• 30 seconds.

Fallback Bridging Configuration Guidelines

A maximum of 31 bridge groups can be configured on the switch.

An interface (an SVI or routed port) can be a member of only one bridge group.

Use a bridge group for each separately bridged (topologically distinct) network connected to the switch.

All protocols except IP (Version 4 and Version 6), Address Resolution Protocol (ARP), reverse ARP (RARP), LOOPBACK, and Frame Relay ARP are fallback bridged.

Creating a Bridge Group

To configure fallback bridging for a set of SVIs or routed ports, these interfaces must be assigned to bridge groups. All interfaces in the same group belong to the same bridge domain. Each SVI or routed port can be assigned to only one bridge group. A maximum of 31 bridge groups can be configured on the switch.

Note The protected port feature is not compatible with fallback bridging. When fallback bridging is enabled, it is possible for packets to be forwarded from one protected port on a switch to another protected port on the same switch if the ports are in different VLANs.

Beginning in privileged EXEC mode, follow these steps to create a bridge group and assign an interface to it. This procedure is required.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**bridge** *bridge-group* **protocol vlan-bridge**	Assign a bridge group number, and specify the VLAN-bridge spanning-tree protocol to run in the bridge group. The **ibm** and **dec** keywords are not supported. For *bridge-group*, specify the bridge group number. The range is 1 to 255. You can create up to 31 bridge groups. Frames are bridged only among interfaces in the same group.
Step 3	**interface** *interface-id*	Specify the interface on which you want to assign the bridge group, and enter interface configuration mode. The specified interface must be one of these: • A routed port: a physical port that you have configured as a Layer 3 port by entering the **no switchport** interface configuration command. • An SVI: a VLAN interface that you created by using the **interface vlan** *vlan-id* global configuration command. **Note** You can assign an IP address to the routed port or to the SVI, but it is not required.
Step 4	**bridge-group** *bridge-group*	Assign the interface to the bridge group created in Step 2. By default, the interface is not assigned to any bridge group. An interface can be assigned to only one bridge group.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a bridge group, use the **no bridge** *bridge-group* global configuration command. To remove an interface from a bridge group and to remove the bridge group, use the **no bridge-group** *bridge-group* interface configuration command.

This example shows how to create bridge group 10, to specify that the VLAN-bridge STP runs in the bridge group, to define an interface as a routed port, and to assign the interface to the bridge group:

```
Switch(config)# bridge 10 protocol vlan-bridge
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# no switchport
Switch(config-if)# no shutdown
Switch(config-if)# bridge-group 10
```

This example shows how to create bridge group 10 and to specify that the VLAN-bridge STP runs in the bridge group. It defines an SVI for VLAN 2 and assigns it to the bridge group:

```
Switch(config)# bridge 10 protocol vlan-bridge
Switch(config)# vlan 2
Switch(config-vlan)# exit
Switch(config)# interface vlan2
Switch(config-if)# bridge-group 10
Switch(config-if)# exit
```

Preventing the Forwarding of Dynamically Learned Stations

By default, the switch forwards any frames for stations that it has dynamically learned. By disabling this activity, the switch only forwards frames whose addresses have been statically configured into the forwarding cache.

Beginning in privileged EXEC mode, follow these steps to prevent the switch from forwarding frames for stations that it has dynamically learned. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**no bridge** *bridge-group* **acquire**	Enable the switch to stop forwarding any frames for stations that it has dynamically learned through the discovery process and to limit frame forwarding to statically configured stations. The switch filters all frames except those whose destined-to addresses have been statically configured into the forwarding cache. To configure a static address, use the **bridge** *bridge-group* **address** *mac-address* {**forward** | **discard**} global configuration command. For *bridge-group*, specify the bridge group number. The range is 1 to 255.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entry.
Step 5	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

To cause the switch to forward frames to stations that it has dynamically learned, use the **bridge** *bridge-group* **acquire** global configuration command.

This example shows how to prevent the switch from forwarding frames for stations that it has dynamically learned in bridge group 10:

```
Switch(config)# no bridge 10 acquire
```

Configuring the Bridge Table Aging Time

A switch forwards, floods, or drops packets based on the bridge table. The bridge table maintains both static and dynamic entries. Static entries are entered by you or learned by the switch. Dynamic entries are entered by the bridge learning process. A dynamic entry is automatically removed after a specified length of time, known as aging time, from the time the entry was created or last updated.

If you are likely to move hosts on a switched network, decrease the aging-time to enable the switch to quickly adapt to the change. If hosts on a switched network do not continuously send packets, increase the aging time to keep the dynamic entries for a longer time and thus reduce the possibility of flooding when the hosts send again.

Beginning in privileged EXEC mode, follow these steps to configure the aging time. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**bridge** *bridge-group* **aging-time** *seconds*	Specify the length of time that a dynamic entry remains in the bridge table from the time the entry was created or last updated. • For *bridge-group*, specify the bridge group number. The range is 1 to 255. • For *seconds*, enter a number from 0 to 1000000. The default is 300 seconds.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entry.
Step 5	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

To return to the default aging-time interval, use the **no bridge** *bridge-group* **aging-time** global configuration command.

This example shows how to change the bridge table aging time to 200 seconds for bridge group 10:

```
Switch(config)# bridge 10 aging-time 200
```

Filtering Frames by a Specific MAC Address

A switch examines frames and sends them through the internetwork according to the destination address; a switch does not forward a frame back to its originating network segment. You can use the software to configure specific administrative filters that filter frames based on information other than the paths to their destinations.

You can filter frames with a particular MAC-layer station destination address. You can configure any number of addresses in the system without a performance penalty.

Beginning in privileged EXEC mode, follow these steps to filter by the MAC-layer address. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**bridge** *bridge-group* **address** *mac-address* {**forward** | **discard**} [*interface-id*]	Specify the MAC address to discard or forward. • For *bridge-group*, specify the bridge group number. The range is 1 to 255. • For **address** *mac-address*, specify the MAC-layer destination address to be filtered. • Specify **forward** if you want the frame destined to the specified interface to be forwarded. Specify **discard** if you want the frame to be discarded. • (Optional) For *interface-id*, specify the interface on which the address can be reached.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entry.
Step 5	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

To disable the frame forwarding ability, use the **no bridge** *bridge-group* **address** *mac-address* global configuration command.

This example shows how to forward a frame with MAC address 0800.cb00.45e9 through an interface in bridge group 1:

```
Switch(config)# bridge 1 address 0800.cb00.45e9 forward gigabitethernet0/1
```

Adjusting Spanning-Tree Parameters

You might need to adjust certain spanning-tree parameters if the default values are not suitable. You configure parameters affecting the entire spanning tree by using variations of the **bridge** global configuration command. You configure interface-specific parameters by using variations of the **bridge-group** interface configuration command.

You can adjust spanning-tree parameters by performing any of the tasks in these sections:

- Changing the Switch Priority, page 5-8 (optional)
- Changing the Interface Priority, page 5-8 (optional)
- Assigning a Path Cost, page 5-9 (optional)
- Adjusting BPDU Intervals, page 5-10 (optional)
- Disabling the Spanning Tree on an Interface, page 5-12 (optional)

Note Only network administrators with a good understanding of how switches and STP function should make adjustments to spanning-tree parameters. Poorly planned adjustments can have a negative impact on performance. A good source on switching is the IEEE 802.1D specification; for more information, see the “References and Recommended Reading” appendix in the *Cisco IOS Configuration Fundamentals Command Reference*.

Changing the Switch Priority

You can globally configure the priority of an individual switch when two switches tie for position as the root switch, or you can configure the likelihood that a switch will be selected as the root switch. This priority is determined by default; however, you can change it.

Beginning in privileged EXEC mode, follow these steps to change the switch priority. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**bridge** *bridge-group* **priority** *number*	Change the priority of the switch. • For *bridge-group*, specify the bridge group number. The range is 1 to 255. • For *number*, enter a number from 0 to 65535. The default is 32768. The lower the number, the more likely the switch will be chosen as the root.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entry.
Step 5	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

To return to the default setting, use the **no bridge** *bridge-group* **priority** global configuration command. To change the priority on an interface, use the **bridge-group priority** interface configuration command (described in the next section).

This example shows how to set the switch priority to 100 for bridge group 10:

```
Switch(config)# bridge 10 priority 100
```

Changing the Interface Priority

You can change the priority for an interface. When two switches tie for position as the root switch, you configure an interface priority to break the tie. The switch with the lowest interface value is elected.

Beginning in privileged EXEC mode, follow these steps to change the interface priority. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to set the priority, and enter interface configuration mode.
Step 3	**bridge-group** *bridge-group* **priority** *number*	Change the priority of an interface. • For *bridge-group*, specify the bridge group number. The range is 1 to 255. • For *number*, enter a number from 0 to 255. The lower the number, the more likely that the interface on the switch will be chosen as the root. The default is 128.
Step 4	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 5	**show running-config**	Verify your entry.
Step 6	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

No **no** form of this command exists. To return to the default setting, use the **no bridge-group** *bridge-group* **priority** interface configuration command.

This example shows how to change the priority of an interface to 20 in bridge group 10:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# bridge-group 10 priority 20
```

Assigning a Path Cost

Each interface has a path cost associated with it. By convention, the path cost is 1000/data rate of the attached LAN, in Mbps.

Beginning in privileged EXEC mode, follow these steps to assign a path cost. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to set the path cost, and enter interface configuration mode.
Step 3	**bridge-group** *bridge-group* **path-cost** *cost*	Assign the path cost of an interface. • For *bridge-group*, specify the bridge group number. The range is 1 to 255. • For *cost*, enter a number from 1 to 65536. The higher the value, the higher the cost. – For 10 Mbps, the default path cost is 100. – For 100 Mbps, the default path cost is 19. – For 1000 Mbps, the default path cost is 4.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entry.
Step 6	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

To return to the default path cost, use the **no bridge-group** *bridge-group* **path-cost** interface configuration command.

This example shows how to change the path cost on an interface to 20 in bridge group 10:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# bridge-group 10 path-cost 20
```

Adjusting BPDU Intervals

You can adjust BPDU intervals as described in these sections:

- Adjusting the Interval between Hello BPDUs, page 5-10 (optional)
- Changing the Forward-Delay Interval, page 5-10 (optional)
- Changing the Maximum-Idle Interval, page 5-11 (optional)

Note Each switch in a spanning tree adopts the interval between hello BPDUs, the forward delay interval, and the maximum idle interval parameters of the root switch, regardless of what its individual configuration might be.

Adjusting the Interval between Hello BPDUs

Beginning in privileged EXEC mode, follow these step to adjust the interval between hello BPDUs. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**bridge** *bridge-group* **hello-time** *seconds*	Specify the interval between hello BPDUs. • For *bridge-group*, specify the bridge group number. The range is 1 to 255. • For *seconds*, enter a number from 1 to 10. The default is 2 seconds.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entry.
Step 5	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

To return to the default setting, use the **no bridge** *bridge-group* **hello-time** global configuration command.

This example shows how to change the hello interval to 5 seconds in bridge group 10:

```
Switch(config)# bridge 10 hello-time 5
```

Changing the Forward-Delay Interval

The forward-delay interval is the amount of time spent listening for topology change information after an interface has been activated for switching and before forwarding actually begins.

Beginning in privileged EXEC mode, follow these steps to change the forward-delay interval. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**bridge** *bridge-group* **forward-time** *seconds*	Specify the forward-delay interval. • For *bridge-group*, specify the bridge group number. The range is 1 to 255. • For *seconds*, enter a number from 10 to 200. The default is 20 seconds.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entry.
Step 5	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

To return to the default setting, use the **no bridge** *bridge-group* **forward-time** global configuration command.

This example shows how to change the forward-delay interval to 10 seconds in bridge group 10:

```
Switch(config)# bridge 10 forward-time 10
```

Changing the Maximum-Idle Interval

If a switch does not receive BPDUs from the root switch within a specified interval, it recomputes the spanning-tree topology.

Beginning in privileged EXEC mode, follow these steps to change the maximum-idle interval (maximum aging time). This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**bridge** *bridge-group* **max-age** *seconds*	Specify the interval that the switch waits to hear BPDUs from the root switch. • For *bridge-group*, specify the bridge group number. The range is 1 to 255. • For *seconds*, enter a number from 10 to 200. The default is 30 seconds.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show running-config**	Verify your entry.
Step 5	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

To return to the default setting, use the **no bridge** *bridge-group* **max-age** global configuration command.

This example shows how to change the maximum-idle interval to 30 seconds in bridge group 10:

```
Switch(config)# bridge 10 max-age 30
```

Disabling the Spanning Tree on an Interface

When a loop-free path exists between any two switched subnetworks, you can prevent BPDUs generated in one switching subnetwork from impacting devices in the other switching subnetwork, yet still permit switching throughout the network as a whole. For example, when switched LAN subnetworks are separated by a WAN, BPDUs can be prevented from traveling across the WAN link.

Beginning in privileged EXEC mode, follow these steps to disable spanning tree on an interface. This procedure is optional.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface, and enter interface configuration mode.
Step 3	**bridge-group** *bridge-group* **spanning-disabled**	Disable spanning tree on the interface. For *bridge-group*, specify the bridge group number. The range is 1 to 255.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show running-config**	Verify your entry.
Step 6	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

To re-enable spanning tree on the interface, use the **no bridge-group** *bridge-group* **spanning-disabled** interface configuration command.

This example shows how to disable spanning tree on an interface in bridge group 10:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# bridge group 10 spanning-disabled
```

Monitoring and Maintaining Fallback Bridging

To monitor and maintain fallback bridging, use one or more of the privileged EXEC commands in Table 5-2:

Table 5-2 Commands for Monitoring and Maintaining Fallback Bridging

Command	Purpose
clear bridge *bridge-group*	Removes any learned entries from the forwarding database and clears the transmit and receive counts for any statically configured entries.
show bridge [*bridge-group*]	Displays details about the bridge group.
show bridge [*bridge-group*] [*interface-id*] [*address*] [**group**] [**verbose**]	Displays classes of entries in the bridge forwarding database.

For information about the fields in these displays, see the *Cisco IOS Bridging and IBM Networking Command Reference, Volume 1 of 2, Release 12.2*.

CHAPTER 6

Troubleshooting

This chapter describes how to identify and resolve Catalyst 3550 software problems related to the Cisco IOS software. Depending on the nature of the problem, you can use the command-line interface (CLI), the device manager, or Network Assistant to identify and solve problems.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release and the *Cisco IOS Command Summary for Cisco IOS Release 12.1*.

This chapter consists of these sections:

- Using Recovery Procedures, page 6-1
- Preventing Autonegotiation Mismatches, page 6-10
- GBIC Module Security and Identification, page 6-10
- Diagnosing Connectivity Problems, page 6-11
- Troubleshooting Power over Ethernet Switch Ports, page 6-16
- Using Debug Commands, page 6-16
- Using the show forward Command, page 6-19
- Using the crashinfo File, page 6-20

Note If after applying ACLs, you are experiencing packet performance problems or receiving messages about TCAM capacity, see the "Displaying ACL Resource Usage and Configuration Problems" section on page 29-42.

Using Recovery Procedures

These recovery procedures require that you have physical access to the switch:

- Recovering from a Software Failure, page 6-2
- Recovering from a Lost or Forgotten Password, page 6-2
- Recovering from a Command Switch Failure, page 6-6
- Recovering from Lost Member Connectivity, page 6-10

Recovering from a Software Failure

Switch software can be corrupted during an upgrade, by downloading the wrong file to the switch, and by deleting the image file. In all of these cases, the switch does not pass the power-on self-test (POST), and there is no connectivity.

This procedure uses the Xmodem Protocol to recover from a corrupt or wrong image file. There are many software packages that support the Xmodem Protocol, and this procedure is largely dependent on the emulation software that you are using.

Follow these steps to recover from a software failure:

Step 1 Connect a PC with terminal-emulation software supporting the XMODEM Protocol to the switch console port.

Step 2 Set the line speed on the emulation software to 9600 baud.

Step 3 Unplug the switch power cord.

Step 4 Press the **Mode** button, and at the same time, reconnect the power cord to the switch.

You can release the **Mode** button a second or two after the LED above port 1X turns off. Several lines of information about the software appear along with instructions:

```
The system has been interrupted prior to initializing the flash file system. The following
commands will initialize the flash file system, and finish loading the operating system
software#

flash_init
load_helper
boot
```

Step 5 Initialize the flash file system:

```
switch# flash_init
```

Step 6 If you had set the console port speed to anything other than 9600, it has been reset to that particular speed. Change the emulation software line speed to match that of the switch console port.

Step 7 Load any helper files:

```
switch# load_helper
```

Step 8 Start the file transfer by using the XMODEM Protocol.

```
switch# copy xmodem: flash:image_filename.bin
```

Step 9 After the XMODEM request appears, use the appropriate command on the terminal-emulation software to start the transfer and to copy the software image into flash memory.

Recovering from a Lost or Forgotten Password

An end user with physical access to the switch can recover from a lost password by interrupting the boot process during power-on and by entering a new password. This is the default configuration for Catalyst 3550 switches.

Note On Catalyst 3550 Fast Ethernet switches, a system administrator can disable some of the functionality of this feature by allowing an end user to reset a password only by agreeing to return to the default configuration. If you are an end user trying to reset a password on a Catalyst 3550 Fast Ethernet switch and password recovery has been disabled, a status message shows this during the recovery process.

Follow these steps if you have forgotten or lost the switch password.

Step 1 Connect a terminal or PC with terminal-emulation software to the switch console port.

Step 2 Set the line speed on the emulation software to 9600 baud.

Step 3 Unplug the switch power cord.

Step 4 Reconnect the power cord to the switch and, within 15 seconds, press the **Mode** button while the System LED above port 1X is still flashing green. Continue pressing the **Mode** button until the System LED turns briefly amber and then solid green; then release the **Mode** button.

Several lines of information about the software appear with instructions, informing you if the password recovery procedure has been disabled or not.

- If you see a message that begins with this:

  The system has been interrupted prior to initializing the flash file system. The following commands will initialize the flash file system

  go to the "Password Recovery with Password Recovery Enabled" section on page 6-3, and follow the steps.

- If you see a message that begins with this:

  ```
  The password-recovery mechanism has been triggered, but is currently disabled.
  ```

  go to the "Procedure with Password Recovery Disabled" section on page 6-5, and follow the steps.

Password Recovery with Password Recovery Enabled

If the password-recovery mechanism is enabled, this message appears:

```
The system has been interrupted prior to initializing the flash file system. The following
commands will initialize the flash file system, and finish loading the operating system
software:

flash_init
load_helper
boot
```

Follow these steps when the password-recovery is enabled:

Step 1 Initialize the flash file system:

```
switch# flash_init
```

Step 2 If you had set the console port speed to anything other than 9600, it has been reset to that particular speed. Change the emulation software line speed to match that of the switch console port.

Step 3 Load any helper files:

```
switch# load_helper
```

Step 4 Display the contents of flash memory:

```
switch# dir flash:
```

The switch file system appears in the directory.

Step 5 Rename the configuration file to config.text.old.

This file contains the password definition.

```
switch# rename flash:config.text flash:config.text.old
```

Step 6 Boot the system:

```
switch# boot
```

You are prompted to start the setup program. Enter **N** at the prompt:

```
Continue with the configuration dialog? [yes/no]: N
```

Step 7 At the switch prompt, enter privileged EXEC mode:

```
Switch> enable
```

Step 8 Rename the configuration file to its original name:

```
Switch# rename flash:config.text.old flash:config.text
```

Step 9 Copy the configuration file into memory:

```
Switch# copy flash:config.text system:running-config
Source filename [config.text]?
Destination filename [running-config]?
```

Press **Return** in response to the confirmation prompts.

The configuration file is now reloaded, and you can change the password.

Step 10 Enter global configuration mode:

```
Switch# configure terminal
```

Step 11 Change the password:

```
Switch (config)# enable secret password
```

The secret password can be from 1 to 25 alphanumeric characters, can start with a number, is case sensitive, and allows spaces but ignores leading spaces.

Step 12 Return to privileged EXEC mode:

```
Switch (config)# exit
Switch#
```

Step 13 Write the running configuration to the startup configuration file:

```
Switch# copy running-config startup-config
```

The new password is now in the startup configuration.

Note This procedure is likely to leave your switch virtual interface in a shutdown state. You can see which interface is in this state by entering the **show running-config** privileged EXEC command. To re-enable the interface, enter the **interface vlan** *vlan-id* global configuration command, and specify the VLAN ID of the shutdown interface. With the switch in interface configuration mode, enter the **no shutdown** command.

Procedure with Password Recovery Disabled

If the password-recovery mechanism is disabled, this message appears:

```
The password-recovery mechanism has been triggered, but
is currently disabled.  Access to the boot loader prompt
through the password-recovery mechanism is disallowed at
this point.  However, if you agree to let the system be
reset back to the default system configuration, access
to the boot loader prompt can still be allowed.

Would you like to reset the system back to the default configuration (y/n)?
```

Caution Returning the switch to the default configuration results in the loss of all existing configurations. We recommend that you contact your system administrator to verify if there are backup switch and VLAN configuration files.

- If you enter **n** (no), the normal boot process continues as if the **Mode** button had not been pressed; you cannot access the boot loader prompt, and you cannot enter a new password. You see the message:

  ```
  Press Enter to continue........
  ```

- If you enter **y** (yes), the configuration file in flash memory and the VLAN database file are deleted. When the default configuration loads, you can reset the password.

Follow these steps when the password-recovery mechanism is disabled:

Step 1 Elect to continue with password recovery and lose the existing configuration:

```
Would you like to reset the system back to the default configuration (y/n)? Y
```

Step 2 Load any helper files:

```
Switch# load_helper
```

Step 3 Display the contents of flash memory:

```
switch# dir flash:
```

The switch file system appears in the directory.

Step 4 Boot the system:

```
Switch# boot
```

You are prompted to start the setup program. To continue with password recovery, enter **N** at the prompt:

```
Continue with the configuration dialog? [yes/no]: N
```

Step 5 At the switch prompt, enter privileged EXEC mode:

```
Switch> enable
```

Step 6 Enter global configuration mode:

```
Switch# configure terminal
```

Step 7 Change the password:

```
Switch (config)# enable secret password
```

The secret password can be from 1 to 25 alphanumeric characters, can start with a number, is case sensitive, and allows spaces but ignores leading spaces.

Step 8 Return to privileged EXEC mode:

```
Switch (config)# exit
Switch#
```

Step 9 Write the running configuration to the startup configuration file:

```
Switch# copy running-config startup-config
```

The new password is now in the startup configuration.

Note This procedure is likely to leave your switch virtual interface in a shutdown state. You can see which interface is in this state by entering the **show running-config** privileged EXEC command. To re-enable the interface, enter the **interface vlan** *vlan-id* global configuration command, and specify the VLAN ID of the shutdown interface. With the switch in interface configuration mode, enter the **no shutdown** command.

Step 10 You must now reconfigure the switch. If the system administrator has the backup switch and VLAN configuration files available, you should use those.

Recovering from a Command Switch Failure

This section describes how to recover from a failed command switch. You can configure a redundant command switch group by using the Hot Standby Router Protocol (HSRP). For more information, see Chapter 5, "Clustering Switches", Chapter 32, "Configuring HSRP", and the *Getting Started with Cisco Network Assistant*, available on Cisco.com.

Note HSRP is the preferred method for supplying redundancy to a cluster.

If you have not configured a standby command switch, and your command switch loses power or fails in some other way, management contact with the member switches is lost, and you must install a new command switch. However, connectivity between switches that are still connected is not affected, and the member switches forward packets as usual. You can manage the members as standalone switches through the service port, or, if they have IP addresses, through the other management interfaces.

You can prepare for a command switch failure by assigning an IP address to a member switch or another switch that is command-capable, making a note of the command-switch password, and cabling your cluster to have redundant connectivity between the member switches and the replacement command switch. This section describes two solutions for replacing a failed command switch:

- Replacing a failed command switch with a cluster member
- Replacing a failed command switch with another switch

For information on command-capable switches, see the release notes.

Replacing a Failed Command Switch with a Cluster Member

To replace a failed command switch with a command-capable member in the same cluster, follow these steps:

Step 1 Disconnect the command switch from the member switches, and physically remove it from the cluster.

Step 2 Insert the member switch in place of the failed command switch, and duplicate its connections to the cluster members.

Step 3 Start a CLI session on the new command switch.

You can access the CLI by using the serviceport or, if an IP address has been assigned to the switch, by using Telnet. For details about using the serviceport, see the switch hardware installation guide.

Step 4 At the switch prompt, enter privileged EXEC mode:

```
Switch> enable
Switch#
```

Step 5 Enter the password of the *failed command switch*.

Step 6 Enter global configuration mode.

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

Step 7 Remove the member switch from the cluster.

```
Switch(config)# no cluster commander-address
```

Step 8 Return to privileged EXEC mode.

```
Switch(config)# end
Switch#
```

Step 9 Use the manufacturing default configuration, or set up the switch through the management module.

Step 10 Use the setup program to configure the switch IP information. This program prompts you for IP address information and passwords. From privileged EXEC mode, enter **setup**, and press **Return**.

```
Switch# setup
         --- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]: y

At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.
```

```
Basic management setup configures only enough connectivity
for management of the system, extended setup will ask you
to configure each interface on the system

Would you like to enter basic management setup? [yes/no]:
```

Step 11 Enter **Y** at the first prompt.

The prompts in the setup program vary depending on the member switch that you selected to be the command switch:

```
Continue with configuration dialog? [yes/no]: y
```

or

```
Configuring global parameters:
```

If this prompt does not appear, enter **enable**, and press **Return**. Enter **setup**, and press **Return** to start the setup program.

Step 12 Respond to the questions in the setup program.

When prompted for the host name, recall that on a command switch, the host name is limited to 28 characters; on a member switch to 31 characters. Do not use *-n*, where *n* is a number, as the last character in a host name for any switch.

When prompted for the Telnet (virtual terminal) password, recall that it can be from 1 to 25 alphanumeric characters, is case sensitive, allows spaces, but ignores leading spaces.

Step 13 When prompted for the **enable secret** and **enable** passwords, enter the passwords of the *failed command switch* again.

Step 14 When prompted, make sure to enable the switch as the cluster command switch, and press **Return**.

Step 15 When prompted, assign a name to the cluster, and press **Return**.

The cluster name can be 1 to 31 alphanumeric characters, dashes, or underscores.

Step 16 After the initial configuration appears, verify that the addresses are correct.

Step 17 If the displayed information is correct, enter **Y**, and press **Return**.

If this information is not correct, enter **N**, press **Return**, and begin again at Step 9.

Step 18 Start your browser, and enter the IP address of the new command switch.

Step 19 From the Cluster menu, select **Add to Cluster** to display a list of candidate switches to add to the cluster.

Replacing a Failed Command Switch with Another Switch

To replace a failed command switch with a switch that is command-capable but not part of the cluster, follow these steps:

Step 1 Disconnect the command switch from the member switches, and physically remove it from the cluster.

Step 2 Insert the new switch in place of the failed command switch, and duplicate its connections to the cluster members.

Step 3 Start a CLI session on the new command switch.

You can access the CLI by using the serviceport or, if an IP address has been assigned to the switch, by using Telnet. For details about using the serviceport, see the switch hardware installation guide.

Step 4 At the switch prompt, enter privileged EXEC mode:

```
Switch> enable
Switch#
```

Step 5 Enter the password of the *failed command switch.*

Step 6 Use the manufacturing default configuration, or set up the switch through the management module.

Step 7 Use the setup program to configure the switch IP information.

This program prompts you for IP address information and passwords. From privileged EXEC mode, enter **setup**, and press **Return**.

```
Switch# setup
         --- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]: y

At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.

Basic management setup configures only enough connectivity
for management of the system, extended setup will ask you
to configure each interface on the system

Would you like to enter basic management setup? [yes/no]:
```

Step 8 Enter **Y** at the first prompt.

The prompts in the setup program vary depending on the switch you selected to be the command switch:

```
Continue with configuration dialog? [yes/no]: y
```

or

```
Configuring global parameters:
```

If this prompt does not appear, enter **enable**, and press **Return**. Enter **setup**, and press **Return** to start the setup program.

Step 9 Respond to the questions in the setup program.

When prompted for the host name, recall that on a command switch, the host name is limited to 28 characters. Do not use *-n*, where *n* is a number, as the last character in a host name for any switch.

When prompted for the Telnet (virtual terminal) password, recall that it can be from 1 to 25 alphanumeric characters, is case sensitive, allows spaces, but ignores leading spaces.

Step 10 When prompted for the **enable secret** and **enable** passwords, enter the passwords of the *failed command switch* again.

Step 11 When prompted, make sure to enable the switch as the cluster command switch, and press **Return**.

Step 12 When prompted, assign a name to the cluster, and press **Return**.

The cluster name can be 1 to 31 alphanumeric characters, dashes, or underscores.

Step 13 When the initial configuration displays, verify that the addresses are correct.

Step 14 If the displayed information is correct, enter **Y**, and press **Return**.

If this information is not correct, enter **N**, press **Return**, and begin again at Step 9.

Step 15 Start your browser, and enter the IP address of the new command switch.

Step 16 From the Cluster menu, select **Add to Cluster** to display a list of candidate switches to add to the cluster.

Recovering from Lost Member Connectivity

Some configurations can prevent the command switch from maintaining contact with member switches. If you are unable to maintain management contact with a member, and the member switch is forwarding packets normally, check for these conflicts:

- A member switch (Cisco Systems Intelligent Gigabit Ethernet Switch Module, Catalyst 3550, 3500 XL, 2955, 2950, 2940, 2900 XL, 2820, and 1900) cannot connect to the command switch through a port that is defined as a network port.
- Catalyst 3500 XL, Catalyst 2900 XL, Catalyst 2820, and Catalyst 1900 member switches must connect to the command switch through a port that belongs to the same management VLAN.
- A member switch (Cisco Systems Intelligent Gigabit Ethernet Switch Module, Catalyst 3550, Catalyst 3500 XL, Catalyst 2950, Catalyst 2940, Catalyst 2900 XL, Catalyst 2820, and Catalyst 1900) connected to the command switch through a secured port can lose connectivity if the port is disabled because of a security violation.

Preventing Autonegotiation Mismatches

The IEEE 802.3ab autonegotiation protocol manages the switch settings for speed (10 Mbps, 100 Mbps, and 1000 Mbps excluding GBIC ports) and duplex (half or full). There are situations when this protocol can incorrectly align these settings, reducing performance. A mismatch occurs under these circumstances:

- A manually set speed or duplex parameter is different from the manually set speed or duplex parameter on the connected port.
- A port is set to autonegotiate, and the connected port is set to full duplex with no autonegotiation.

To maximize switch performance and ensure a link, follow one of these guidelines when changing the settings for duplex and speed:

- Let both ports autonegotiate both speed and duplex.
- Manually set the speed and duplex parameters for the ports on both ends of the connection.

Note If a remote device does not autonegotiate, configure the duplex settings on the two ports to match. The speed parameter can adjust itself even if the connected port does not autonegotiate.

GBIC Module Security and Identification

Cisco Course Wave Division Multiplexer (CWDM) Gigabit Interface Converter (GBIC) modules have a serial EEPROM that contains the module serial number, the vendor name and ID, a unique security code, and cyclic redundancy check (CRC). When a CWDM GBIC module is inserted in the switch, the switch

software reads the EEPROM to check the serial number, vendor name and vendor ID, and recompute the security code and CRC. If the serial number, the vendor name or vendor ID, the security code, or CRC is invalid, the switch places the interface in an error-disabled state.

Note If you are using a non-Cisco CWDM GBIC module, remove the GBIC module from the switch, and replace it with a Cisco module.

After inserting a Cisco GBIC module, use the **errdisable recovery cause gbic-invalid** global configuration command to verify the port status, and enter a time interval for recovering from the error-disabled state. After the elapsed interval, the switch brings the interface out of the error-disabled state and retries the operation. For more information about the **errdisable recovery** command, see the command reference for this release.

Diagnosing Connectivity Problems

This section describes how to troubleshoot connectivity problems:

- Using Ping, page 6-11
- Using IP Traceroute, page 6-12
- Using Layer 2 Traceroute, page 6-14

Using Ping

This section consists of this information:

- Understanding Ping, page 6-11
- Executing Ping, page 6-11

Understanding Ping

The switch supports IP ping, which you can use to test connectivity to remote hosts. Ping sends an echo request packet to an address and waits for a reply. Ping returns one of these responses:

- Normal response—The normal response (*hostname* is alive) occurs in 1 to 10 seconds, depending on network traffic.
- Destination does not respond—If the host does not respond, a *no-answer* message is returned.
- Unknown host—If the host does not exist, an *unknown host* message is returned.
- Destination unreachable—If the default gateway cannot reach the specified network, a *destination-unreachable* message is returned.
- Network or host unreachable—If there is no entry in the route table for the host or network, a *network or host unreachable* message is returned.

Executing Ping

If you attempt to ping a host in a different IP subnetwork, you must define a static route to the network or have IP routing configured to route between those subnets. For more information, see Chapter 31, “Configuring IP Unicast Routing.”

IP routing is disabled by default on all switches. If you need to enable or configure IP routing, see Chapter 31, "Configuring IP Unicast Routing."

Beginning in privileged EXEC mode, use this command to ping another device on the network from the switch:

Command	Purpose
ping [ip] {*host* | *address*}	Ping a remote host through IP or by supplying the host name or network address.

Note Though other protocol keywords are available with the **ping** command, they are not supported in this release.

This example shows how to ping an IP host:

```
Switch# ping 172.20.52.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 172.20.52.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Switch#
```

Table 6-1 describes the possible ping character output.

Table 6-1 Ping Output Display Characters

Character	Description
!	Each exclamation point means receipt of a reply.
.	Each period means the network server timed out while waiting for a reply.
U	A destination unreachable error PDU was received.
C	A congestion experienced packet was received.
I	User interrupted test.
?	Unknown packet type.
&	Packet lifetime exceeded.

To end a ping session, enter the escape sequence (**Ctrl-^ X** by default). You enter the default by simultaneously pressing and releasing the **Ctrl**, **Shift**, and **6** keys, and then pressing the **X** key.

Using IP Traceroute

This section consists of this information:

- Understanding IP Traceroute, page 6-13
- Executing IP Traceroute, page 6-13

Understanding IP Traceroute

You can use IP traceroute to identify the path that packets take through the network on a hop-by-hop basis. The command output displays all network layer (Layer 3) devices, such as routers, that the traffic passes through on the way to the destination.

Your switches can participate as the source or destination of the **traceroute** privileged EXEC command and might or might not appear as a hop in the **traceroute** command output. If the switch is the destination of the traceroute, it appears as the final destination in the traceroute output. Intermediate switches do not show up in the traceroute output if they are only bridging the packet from one port to another within the same VLAN. However, if the intermediate switch is a multilayer switch that is routing a particular packet, this switch shows up as a hop in the traceroute output.

The **traceroute** privileged EXEC command uses the Time To Live (TTL) field in the IP header to cause routers and servers to generate specific return messages. Traceroute starts by sending a User Datagram Protocol (UDP) datagram to the destination host with the TTL field set to 1. If a router finds a TTL value of 1 or 0, it drops the datagram and sends back an Internet Control Message Protocol (ICMP) time-to-live-exceeded message to the sender. Traceroute determines the address of the first hop by examining the source address field of the ICMP time-to-live-exceeded message.

To identify the next hop, traceroute sends a UDP packet with a TTL value of 2. The first router decrements the TTL field by 1 and sends the datagram to the next router. The second router sees a TTL value of 1, discards the datagram, and returns the time-to-live-exceeded message to the source. This process continues until the TTL is incremented to a value large enough for the datagram to reach the destination host (or until the maximum TTL is reached).

To determine when a datagram reaches its destination, traceroute sets the UDP destination port number in the datagram to a very large value that the destination host is unlikely to be using. When a host receives a datagram destined to itself containing a destination port number that is unused locally, it sends an ICMP port unreachable error to the source. Because all errors except `port unreachable` errors come from intermediate hops, the receipt of a `port unreachable` error means this message was sent by the destination.

Executing IP Traceroute

Beginning in privileged EXEC mode, follow this step to trace the path packets take through the network:

Command	Purpose
traceroute ip *host*	Trace the path packets take through the network by using IP.

Note Though other protocol keywords are available with the **traceroute** privileged EXEC command, they are not supported in this release.

This example shows how to perform a **traceroute** to an IP host:

```
Switch# traceroute ip 171.9.15.10

Type escape sequence to abort.
Tracing the route to 171.69.115.10

  1 172.2.52.1 0 msec 0 msec 4 msec
  2 172.2.1.203 12 msec 8 msec 0 msec
  3 171.9.16.6 4 msec 0 msec 0 msec
```

```
  4 171.9.4.5 0 msec 4 msec 0 msec
  5 171.9.121.34 0 msec 4 msec 4 msec
  6 171.9.15.9 120 msec 132 msec 128 msec
  7 171.9.15.10 132 msec 128 msec 128 msec
Switch#
```

The display shows the hop count, the IP address of the router, and the round-trip time in milliseconds for each of the three probes that are sent.

Table 6-2 Traceroute Output Display Characters

Character	Description
*	The probe timed out.
?	Unknown packet type.
A	Administratively unreachable. Usually, this output means that an access list is blocking traffic.
H	Host unreachable.
N	Network unreachable.
P	Protocol unreachable.
Q	Source quench.
U	Port unreachable.

To end a trace in progress, enter the escape sequence (**Ctrl-^ X** by default). You enter the default by simultaneously pressing and releasing the **Ctrl**, **Shift**, and **6** keys, and then pressing the **X** key.

Using Layer 2 Traceroute

This section describes this information:

- Understanding Layer 2 Traceroute, page 6-14
- Usage Guidelines, page 6-15
- Displaying the Physical Path, page 6-15

Understanding Layer 2 Traceroute

The Layer 2 traceroute feature allows the switch to identify the physical path that a packet takes from a source device to a destination device. Layer 2 traceroute supports only unicast source and destination MAC addresses. It determines the path by using the MAC address tables of the switches in the path. When the switch detects a device in the path that does not support Layer 2 traceroute, the switch continues to send Layer 2 trace queries and lets them time out.

The switch can only identify the path from the source device to the destination device. It cannot identify the path that a packet takes from source host to the source device or from the destination device to the destination host.

Usage Guidelines

These are the Layer 2 traceroute usage guidelines:

- Cisco Discovery Protocol (CDP) must be enabled on all the devices in the network. For Layer 2 traceroute to functional properly, do not disable CDP. If any devices in the physical path are transparent to CDP, the switch cannot identify the path through these devices.

  Note For more information about enabling CDP, see Chapter 22, "Configuring CDP."

- A switch is reachable from another switch when you can test connectivity by using the **ping** privileged EXEC command. All switches in the physical path must be reachable from each other.
- The maximum number of hops identified in the path is ten.
- You can enter the **traceroute mac** or the **traceroute mac ip** privileged EXEC command on a switch that is not in the physical path from the source device to the destination device. All switches in the path must be reachable from this switch.
- The **traceroute mac** command output shows the Layer 2 path only when the specified source and destination MAC addresses belong to the same VLAN. If you specify source and destination MAC addresses that belong to different VLANs, the Layer 2 path is not identified, and an error message appears.
- If you specify a multicast source or destination MAC address, the path is not identified, and an error message appears.
- If the source or destination MAC address belongs to multiple VLANs, you must specify the VLAN to which both the source and destination MAC addresses belong. If the VLAN is not specified, the path is not identified, and an error message appears.
- The **traceroute mac ip** command output shows the Layer 2 path when the specified source and destination IP addresses belong to the same subnet. When you specify the IP addresses, the switch uses Address Resolution Protocol (ARP) to associate the IP addresses with the corresponding MAC addresses and the VLAN IDs.
  - If an ARP entry exists for the specified IP address, the switch uses the associated MAC address and identifies the physical path.
  - If an ARP entry does not exist, the switch sends an ARP query and tries to resolve the IP address. If the IP address is not resolved, the path is not identified, and an error message appears.
- When multiple devices are attached to one port through hubs (for example, multiple CDP neighbors are detected on a port), the Layer 2 traceroute feature is not supported. When more than one CDP neighbor is detected on a port, the Layer 2 path is not identified, and an error message appears.
- This feature is not supported in Token Ring VLANs.

Displaying the Physical Path

You can display physical path that a packet takes from a source device to a destination device by using one of these privileged EXEC commands:

- **traceroute mac** [**interface** *interface-id*] {*source-mac-address*} [**interface** *interface-id*] {*destination-mac-address*} [**vlan** *vlan-id*] [**detail**]
- **traceroute mac ip** {*source-ip-address* | *source-hostname*} {*destination-ip-address* | *destination-hostname*} [**detail**]

For more information, see the command reference for this release.

Troubleshooting Power over Ethernet Switch Ports

This section consists of this information:

- Disabled Port Caused by Power Loss, page 6-16
- Disabled Port Caused by False Link-Up, page 6-16

Disabled Port Caused by Power Loss

If a powered device (such as a Cisco IP Phone 7910) that is connected to a Power over Ethernet (PoE) switch port and is powered by an AC power source loses power from the AC power source, the device might enter an error-disabled state. To recover from an error-disabled state, enter the **shutdown** interface configuration command followed by the **no shutdown** interface command. You can also configure automatic recovery on the switch to recover from the error-disabled state. The **errdisable recovery cause loopback** and the **errdisable recovery interval** *seconds* global configuration commands automatically take the interface out of the error-disabled state after the specified period of time.

Use these commands, described in the command reference for this release, to monitor the PoE port status:

- **show controllers power inline** privileged EXEC command
- **show power inline** privileged EXEC command
- **debug ilpower** privileged EXEC command

Disabled Port Caused by False Link-Up

If a Cisco powered device is connected to a port and you configure the port by using the **power inline never** interface configuration command, a false link-up can occur on the port, placing it into an error-disabled state. To take the port out of the error-disabled state, enter the **shutdown** and the **no shutdown i**nterface configuration commands.

You should not connect a Cisco powered device to a port that has been configured with the **power inline never** command.

Using Debug Commands

This section explains how you use the **debug** commands to diagnose and resolve internetworking problems. It contains this information:

- Enabling Debugging on a Specific Feature, page 6-17
- Enabling All-System Diagnostics, page 6-17
- Redirecting Debug and Error Message Output, page 6-18
- Using the debug auto qos Command, page 6-18

Caution Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. It is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

Note For complete syntax and usage information for specific **debug** commands, see the command reference for this release.

Enabling Debugging on a Specific Feature

All **debug** commands are entered in privileged EXEC mode, and most **debug** commands take no arguments. For example, beginning in privileged EXEC mode, enter this command to enable the debugging for EtherChannel:

```
Switch# debug etherchannel
```

The switch continues to generate output until you enter the **no** form of the command.

If you enable a **debug** command and no output appears, consider these possibilities:

- The switch might not be properly configured to generate the type of traffic that you want to monitor. Use the **show running-config** command to verify the configuration.
- Even if the switch is properly configured, it might not generate the type of traffic that you want to monitor during the particular period that debugging is enabled. Depending on the feature you are debugging, you can use commands such as the TCP/IP **ping** command to generate network traffic.

To disable debugging of EtherChannel, enter this command in privileged EXEC mode:

```
Switch# no debug etherchannel
```

Alternately, in privileged EXEC mode, you can enter the **undebug** form of the command:

```
Switch# undebug etherchannel
```

To display the state of each debugging option, enter this command in privileged EXEC mode:

```
Switch# show debugging
```

Enabling All-System Diagnostics

Beginning in privileged EXEC mode, enter this command to enable all-system diagnostics:

```
Switch# debug all
```

Caution Because debugging output takes priority over other network traffic, and because the **debug all** privileged EXEC command generates more output than any other **debug** command, it can severely diminish switch performance or even render it unusable. In virtually all cases, it is best to use more specific **debug** commands.

The **no debug all** privileged EXEC command disables all diagnostic output. Using the **no debug all** command is a convenient way to ensure that you have not accidentally left any **debug** commands enabled.

Redirecting Debug and Error Message Output

By default, the network server sends the output from **debug** commands and system error messages to the console. If you use this default, you can use a virtual terminal connection to monitor debug output instead of connecting to the port.

Possible destinations include the console, virtual terminals, internal buffer, and UNIX hosts running a syslog server. The syslog format is compatible with 4.3 Berkeley Standard Distribution (BSD) UNIX and its derivatives.

Note Be aware that the debugging destination you use affects system overhead. Logging messages to the console produces very high overhead, whereas logging messages to a virtual terminal produces less overhead. Logging messages to a syslog server produces even less, and logging to an internal buffer produces the least overhead of any method.

For more information about system message logging, see Chapter 26, "Configuring System Message Logging."

Using the debug auto qos Command

You can use the **debug auto qos** privileged EXEC command to display quality of service (QoS) commands that are automatically generated when automatic-QoS (auto-QoS) is enabled.

Beginning in privileged EXEC mode, follow these steps to display the QoS commands and enable auto-QoS for voice over IP (VoIP) within a QoS domain:

	Command	Purpose
Step 1	debug auto qos	Enable debugging for auto-QoS. When debugging is enabled, the switch displays the QoS commands that are automatically generated when auto-QoS is enabled or disabled.
Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**interface** *interface-id*	Specify the interface that is connected to a Cisco IP Phone, and enter interface configuration mode. You also can specify the uplink interface that is connected to another switch or router in the interior of the network.
Step 4	**auto qos voip** {cisco-phone | trust}	Enable auto-QoS. The keywords have these meanings: • **cisco-phone**—If the interface is connected to a Cisco IP Phone, the QoS labels of incoming packets are trusted only when the Cisco IP phone is detected. • **trust**—The uplink interface is connected to a trusted switch or router, and the VoIP classification in the ingress packet is trusted.

	Command	Purpose
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show auto qos interface** *interface-id*	Verify your entries. This command displays the auto-QoS configuration that was initially applied; it does not display any user changes to the configuration that might be in effect.

For more information about auto-QoS, see the "Configuring Auto-QoS" section on page 28-17.

This example shows how to display the QoS commands that are automatically generated when auto-QoS is enabled:

```
Switch# debug auto qos
AutoQoS debugging is on
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# auto qos voip cisco-phone
```

Using the show forward Command

The output from the **show forward** privileged EXEC command has some useful information about the disposition of a packet entering an interface. Depending upon the parameters entered about the packet, the output shows lookup table results, maps and masks used to calculate forwarding destinations, bitmaps, and exit information.

Note For more syntax and usage information for the **show forward** command, see the command reference for this release.

This is an example of the output from the **show forward** privileged EXEC command for Fast Ethernet port 8, where VLAN ID, source and destination MAC addresses, and source and destination IP addresses were specified.

```
Switch# show forward fastethernet0/8 vlan 8 0000.1111.2222 0022.3355.9800 ip 8.8.8.10
4.4.4.33 255
signature:00000007, comparison ind:10, control info:2000941A control map:00000000
vlan:8, vlanid entry:000C0012 00000000 00000000 04620000
vlan:8, vlanid entry:000C0012 00000000 00000000 04620000

lookup  key                                 bk adata    rawoff secoff sec
qos     940808080A04040421 800000000000FF0000 0  00000000 006304 004064  4
acl     940808080A04040421 800000000000FF0000 1  00000082 045408 002016  1
learn   187008000011112222 801008002233559800 0  80010003 002176 002176  0
forw    187008000011112222 801008002233559800 1  40020000 043328 010560  5

bridgeDestMap:  00000000 00000000 0000FFFF FFFFFFC7
vlanMask:       00000000 00000000 0000FFFF FFFFFE7F
portMask:       00000000 00000000 00000000 00000080
sourceMask:     00000000 00000000 00000000 00000000
globalMap:      00000000 00000000 00000000 00000000
globalMask:     00000000 00000000 0002FFFF EFFFFC03
forwMap:        00000000 00000000 00000000 00000100
```

```
frame notifies:
src u_dat vlan fl q-map
2   00    8    00 00000000 00000000 00000000 00000100
Egress q 8
signature:00000007, comparison ind:10, control info:2000941A control map:00000000
vlan:8, vlanid entry:000C0012 00000000 00000000 04620000
FastEthernet0/9 vlan 8, dst 0022.3355.9800 src 0000.1111.2222, cos 0x0, dscp 0x0
```

Much of this information is useful mainly for Technical Support personnel, who have access to detailed information about the switch ASICs. However, you can look at the *Egress q* section to get information about the output interface. There is an egress section for each separate destination port. The important information is in the line containing the name of the output interface, output VLAN ID, and rewritten destination MAC address for the frame. The example shows that the output interface is Fast Ethernet port 9, that the output VLAN is VLAN 8, and shows the rewritten source and destination MAC address for the frame.

If the output interface is a trunk port that needs to send multiple copies of the frame on different VLANs (for example, for IP multicast frames), several lines might have the same output interface name, but different output VLANs. If output security access control lists (ACLs) are present, it is possible that one or more of these *Egress q* sections will not contain a line listing an output port. This happens when the output ACL denies the packet.

When the CPU is one of the destinations for a packet, a *Cpu q* section appears, followed by a queue name. This name should correspond to one of the queue names in the output from the **show controllers cpu-interface** privileged EXEC command, where statistics appear for the number of packets received at each queue.

This is an example of the *Cpu q* section display:

```
Cpu q:100 - routing queue
```

Using the crashinfo File

The crashinfo file saves information that helps Cisco technical support representatives to debug problems that caused the software image to fail (crash). The switch writes the crash information to the console at the time of the failure, and the file is created the next time you boot up the image after the failure (instead of while the system is failing).

The switch creates two types of crashinfo files:

- Basic crashinfo file—The switch automatically creates this file the next time you boot up the Cisco IOS image after the failure.
- Extended crashinfo file—In Cisco IOS Release 12.2(35)SE or later, the switch automatically creates this file when the system is failing.

Basic crashinfo Files

The information in the basic file includes the Cisco IOS image name and version that failed, a list of the processor registers, and a stack trace. You can provide this information to the Cisco technical support representative by using the **show tech-support** privileged EXEC command.

Basic crashinfo files are kept in this directory on the flash file system:

- flash:/crashinfo/.
- The filenames are crashinfo_*n* where *n* is a sequence number.

Each new crashinfo file that is created uses a sequence number that is larger than any previously existing sequence number, so the file with the largest sequence number describes the most recent failure. Version numbers are used instead of a timestamp because the switches do not include a real-time clock. You cannot change the name of the file that the system will use when it creates the file. However, after the file is created, you can use the **rename** privileged EXEC command to rename it, but the contents of the renamed file will not be displayed by the **show tech-support** privileged EXEC command. You can delete crashinfo files by using the **delete** privileged EXEC command.

You can display the most recent basic crashinfo file (that is, the file with the highest sequence number at the end of its filename) by entering the **show tech-support** privileged EXEC command. You also can access the file by using any command that can copy or display files, such as the **more** or the **copy** privileged EXEC command.

Extended crashinfo Files

In Cisco IOS Release 12.2(35)SE or later, the switch creates the extended crashinfo file when the system is failing. The information in the extended file includes additional information that can help determine the cause of the switch failure. You provide this information to the Cisco technical support representative by manually accessing the file and using the **more** or the **copy** privileged EXEC command.

Extended crashinfo files are kept in this directory on the flash file system:

flash:/crashinfo_ext/.

The filenames are crashinfo_ext_*n* where *n* is a sequence number.

You can configure the switch to not create the extended creashinfo file by using the **no exception crashinfo** global configuration command.

APPENDIX A

Supported MIBs

This appendix lists the supported MIBs for this release. It contains these sections:

- MIB List, page A-1
- Using FTP to Access the MIB Files, page A-3

MIB List

- BRIDGE-MIB (RFC1493)
- CISCO-BULK-FILE-MIB
- CISCO-CDP-MIB
- CISCO-CLUSTER-MIB
- CISCO_CONFIG_COPY_MIB
- CISCO-CONFIG-MAN-MIB
- CISCO-DHCP-SNOOPING-MIB
- CISCO-ENTITY-MIB
- CISCO_ENVMON_MIB
- CISCO-FLASH-MIB
- CISCO-FTP-CLIENT-MIB
- CISCO-HSRP-MIB
- CISCO-HSRP-EXT-MIB
- CISCO-IGMP-FILTER-MIB
- CISCO-IMAGE-MIB
- CISCO-IPMROUTE-MIB
- CISCO-L2L3-INTERFACE-MIB
- CISCO-MAC-NOTIFICATION-MIB
- CISCO-MEMORY-POOL-MIB
- CISCO-PAE-MIB
- CISCO-PAGP-MIB

- CISCO-PORT-QOS-MIB (only the packet counters are supported; the octet counters are not supported)
- CISCO-PORT-SECURITY-MIB
- CISCO-PROCESS-MIB
- CISCO-RTTMON-MIB (subsystems supported: sub_rtt_rmon and sub_rtt_rmonlib)
- CISCO-STACK-MIB (only a subset of the available MIB objects are implemented; not all objects are supported)
- CISCO_STACKMAKER_MIB
- CISCO-STP-EXTENSIONS-MIB
- CISCO-SYSLOG-MIB
- CISCO-TCP-MIB
- CISCO-VLAN-IFTABLE-RELATIONSHIP-MIB
- CISCO-VLAN-MEMBERSHIP-MIB
- CISCO-VTP-MIB
- ENTITY-MIB
- EXTENDED-BRIDGE-MIB
- IEEE8021-PAE-MIB
- IF-MIB (RFC 1573)
- IGMP-MIB
- IPMROUTE-MIB
- OLD-CISCO-CHASSIS-MIB
- OLD-CISCO-SYSTEM-MIB
- OLD-CISCO-TS-MIB
- PIM-MIB
- RFC1213-MIB
- RFC1253-MIB (OSPF)
- RMON-MIB (RFC 1757)
- RMON2-MIB
- SNMPv2-MIB
- TCP-MIB
- UDP-MIB

Using FTP to Access the MIB Files

You can get each MIB file by using this procedure:

Step 1 Make sure that your FTP client is in passive mode.

Note Some FTP clients do not support passive mode.

Step 2 Use FTP to access the server **ftp.cisco.com**.

Step 3 Log in with the username **anonymous**.

Step 4 Enter your e-mail username when prompted for the password.

Step 5 At the `ftp>` prompt, change directories to **/pub/mibs/v1** and the **/pub/mibs/v2**.

Step 6 Use the **get** *MIB_filename* command to obtain a copy of the MIB file.

You can check this URL for a list of MIBs supported by the Catalyst 3550 switch:

ftp://ftp.cisco.com/pub/mibs/supportlists/cat3550/cat3550-supportlist.html

You can also access information about MIBs on the Cisco website:

http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

APPENDIX **B**

Working with the Cisco IOS File System, Configuration Files, and Software Images

This appendix describes how to manipulate the Catalyst 3550 flash file system, how to copy configuration files, and how to archive (upload and download) software images.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release and the *Cisco IOS Configuration Fundamentals Command Reference, Release12.2*.

This appendix consists of these sections:

- Working with the Flash File System, page B-1
- Working with Configuration Files, page B-7
- Working with Software Images, page B-18

Working with the Flash File System

The flash file system on your switch provides several commands to help you manage software image and configuration files.

The flash file system is a single flash device on which you can store files. This flash device is called *flash:*.

This section contains this information:

- Displaying Available File Systems, page B-2
- Setting the Default File System, page B-3
- Displaying Information about Files on a File System, page B-3
- Creating and Removing Directories, page B-4
- Copying Files, page B-4
- Deleting Files, page B-5
- Creating, Displaying, and Extracting tar Files, page B-5
- Displaying the Contents of a File, page B-7

Displaying Available File Systems

To display the available file systems on your switch, use the **show file systems** privileged EXEC command as shown in this example:

```
Switch# show file systems
File Systems:

     Size(b)     Free(b)      Type  Flags  Prefixes
*   16128000    11118592     flash     rw   flash:
    16128000    11118592   unknown     rw   zflash:
       32768       26363     nvram     rw   nvram:
           -           -   network     rw   tftp:
           -           -    opaque     rw   null:
           -           -    opaque     rw   system:
           -           -    opaque     ro   xmodem:
           -           -    opaque     ro   ymodem:
           -           -   network     rw   rcp:
           -           -   network     rw   ftp:
```

Table B-1 show file systems Field Descriptions

Field	Value
Size(b)	Amount of memory in the file system in bytes.
Free(b)	Amount of free memory in the file system in bytes.
Type	Type of file system. **flash**—The file system is for a flash memory device. **nvram**—The file system is for an NVRAM device. **opaque**—The file system is a locally generated *pseudo* file system (for example, the *system*) or a download interface, such as brimux. **unknown**—The file system is an unknown type.
Flags	Permission for file system. **ro**—read-only. **rw**—read/write. **wo**—write-only.
Prefixes	Alias for file system. **flash:**—Flash file system. **nvram:**—NVRAM. **null:**—Null destination for copies. You can copy a remote file to null to determine its size. **rcp:**—Remote Copy Protocol (RCP) network server. **system:**—Contains the system memory, including the running configuration. **tftp:**—TFTP network server. **xmodem:**—Obtain the file from a network machine by using the Xmodem protocol. **ymodem:**—Obtain the file from a network machine by using the Ymodem protocol. **zflash:**—Read-only file decompression file system, which mirrors the contents of the flash file system.

Setting the Default File System

You can specify the file system or directory that the system uses as the default file system by using the **cd** *filesystem:* privileged EXEC command. You can set the default file system to omit the *filesystem:* argument from related commands. For example, for all privileged EXEC commands that have the optional *filesystem:* argument, the system uses the file system specified by the **cd** command.

By default, the default file system is *flash:*.

You can display the current default file system as specified by the **cd** command by using the **pwd** privileged EXEC command.

Displaying Information about Files on a File System

You can view a list of the contents of a file system before manipulating its contents. For example, before copying a new configuration file to flash memory, you might want to verify that the file system does not already contain a configuration file with the same name. Similarly, before copying a flash configuration file to another location, you might want to verify its filename for use in another command.

To display information about files on a file system, use one of the privileged EXEC commands in Table B-2:

Table B-2 Commands for Displaying Information About Files

Command	Description
dir [**/all**] [*filesystem***:**][*filename*]	Display a list of files on a file system.
show file systems	Display more information about each of the files on a file system.
show file information *file-url*	Display information about a specific file.
show file descriptors	Display a list of open file descriptors. File descriptors are the internal representations of open files. You can use this command to see if another user has a file open.

Changing Directories and Displaying the Working Directory

Beginning in privileged EXEC mode, follow these steps to change directories and display the working directory.

	Command	Purpose
Step 1	**dir** *filesystem***:**	Display the directories on the specified file system. For *filesystem***:**, use **flash:** for the system board flash device.
Step 2	**cd new_configs**	Change to the directory of interest. The command example shows how to change to the directory named *new_configs*.
Step 3	**pwd**	Display the working directory.

Creating and Removing Directories

Beginning in privileged EXEC mode, follow these steps to create and remove a directory:

	Command	Purpose
Step 1	**dir** *filesystem***:**	Display the directories on the specified file system. For *filesystem***:**, use **flash:** for the system board flash device.
Step 2	**mkdir old_configs**	Create a new directory. The command example shows how to create the directory named *old_configs*. Directory names are case sensitive. Directory names are limited to 45 characters between the slashes (/); the name cannot contain control characters, spaces, deletes, slashes, quotes, semicolons, or colons.
Step 3	**dir** *filesystem***:**	Verify your entry.

To delete a directory with all its files and subdirectories, use the **delete /force /recursive** *filesystem***:/***file-url* privileged EXEC command.

Use the **/recursive** keyword to delete the named directory and all subdirectories and the files contained in it. Use the **/force** keyword to suppress the prompting that confirms a deletion of each file in the directory. You are prompted only once at the beginning of this deletion process. Use the **/force** and **/recursive** keywords for deleting old software images that were installed by using the **archive download-sw** command but are no longer needed.

For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the name of the directory to be deleted. All the files in the directory and the directory are removed.

Caution When files and directories are deleted, their contents cannot be recovered.

Copying Files

To copy a file from a source to a destination, use the **copy** [**/erase**] *source-url destination-url* privileged EXEC command. For the source and destination URLs, you can use **running-config** and **startup-config** keyword shortcuts. For example, the **copy running-config startup-config** command saves the currently running configuration file to the NVRAM section of flash memory to be used as the configuration during system initialization.

You can also copy to and from special file systems (**xmodem:**, **ymodem:**) as the source or destination for the file from a network machine that uses the Xmodem or Ymodem protocol.

Network file system URLs include **ftp:**, **rcp:**, and **tftp:** and have these syntaxes:

FTP—**ftp:**[[**//***username* [**:***password*]**@***location*]**/***directory*]**/***filename*

Remote Copy Protocol (RCP)—**rcp:**[[**//***username***@***location*]**/***directory*]**/***filename*

TFTP—**tftp:**[[**//***location*]**/***directory*]**/***filename*

Local writable file systems include flash:.

Some invalid combinations of source and destination exist. Specifically, you cannot copy these combinations:

- From a running configuration to a running configuration
- From a startup configuration to a startup configuration
- From a device to the same device (for example, the **copy flash: flash:** command is invalid)

For specific examples of using the **copy** command with configuration files, see the "Working with Configuration Files" section on page B-7.

To copy software images either by downloading a new version or uploading the existing one, use the **archive download-sw** or the **archive upload-sw** privileged EXEC command. For more information, see the "Working with Software Images" section on page B-18.

Deleting Files

When you no longer need a file on a flash memory device, you can permanently delete it. To delete a file or directory from a specified flash device, use the **delete** [**/force**] [**/recursive**] [*filesystem***:**]*/file-url* privileged EXEC command.

Use the **/recursive** keyword for deleting a directory and all subdirectories and the files contained in it. Use the **/force** keyword to suppress the prompting that confirms a deletion of each file in the directory. You are prompted only once at the beginning of this deletion process. Use the **/force** and **/recursive** keywords for deleting old software images that were installed by using the **archive download-sw** command but are no longer needed.

If you omit the *filesystem***:** option, the switch uses the default device specified by the **cd** command. For *file-url*, you specify the path (directory) and the name of the file to be deleted.

If you attempt to delete the file specified by the CONFIG_FILE or BOOT environment variable, the system prompts you to confirm the deletion. If you attempt to delete the last valid system image specified in the BOOT environment variable, the system prompts you to confirm the deletion.

Caution When files are deleted, their contents cannot be recovered.

This example shows how to delete the file *myconfig* from the default flash memory device:

```
Switch# delete myconfig
```

Creating, Displaying, and Extracting tar Files

You can create a tar file and write files into it, list the files in a tar file, and extract the files from a tar file as described in the next sections.

Creating a tar File

To create a tar file and write files into it, use the privileged EXEC command:

archive tar /create *destination-url* **flash:/***file-url*

For *destination-url*, specify the destination URL alias for the local or network file system and the name of the tar file to create. These options are supported:

- For the local flash file system, the syntax is
  flash:
- For the FTP, the syntax is **ftp:**[[*//username*[**:***password*]*@location*]*/directory]/tar-filename***.tar**
- For the RCP, the syntax is **rcp:**[[*//username@location*]*/directory*]*/tar-filename***.tar**
- For the TFTP, the syntax is **tftp:**[[*//location*]*/directory*]*/tar-filename***.tar**

The *tar-filename***.tar** is the tar file to be created.

For **flash:/***file-url*, specify the location on the local flash file system from which the new tar file is created. You can also specify an optional list of files or directories within the source directory to write to the new tar file. If none are specified, all files and directories at this level are written to the newly created tar file.

This example shows how to create a tar file. This command writes the contents of the *new-configs* directory on the local flash device to a file named *saved.tar* on the TFTP server at 172.20.10.30:

```
Switch# archive tar /create tftp:172.20.10.30/saved.tar flash:/new-configs
```

Displaying the Contents of a tar File

To display the contents of a tar file on the screen, use this privileged EXEC command:

archive tar /table *source-url*

For *source-url*, specify the source URL alias for the local or network file system. These options are supported:

- For the local flash file system, the syntax is
  flash:
- For the FTP, the syntax is **ftp:**[[*//username*[**:***password*]*@location*]*/directory*]*/tar-filename***.tar**
- For the RCP, the syntax is **rcp:**[[*//username@location*]*/directory*]*/tar-filename***.tar**
- For the TFTP, the syntax is **tftp:**[[*//location*]*/directory*]*/tar-filename***.tar**

The *tar-filename***.tar** is the tar file to display.

You can also limit the display of the files by specifying an optional list of files or directories after the tar file; then only these files are displayed. If none are specified, all files and directories are displayed.

This example shows how to display the contents of the *c3550-ipservices-tar.122-25.SEB.tar* file that is in flash memory:

```
Switch# archive tar /table flash:c3550-ipservices-tar.122-25.SEB.tar
info (219 bytes)
c3550-ipservices-mz.122-25.SEB/ (directory)
c3550-ipservices-mz.122-25.SEB/html/ (directory)
c3550-ipservices-mz.122-25.SEB/c3550-ipservices-mz.122-25.SEB.bin (6074880 bytes)
c3550-ipservices-mz.122-25.SEB/info (219 bytes)
info.ver (219 bytes)
```

This example shows how to display only the *c3550-ipservices-mz.122-25.SEB/html* directory and its contents:

```
Switch# archive tar /table flash:c3550-ipservices-mz.122-25.SEB.tar
c3550-ipservices-mz.122-25.SEB/html
c3550-ipservices-mz.122-25.SEB/html/ (directory)
c3550-ipservices-mz.122-25SEB/html/const.htm (556 bytes)
c3550-ipservices-mz.122-25SEB/html/xhome.htm (9373 bytes)
```

```
c3550-ipservices-mz.122-25SEB/html/menu.css (1654 bytes)
<output truncated>
```

Extracting a tar File

To extract a tar file into a directory on the flash file system, use the privileged EXEC command:

archive tar /xtract *source-url* **flash:/***file-url* [*dir/file...*]

For *source-url*, specify the source URL alias for the local file system. These options are supported:

- For the local flash file system, the syntax is
  flash:
- For the FTP, the syntax is **ftp:**[[**//***username*[**:***password*]@*location*]**/***directory*]**/***tar-filename***.tar**
- For the RCP, the syntax is **rcp:**[[**//***username*@*location*]**/***directory*]**/***tar-filename***.tar**
- For the TFTP, the syntax is **tftp:**[[**//***location*]**/***directory*]**/***tar-filename***.tar**

The *tar-filename***.tar** is the tar file from which to extract files.

For **flash:/***file-url* [*dir/file...*], specify *t*he location on the local flash file system into which the tar file is extracted. Use the *dir/file...* option to specify an optional list of files or directories within the tar file to be extracted. If none are specified, all files and directories are extracted.

This example shows how to extract the contents of a tar file located on the TFTP server at 172.20.10.30. This command extracts just the *new-configs* directory into the root directory on the local flash file system. The remaining files in the *saved.tar* file are ignored.

```
Switch# archive tar /xtract tftp://172.20.10.30/saved.tar flash:/new-configs
```

Displaying the Contents of a File

To display the contents of any readable file, including a file on a remote file system, use the **more** [**/ascii** | **/binary** | **/ebcdic**] *file-url* privileged EXEC command:

This example shows how to display the contents of a configuration file on a TFTP server:

```
Switch# more tftp://serverA/hampton/savedconfig
!
! Saved configuration on server
!
version 11.3
service timestamps log datetime localtime
service linenumber
service udp-small-servers
service pt-vty-logging
!

<output truncated>
```

Working with Configuration Files

This section describes how to create, load, and maintain configuration files. You can create a basic configuration file by using the **setup** program or by entering the **setup** privileged EXEC command. For more information, see Chapter 3, “Assigning the Switch IP Address and Default Gateway.”

You can copy (*download*) configuration files from a TFTP, FTP, or RCP server to the running configuration or startup configuration of the switch. You might want to perform this for one of these reasons:

- To restore a backed-up configuration file.
- To use the configuration file for another switch. For example, you might add another switch to your network and want it to have a configuration similar to the original switch. By copying the file to the new switch, you can change the relevant parts rather than recreating the whole file.
- To load the same configuration commands on all the switches in your network so that all the switches have similar configurations.

You can copy (*upload*) configuration files from the switch to a file server by using TFTP, FTP, or RCP. You might perform this task to back up a current configuration file to a server before changing its contents so that you can later restore the original configuration file from the server.

The protocol that you use depends on which type of server you are using. The FTP and RCP transport mechanisms provide faster performance and more reliable delivery of data than TFTP. These improvements are possible because FTP and RCP are built on and use the TCP/IP stack, which is connection-oriented.

This section includes this information:

- Guidelines for Creating and Using Configuration Files, page B-8
- Configuration File Types and Location, page B-9
- Creating a Configuration File By Using a Text Editor, page B-9
- Copying Configuration Files By Using TFTP, page B-9
- Copying Configuration Files By Using FTP, page B-12
- Copying Configuration Files By Using RCP, page B-15
- Clearing Configuration Information, page B-18

Guidelines for Creating and Using Configuration Files

Creating configuration files can aid in your switch configuration. Configuration files can contain some or all of the commands needed to configure one or more switches. For example, you might want to download the same configuration file to several switches that have the same hardware configuration.

Use these guidelines when creating a configuration file:

- We recommend that you connect through the console port when using configuration files to configure the switch. If you configure the switch from a Telnet session, IP addresses are not changed, and ports and modules are not disabled.
- If no passwords have been set on the switch, you must set them on each switch by entering the **enable secret** *secret-password* global configuration command. Enter a blank line for this command. The password is saved in the configuration file as clear text.
- If passwords already exist, you cannot enter the **enable secret** *secret-password* global configuration command in the file because the password verification will fail. If you enter a password in the configuration file, the switch mistakenly attempts to execute the passwords as commands as it executes the file.

Note The **copy** {**ftp:** | **rcp:** | **tftp:**} **system:running-config** privileged EXEC command loads the configuration files on the switch as if you were entering the commands at the command line. The switch does not erase the existing running configuration before adding the commands. If a command in the copied configuration file replaces a command in the existing configuration file, the existing command is erased. For example, if the copied configuration file contains a different IP address in a particular command than the existing configuration, the IP address in the copied configuration is used. However, some commands in the existing configuration might not be replaced or negated. In this case, the resulting configuration file is a mixture of the existing configuration file and the copied configuration file, with the copied configuration file having precedence.

To restore a configuration file to an exact copy of a file stored on a server, copy the configuration file directly to the startup configuration (by using the **copy** {**ftp:** | **rcp:** | **tftp:**} **nvram:startup-config** privileged EXEC command), and reload the switch.

Configuration File Types and Location

Startup configuration files are used during system startup to configure the software. Running configuration files contain the current configuration of the software. The two configuration files can be different. For example, you might want to change the configuration for a short time period rather than permanently. In this case, you would change the running configuration but not save the configuration by using the **copy running-config startup-config** privileged EXEC command.

The running configuration is saved in DRAM; the startup configuration is stored in the NVRAM section of flash memory.

Creating a Configuration File By Using a Text Editor

When creating a configuration file, you must list commands logically so that the system can respond appropriately. This is one method of creating a configuration file:

Step 1 Copy an existing configuration from a switch to a server.

For more information, see the "Downloading the Configuration File By Using TFTP" section on page B-10, the "Downloading a Configuration File By Using FTP" section on page B-13, or the "Downloading a Configuration File By Using RCP" section on page B-16.

Step 2 Open the configuration file in a text editor, such as vi or emacs on UNIX or Notepad on a PC.

Step 3 Extract the portion of the configuration file with the desired commands, and save it in a new file.

Step 4 Copy the configuration file to the appropriate server location. For example, copy the file to the TFTP directory on the workstation (usually /tftpboot on a UNIX workstation).

Step 5 Make sure the permissions on the file are set to world-read.

Copying Configuration Files By Using TFTP

You can configure the switch by using configuration files you create, download from another switch, or download from a TFTP server. You can copy (upload) configuration files to a TFTP server for storage.

This section includes this information:

- Preparing to Download or Upload a Configuration File By Using TFTP, page B-10
- Downloading the Configuration File By Using TFTP, page B-10
- Uploading the Configuration File By Using TFTP, page B-11

Preparing to Download or Upload a Configuration File By Using TFTP

Before you begin downloading or uploading a configuration file by using TFTP, do these tasks:

- Ensure that the workstation acting as the TFTP server is properly configured. On a Sun workstation, make sure that the /etc/inetd.conf file contains this line:

  ```
  tftp dgram udp wait root /usr/etc/in.tftpd in.tftpd -p -s /tftpboot
  ```

  Make sure that the /etc/services file contains this line:

  ```
  tftp 69/udp
  ```

  Note You must restart the inetd daemon after modifying the /etc/inetd.conf and /etc/services files. To restart the daemon, either stop the inetd process and restart it, or enter a **fastboot** command (on the SunOS 4.x) or a **reboot** command (on Solaris 2.x or SunOS 5.x). For more information on the TFTP daemon, see the documentation for your workstation.

- Ensure that the switch has a route to the TFTP server. The switch and the TFTP server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the TFTP server by using the **ping** command.
- Ensure that the configuration file to be downloaded is in the correct directory on the TFTP server (usually /tftpboot on a UNIX workstation).
- For download operations, ensure that the permissions on the file are set correctly. The permission on the file should be world-read.
- Before uploading the configuration file, you might need to create an empty file on the TFTP server. To create an empty file, enter the **touch** *filename* command, where *filename* is the name of the file you will use when uploading it to the server.
- During upload operations, if you are overwriting an existing file (including an empty file, if you had to create one) on the server, ensure that the permissions on the file are set correctly. Permissions on the file should be world-write.

Downloading the Configuration File By Using TFTP

To configure the switch by using a configuration file downloaded from a TFTP server, follow these steps:

Step 1 Copy the configuration file to the appropriate TFTP directory on the workstation.

Step 2 Verify that the TFTP server is properly configured by referring to the "Preparing to Download or Upload a Configuration File By Using TFTP" section on page B-10.

Step 3 Log into the switch through the console port or a Telnet session.

Step 4 Download the configuration file from the TFTP server to configure the switch.

Specify the IP address or host name of the TFTP server and the name of the file to download.

Use one of these privileged EXEC commands:

- **copy tftp:**[[[*//location*]*/directory*]*/filename*] **system:running-config**
- **copy tftp:**[[[*//location*]*/directory*]*/filename*] **nvram:startup-config**

The configuration file downloads, and the commands are executed as the file is parsed line-by-line.

This example shows how to configure the software from the file *tokyo-confg* at IP address 172.16.2.155:

```
Switch# copy tftp://172.16.2.155/tokyo-confg system:running-config
Configure using tokyo-confg from 172.16.2.155? [confirm] y
Booting tokyo-confg from 172.16.2.155:!!! [OK - 874/16000 bytes]
```

Uploading the Configuration File By Using TFTP

To upload a configuration file from a switch to a TFTP server for storage, follow these steps:

Step 1 Verify that the TFTP server is properly configured by referring to the "Preparing to Download or Upload a Configuration File By Using TFTP" section on page B-10.

Step 2 Log into the switch through the console port or a Telnet session.

Step 3 Upload the switch configuration to the TFTP server. Specify the IP address or host name of the TFTP server and the destination filename.

Use one of these privileged EXEC commands:

- **copy system:running-config tftp:**[[[*//location*]*/directory*]*/filename*]
- **copy nvram:startup-config tftp:**[[[*//location*]*/directory*]*/filename*]

The file is uploaded to the TFTP server.

This example shows how to upload a configuration file from a switch to a TFTP server:

```
Switch# copy system:running-config tftp://172.16.2.155/tokyo-confg
Write file tokyo-confg on host 172.16.2.155? [confirm] y
#
Writing tokyo-confg!!! [OK]
```

Copying Configuration Files By Using FTP

You can copy configuration files to or from an FTP server.

The FTP protocol requires a client to send a remote username and password on each FTP request to a server. When you copy a configuration file from the switch to a server by using FTP, the software sends the first valid username in this list:

- The username specified in the **copy** command if a username is specified.
- The username set by the **ip ftp username** *username* global configuration command if the command is configured.
- Anonymous.

The switch sends the first valid password in this list:

- The password specified in the **copy** command if a password is specified.
- The password set by the **ip ftp password** *password* global configuration command if the command is configured.
- The switch forms a password named *username@switchname.domain*. The variable *username* is the username associated with the current session, *switchname* is the configured host name, and *domain* is the domain of the switch.

The username and password must be associated with an account on the FTP server. If you are writing to the server, the FTP server must be properly configured to accept your FTP write request.

Use the **ip ftp username** and **ip ftp password** commands to specify a username and password for all copies. Include the username in the **copy** command if you want to specify only a username for that copy operation.

If the server has a directory structure, the configuration file is written to or copied from the directory associated with the username on the server. For example, if the configuration file is in the home directory of a user on the server, specify that user's name as the remote username.

For more information, see the documentation for your FTP server.

This section includes this information:

- Preparing to Download or Upload a Configuration File By Using FTP, page B-12
- Downloading a Configuration File By Using FTP, page B-13
- Uploading a Configuration File By Using FTP, page B-14

Preparing to Download or Upload a Configuration File By Using FTP

Before you begin downloading or uploading a configuration file by using FTP, do these tasks:

- Ensure that the switch has a route to the FTP server. The switch and the FTP server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the FTP server by using the **ping** command.
- If you are accessing the switch through the console or a Telnet session and you do not have a valid username, make sure that the current FTP username is the one that you want to use for the FTP download. You can enter the **show users** privileged EXEC command to view the valid username. If you do not want to use this username, create a new FTP username by using the **ip ftp username** *username* global configuration command during all copy operations. The new username is stored in

NVRAM. If you are accessing the switch through a Telnet session and you have a valid username, this username is used, and you do not need to set the FTP username. Include the username in the **copy** command if you want to specify a username for only that copy operation.

- When you upload a configuration file to the FTP server, it must be properly configured to accept the write request from the user on the switch.

For more information, see the documentation for your FTP server.

Downloading a Configuration File By Using FTP

Beginning in privileged EXEC mode, follow these steps to download a configuration file by using FTP:

	Command	Purpose
Step 1		Verify that the FTP server is properly configured by referring to the "Preparing to Download or Upload a Configuration File By Using FTP" section on page B-12.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**configure terminal**	Enter global configuration mode on the switch. This step is required only if you override the default remote username or password (see Steps 4, 5, and 6).
Step 4	**ip ftp username** *username*	(Optional) Change the default remote username.
Step 5	**ip ftp password** *password*	(Optional) Change the default password.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**copy ftp:**[[[**//**[*username*[**:***password*]**@**]*location*]**/***directory*]**/***filename*] **system:running-config** or **copy ftp:**[[[**//**[*username*[**:***password*]**@**]*location*]**/***directory*]**/***filename*] **nvram:startup-config**	Using FTP, copy the configuration file from a network server to the running configuration or to the startup configuration file.

This example shows how to copy a configuration file named *host1-confg* from the *netadmin1* directory on the remote server with an IP address of 172.16.101.101 and to load and run those commands on the switch:

```
Switch# copy ftp://netadmin1:mypass@172.16.101.101/host1-confg system:running-config
Configure using host1-confg from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-confg:![OK]
Switch#
%SYS-5-CONFIG: Configured from host1-config by ftp from 172.16.101.101
```

This example shows how to specify a remote username of *netadmin1*. The software copies the configuration file *host2-confg* from the *netadmin1* directory on the remote server with an IP address of 172.16.101.101 to the switch startup configuration.

```
Switch# configure terminal
Switch(config)# ip ftp username netadmin1
Switch(config)# ip ftp password mypass
Switch(config)# end
Switch# copy ftp: nvram:startup-config
```

```
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[rtr2-confg]? host2-confg
Configure using host2-confg from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-confg:![OK]
[OK]
Switch#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by ftp from
172.16.101.101
```

Uploading a Configuration File By Using FTP

Beginning in privileged EXEC mode, follow these steps to upload a configuration file by using FTP:

	Command	Purpose
Step 1		Verify that the FTP server is properly configured by referring to the "Preparing to Download or Upload a Configuration File By Using FTP" section on page B-12.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**configure terminal**	Enter global configuration mode. This step is required only if you override the default remote username or password (see Steps 4, 5, and 6).
Step 4	**ip ftp username** *username*	(Optional) Change the default remote username.
Step 5	**ip ftp password** *password*	(Optional) Change the default password.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**copy system:running-config ftp:**[[[**//**[*username*[**:***password*]@]*location*]**/***directory*] **/***filename*] or **copy nvram:startup-config ftp:**[[[**//**[*username*[**:***password*]@]*location*]**/***directory*] **/***filename*]	Using FTP, store the switch running or startup configuration file to the specified location.

This example shows how to copy the running configuration file named *switch2-confg* to the *netadmin1* directory on the remote host with an IP address of 172.16.101.101:

```
Switch# copy system:running-config ftp://netadmin1:mypass@172.16.101.101/switch2-confg
Write file switch2-confg on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Switch#
```

This example shows how to store a startup configuration file on a server by using FTP to copy the file:

```
Switch# configure terminal
Switch(config)# ip ftp username netadmin2
Switch(config)# ip ftp password mypass
Switch(config)# end
Switch# copy nvram:startup-config ftp:
Remote host[]? 172.16.101.101
```

```
Name of configuration file to write [switch2-confg]?
Write file switch2-confg on host 172.16.101.101?[confirm]
![OK]
```

Copying Configuration Files By Using RCP

The RCP provides another method of downloading, uploading, and copying configuration files between remote hosts and the switch. Unlike TFTP, which uses User Datagram Protocol (UDP), a connectionless protocol, RCP uses TCP, which is connection-oriented.

To use RCP to copy files, the server from or to which you will be copying files must support RCP. The RCP copy commands rely on the rsh server (or daemon) on the remote system. To copy files by using RCP, you do not need to create a server for file distribution as you do with TFTP. You only need to have access to a server that supports the remote shell (rsh). (Most UNIX systems support rsh.) Because you are copying a file from one place to another, you must have read permission on the source file and write permission on the destination file. If the destination file does not exist, RCP creates it for you.

The RCP requires a client to send a remote username with each RCP request to a server. When you copy a configuration file from the switch to a server, the software sends the first valid username in this list:

- The username specified in the **copy** command if a username is specified.
- The username set by the **ip rcmd remote-username** *username* global configuration command if the command is configured.
- The remote username associated with the current TTY (terminal) process. For example, if the user is connected to the router through Telnet and was authenticated through the **username** command, the switch software sends the Telnet username as the remote username.
- The switch host name.

For a successful RCP copy request, you must define an account on the network server for the remote username. If the server has a directory structure, the configuration file is written to or copied from the directory associated with the remote username on the server. For example, if the configuration file is in the home directory of a user on the server, specify that user's name as the remote username.

This section includes this information:

- Preparing to Download or Upload a Configuration File By Using RCP, page B-15
- Downloading a Configuration File By Using RCP, page B-16
- Uploading a Configuration File By Using RCP, page B-17

Preparing to Download or Upload a Configuration File By Using RCP

Before you begin downloading or uploading a configuration file by using RCP, do these tasks:

- Ensure that the workstation acting as the RCP server supports the remote shell (rsh).
- Ensure that the switch has a route to the RCP server. The switch and the server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the RCP server by using the **ping** command.
- If you are accessing the switch through the console or a Telnet session and you do not have a valid username, make sure that the current RCP username is the one that you want to use for the RCP download. You can enter the **show users** privileged EXEC command to view the valid username. If you do not want to use this username, create a new RCP username by using the **ip rcmd remote-username** *username* global configuration command to be used during all copy operations. The new username is stored in NVRAM. If you are accessing the switch through a Telnet session

and you have a valid username, this username is used, and you do not need to set the RCP username. Include the username in the **copy** command if you want to specify a username for only that copy operation.

- When you upload a file to the RCP server, it must be properly configured to accept the RCP write request from the user on the switch. For UNIX systems, you must add an entry to the .rhosts file for the remote user on the RCP server. For example, suppose that the switch contains these configuration lines:

```
hostname Switch1
ip rcmd remote-username User0
```

If the switch IP address translates to *Switch1.company.com*, the .rhosts file for User0 on the RCP server should contain this line:

```
Switch1.company.com Switch1
```

For more information, see the documentation for your RCP server.

Downloading a Configuration File By Using RCP

Beginning in privileged EXEC mode, follow these steps to download a configuration file by using RCP:

	Command	Purpose
Step 1		Verify that the RCP server is properly configured by referring to the "Preparing to Download or Upload a Configuration File By Using RCP" section on page B-15.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**configure terminal**	Enter global configuration mode. This step is required only if you override the default remote username (see Steps 4 and 5).
Step 4	**ip rcmd remote-username** *username*	(Optional) Specify the remote username.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**copy rcp:**[[[**//**[*username*@]*location*]**/***directory*]**/***filename*] **system:running-config** or **copy rcp:**[[[**//**[*username*@]*location*]**/***directory*]**/***filename*] **nvram:startup-config**	Using RCP, copy the configuration file from a network server to the running configuration or to the startup configuration file.

This example shows how to copy a configuration file named *host1-confg* from the *netadmin1* directory on the remote server with an IP address of 172.16.101.101 and load and run those commands on the switch:

```
Switch# copy rcp://netadmin1@172.16.101.101/host1-confg system:running-config
Configure using host1-confg from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-confg:![OK]
Switch#
%SYS-5-CONFIG: Configured from host1-config by rcp from 172.16.101.101
```

This example shows how to specify a remote username of *netadmin1*. Then it copies the configuration file *host2-confg* from the *netadmin1* directory on the remote server with an IP address of 172.16.101.101 to the startup configuration:

```
Switch# configure terminal
Switch(config)# ip rcmd remote-username netadmin1
Switch(config)# end
Switch# copy rcp: nvram:startup-config
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[rtr2-confg]? host2-confg
Configure using host2-confg from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-confg:![OK]
[OK]
Switch#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by rcp from
172.16.101.101
```

Uploading a Configuration File By Using RCP

Beginning in privileged EXEC mode, follow these steps to upload a configuration file by using RCP:

	Command	Purpose
Step 1		Verify that the RCP server is properly configured by referring to the "Preparing to Download or Upload a Configuration File By Using RCP" section on page B-15.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**configure terminal**	Enter global configuration mode. This step is required only if you override the default remote username (see Steps 4 and 5).
Step 4	**ip rcmd remote-username** *username*	(Optional) Specify the remote username.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**copy system:running-config rcp:**[[[//[*username*@]*location*]/*directory*]/*filename*] or **copy nvram:startup-config rcp:**[[[//[*username*@]*location*]/*directory*]/*filename*]	Using RCP, copy the configuration file from a switch running or startup configuration file to a network server.

This example shows how to copy the running configuration file named *switch2-confg* to the *netadmin1* directory on the remote host with an IP address of 172.16.101.101:

```
Switch# copy system:running-config rcp://netadmin1@172.16.101.101/switch2-confg
Write file switch-confg on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Switch#
```

This example shows how to store a startup configuration file on a server:

```
Switch# configure terminal
Switch(config)# ip rcmd remote-username netadmin2
Switch(config)# end
Switch# copy nvram:startup-config rcp:
Remote host[]? 172.16.101.101
Name of configuration file to write [switch2-confg]?
Write file switch2-confg on host 172.16.101.101?[confirm]
![OK]
```

Clearing Configuration Information

You can clear the configuration information from the startup configuration. If you reboot the switch with no startup configuration, the switch enters the setup program so that you can reconfigure the switch with all new settings.

Clearing the Startup Configuration File

To clear the contents of your startup configuration, use the **erase nvram:** or the **erase startup-config** privileged EXEC command.

Caution You cannot restore the startup configuration file after it has been deleted.

Deleting a Stored Configuration File

To delete a saved configuration from flash memory, use the **delete flash:***filename* privileged EXEC command. Depending on the setting of the **file prompt** global configuration command, you might be prompted for confirmation before you delete a file. By default, the switch prompts for confirmation on destructive file operations. For more information about the **file prompt** command, see the *Cisco IOS Command Reference for Release 12.1*.

Caution You cannot restore a file after it has been deleted.

Working with Software Images

This section describes how to archive (download and upload) software image files, which contain the system software, Cisco IOS code, and the embedded device manager software.

You can download a switch image file from a TFTP, FTP, or RCP server to upgrade the switch software. If you do not have access to a TFTP server, you can download a software image file directly to your PC or workstation by using a web browser (HTTP) and then by using the device manager or Network Assistant to upgrade your switch. See the release notes for information about upgrading your switch by using a TFTP server or a web browser (HTTP).

You can replace the current image with the new one or keep the current image in flash memory after a download.

You upload a switch image file to a TFTP, FTP, or RCP server for backup purposes. You can use this uploaded image for future downloads to the same switch or another of the same type.

The protocol that you use depends on which type of server that you are using. The FTP and RCP transport mechanisms provide faster performance and more reliable delivery of data than TFTP. These improvements are possible because FTP and RCP are built on and use the TCP/IP stack, which is connection-oriented.

This section includes this information:

- Image Location on the Switch, page B-19
- tar File Format of Images on a Server or Cisco.com, page B-19
- Copying Image Files By Using TFTP, page B-20
- Copying Image Files By Using FTP, page B-23
- Copying Image Files By Using RCP, page B-27

Note For a list of software images and the supported upgrade paths, see the release notes.

Image Location on the Switch

The software image is stored as a *.bin* file in a directory that shows the version number. A subdirectory contains the HTML files needed for web management. The image is stored on the system board flash memory (flash:).

You can use the **show version** privileged EXEC command to see the software version that is currently running on your switch. In the display, check the line that begins with `System image file is...`. It shows the directory name in flash memory where the image is stored.

You can also use the **dir** *filesystem***:** privileged EXEC command to see the directory names of other software images you might have stored in flash memory.

tar File Format of Images on a Server or Cisco.com

Software images located on a server or downloaded from Cisco.com are provided in a tar file format, which contains these files:

- *info* file

  The info file is always at the beginning of the tar file and has information about the files within it.
- Cisco IOS image
- Web management files needed by the HTTP server on the switch
- *info.ver* file

  The info.ver file is always at the end of the tar file and has the same information as the info file. Because it is the last file in the tar file, its existence means that all files in the image have been downloaded.

This example shows the information in the info and info.ver files:

```
version_suffix: ipservices-122-25.SEB
version_directory: c3550-ipservices-mz.122-25.SEB
image_name: c3550-ipservices-mz.122-25.SEB.bin
ios_image_file_size: 6074880
total_image_file_size: 7736832
image_feature: IP|LAYER_3|SSH|3DES|MIN_DRAM_MEG=24
image_family: C3550
info_end:
```

Table B-3 info and info.ver File Description

Field	Description
version_suffix	Specifies the software image version string suffix
version_directory	Specifies the directory where the software image and the HTML subdirectory are installed
image_name	Specifies the name of the software image within the tar file
ios_image_file_size	Specifies the software image size in the tar file, which is an approximate measure of how much flash space is required to hold just the software image
total_image_file_size	Specifies the size of all the images (the software image and the HTML files) in the tar file, which is an approximate measure of how much flash space is required to hold them
image_feature	Describes the core functionality of the image
image_family	Describes the family of products on which the software can be installed
image_min_dram	Specifies the minimum amount of DRAM needed to run this image

Copying Image Files By Using TFTP

You can download a switch image from a TFTP server or upload the image from the switch to a TFTP server.

You download a switch image file from a server to upgrade the switch software. You can overwrite the current image with the new one or keep the current image after a download.

You upload a switch image file to a server for backup purposes; this uploaded image can be used for future downloads to the same or another switch of the same type.

This section includes this information:

- Preparing to Download or Upload an Image File By Using TFTP, page B-21
- Downloading an Image File By Using TFTP, page B-21
- Uploading an Image File By Using TFTP, page B-23

Preparing to Download or Upload an Image File By Using TFTP

Before you begin downloading or uploading an image file by using TFTP, do these tasks:

- Ensure that the workstation acting as the TFTP server is properly configured. On a Sun workstation, make sure that the /etc/inetd.conf file contains this line:

  ```
  tftp dgram udp wait root /usr/etc/in.tftpd in.tftpd -p -s /tftpboot
  ```

  Make sure that the /etc/services file contains this line:

  ```
  tftp 69/udp
  ```

  Note You must restart the inetd daemon after modifying the /etc/inetd.conf and /etc/services files. To restart the daemon, either stop the inetd process and restart it, or enter a **fastboot** command (on the SunOS 4.x) or a **reboot** command (on Solaris 2.x or SunOS 5.x). For more information on the TFTP daemon, see the documentation for your workstation.

- Ensure that the switch has a route to the TFTP server. The switch and the TFTP server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the TFTP server by using the **ping** command.
- Ensure that the image to be downloaded is in the correct directory on the TFTP server (usually /tftpboot on a UNIX workstation).
- For download operations, ensure that the permissions on the file are set correctly. The permission on the file should be world-read.
- Before uploading the image file, you might need to create an empty file on the TFTP server. To create an empty file, enter the **touch** *filename* command, where *filename* is the name of the file you will use when uploading the image to the server.
- During upload operations, if you are overwriting an existing file (including an empty file, if you had to create one) on the server, ensure that the permissions on the file are set correctly. Permissions on the file should be world-write.

Downloading an Image File By Using TFTP

You can download a new image file and replace the current image or keep the current image.

Beginning in privileged EXEC mode, follow Steps 1 through 3 to download a new image from a TFTP server and overwrite the existing image. To keep the current image, omit Step 3.

	Command	Purpose
Step 1		Copy the image to the appropriate TFTP directory on the workstation. Make sure that the TFTP server is properly configured; see the "Preparing to Download or Upload an Image File By Using TFTP" section on page B-21.
Step 2		Log into the switch through the console port or a Telnet session.

	Command	Purpose
Step 3	**archive download-sw /overwrite /reload tftp:**[[*//location*]*/directory*]*/image-name*.**tar**	Download the image file from the TFTP server to the switch, and overwrite the current image. • The **/overwrite** option overwrites the software image in flash with the downloaded image. • The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. • For **//***location*, specify the IP address of the TFTP server. • For */directory/image-name*.**tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.
Step 4	**archive download-sw /leave-old-sw /reload tftp:**[[*//location*]*/directory*]*/image-name*.**tar**	Download the image file from the TFTP server to the switch, and keep the current image. • The **/leave-old-sw** option keeps the old software version after a download. • The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. • For **//***location*, specify the IP address of the TFTP server. • For */directory/image-name*.**tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.

The download algorithm verifies that the image is appropriate for the switch model and that enough DRAM is present, or it stops the process and reports an error. If you specify the **/overwrite** option, the download algorithm removes the existing image on the flash device whether or not it is the same as the new one, downloads the new image, and then reloads the software.

Note If the flash device has sufficient space to hold two images and you want to overwrite one of these images with the same version, you must specify the **/overwrite** option.

If you specify the **/leave-old-sw**, the existing files are not removed. If there is not enough space to install the new image and keep the current running image, the download process stops, and an error message appears.

The algorithm installs the downloaded image on the system board flash device (flash:). The image is placed into a new directory named with the software version string, and the BOOT environment variable is updated to point to the newly installed image.

If you kept the old image during the download process (you specified the **/leave-old-sw** keyword), you can remove it by entering the **delete /force /recursive** *filesystem***:/***file-url* privileged EXEC command. For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the directory name of the old image. All the files in the directory and the directory are removed.

Caution For the download and upload algorithms to operate properly, do *not* rename image names.

Uploading an Image File By Using TFTP

You can upload an image from the switch to a TFTP server. You can later download this image to the switch or to another switch of the same type.

Use the upload feature only if the HTML pages associated with the device manager have been installed with the existing image.

Beginning in privileged EXEC mode, follow these steps to upload an image to a TFTP server:

	Command	Purpose
Step 1		Make sure that the TFTP server is properly configured; see the "Preparing to Download or Upload an Image File By Using TFTP" section on page B-21.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**archive upload-sw tftp:**[[*//location*]*/directory*]*/image-name***.tar**	Upload the currently running switch image to the TFTP server. • For *//location*, specify the IP address of the TFTP server. • For */directory/image-name***.tar**, specify the directory (optional) and the name of the software image to be uploaded. Directory and image names are case sensitive. The *image-name***.tar** is the name of the software image to be stored on the server.

The **archive upload-sw** privileged EXEC command builds an image file on the server by uploading these files in order: info, the Cisco IOS image, the HTML files, and info.ver. After these files are uploaded, the upload algorithm creates the tar file format.

Caution For the download and upload algorithms to operate properly, do *not* rename image names.

Copying Image Files By Using FTP

You can download a switch image from an FTP server or upload the image from the switch to an FTP server.

You download a switch image file from a server to upgrade the switch software. You can overwrite the current image with the new one or keep the current image after a download.

You upload a switch image file to a server for backup purposes. You can use this uploaded image for future downloads to the switch or another switch of the same type.

This section includes this information:

- Preparing to Download or Upload an Image File By Using FTP, page B-24
- Downloading an Image File By Using FTP, page B-25
- Uploading an Image File By Using FTP, page B-26

Preparing to Download or Upload an Image File By Using FTP

You can copy images files to or from an FTP server.

The FTP protocol requires a client to send a remote username and password on each FTP request to a server. When you copy an image file from the switch to a server by using FTP, the software sends the first valid username in this list:

- The username specified in the **archive download-sw** or **archive upload-sw** privileged EXEC command if a username is specified.
- The username set by the **ip ftp username** *username* global configuration command if the command is configured.
- Anonymous.

The switch sends the first valid password in this list:

- The password specified in the **archive download-sw** or **archive upload-sw** privileged EXEC command if a password is specified.
- The password set by the **ip ftp password** *password* global configuration command if the command is configured.
- The switch forms a password named *username@switchname.domain*. The variable *username* is the username associated with the current session, *switchname* is the configured host name, and *domain* is the domain of the switch.

The username and password must be associated with an account on the FTP server. If you are writing to the server, the FTP server must be properly configured to accept the FTP write request from you.

Use the **ip ftp username** and **ip ftp password** commands to specify a username and password for all copies. Include the username in the **archive download-sw** or **archive upload-sw** privileged EXEC command if you want to specify a username only for that operation.

If the server has a directory structure, the image file is written to or copied from the directory associated with the username on the server. For example, if the image file resides in the home directory of a user on the server, specify that user's name as the remote username.

Before you begin downloading or uploading an image file by using FTP, do these tasks:

- Ensure that the switch has a route to the FTP server. The switch and the FTP server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the FTP server by using the **ping** command.
- If you are accessing the switch through the console or a Telnet session and you do not have a valid username, make sure that the current FTP username is the one that you want to use for the FTP download. You can enter the **show users** privileged EXEC command to view the valid username. If you do not want to use this username, create a new FTP username by using the **ip ftp username** *username* global configuration command. This new name will be used during all archive operations. The new username is stored in NVRAM. If you are accessing the switch through a Telnet session and you have a valid username, this username is used, and you do not need to set the FTP username. Include the username in the **archive download-sw** or **archive upload-sw** privileged EXEC command if you want to specify a username for that operation only.
- When you upload an image file to the FTP server, it must be properly configured to accept the write request from the user on the switch.

For more information, see the documentation for your FTP server.

Downloading an Image File By Using FTP

You can download a new image file and overwrite the current image or keep the current image.

Beginning in privileged EXEC mode, follow Steps 1 through 7 to download a new image from an FTP server and overwrite the existing image. To keep the current image, omit Step 7.

	Command	Purpose
Step 1		Verify that the FTP server is properly configured by referring to the "Preparing to Download or Upload an Image File By Using FTP" section on page B-24.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**configure terminal**	Enter global configuration mode. This step is required only if you override the default remote username or password (see Steps 4, 5, and 6).
Step 4	**ip ftp username** *username*	(Optional) Change the default remote username.
Step 5	**ip ftp password** *password*	(Optional) Change the default password.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**archive download-sw /overwrite /reload ftp:**[[**//***username*[**:***password*]**@***location*]**/***directory*] **/***image-name***.tar**	Download the image file from the FTP server to the switch, and overwrite the current image. • The **/overwrite** option overwrites the software image in flash with the downloaded image. • The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. • For **//***username*[**:***password*], specify the username and password; these must be associated with an account on the FTP server. For more information, see the "Preparing to Download or Upload an Image File By Using FTP" section on page B-24. • For **@***location*, specify the IP address of the FTP server. • For *directory***/***image-name***.tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.

	Command	Purpose
Step 8	**archive download-sw /leave-old-sw /reload ftp:**[[*//username*[**:***password*]@*location*]*/directory*] */image-name***.tar**	Download the image file from the FTP server to the switch, and keep the current image. • The **/leave-old-sw** option keeps the old software version after a download. • The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. • For **//***username*[**:***password*], specify the username and password. These must be associated with an account on the FTP server. For more information, see the "Preparing to Download or Upload an Image File By Using FTP" section on page B-24. • For @*location*, specify the IP address of the FTP server. • For *directory/image-name***.tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.

The download algorithm verifies that the image is appropriate for the switch model and that enough DRAM is present, or it stops the process and reports an error. If you specify the **/overwrite** option, the download algorithm removes the existing image on the flash device, whether or not it is the same as the new one, downloads the new image, and then reloads the software.

Note If the flash device has sufficient space to hold two images and you want to overwrite one of these images with the same version, you must specify the **/overwrite** option.

If you specify the **/leave-old-sw**, the existing files are not removed. If there is not enough space to install the new image and keep the running image, the download process stops, and an error message appears.

The algorithm installs the downloaded image onto the system board flash device (flash:). The image is placed into a new directory named with the software version string, and the BOOT environment variable is updated to point to the newly installed image.

If you kept the old image during the download process (you specified the **/leave-old-sw** keyword), you can remove it by entering the **delete /force /recursive** *filesystem***:/***file-url* privileged EXEC command. For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the directory name of the old software image. All the files in the directory and the directory are removed.

Caution For the download and upload algorithms to operate properly, do *not* rename image names.

Uploading an Image File By Using FTP

You can upload an image from the switch to an FTP server. You can later download this image to the same switch or to another switch of the same type.

Use the upload feature only if the HTML pages associated with the device manager have been installed with the existing image.

Beginning in privileged EXEC mode, follow these steps to upload an image to an FTP server:

	Command	Purpose
Step 1		Verify that the FTP server is properly configured by referring to the "Preparing to Download or Upload a Configuration File By Using FTP" section on page B-12.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**configure terminal**	Enter global configuration mode. This step is required only if you override the default remote username or password (see Steps 4, 5, and 6).
Step 4	**ip ftp username** *username*	(Optional) Change the default remote username.
Step 5	**ip ftp password** *password*	(Optional) Change the default password.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**archive upload-sw ftp:**[[**//**[*username*[**:***password*]**@**]*location*]**/***directory*]**/** *image-name***.tar**	Upload the currently running switch image to the FTP server. • For **//***username***:***password*, specify the username and password. These must be associated with an account on the FTP server. For more information, see the "Preparing to Download or Upload an Image File By Using FTP" section on page B-24. • For **@***location*, specify the IP address of the FTP server. • For **/***directory***/***image-name***.tar**, specify the directory (optional) and the name of the software image to be uploaded. Directory and image names are case sensitive. The *image-name***.tar** is the name of the software image to be stored on the server.

The **archive upload-sw** command builds an image file on the server by uploading these files in order: info, the Cisco IOS image, the HTML files, and info.ver. After these files are uploaded, the upload algorithm creates the tar file format.

Caution For the download and upload algorithms to operate properly, do *not* rename image names.

Copying Image Files By Using RCP

You can download a switch image from an RCP server or upload the image from the switch to an RCP server.

You download a switch image file from a server to upgrade the switch software. You can overwrite the current image with the new one or keep the current image after a download.

You upload a switch image file to a server for backup purposes. You can use this uploaded image for future downloads to the same switch or another of the same type.

This section includes this information:

- Preparing to Download or Upload an Image File By Using RCP, page B-28
- Downloading an Image File By Using RCP, page B-29
- Uploading an Image File By Using RCP, page B-31

Preparing to Download or Upload an Image File By Using RCP

RCP provides another method of downloading and uploading image files between remote hosts and the switch. Unlike TFTP, which uses User Datagram Protocol (UDP), a connectionless protocol, RCP uses TCP, which is connection-oriented.

To use RCP to copy files, the server from or to which you will be copying files must support RCP. The RCP copy commands rely on the rsh server (or daemon) on the remote system. To copy files by using RCP, you do not need to create a server for file distribution as you do with TFTP. You only need to have access to a server that supports the remote shell (rsh). (Most UNIX systems support rsh.) Because you are copying a file from one place to another, you must have read permission on the source file and write permission on the destination file. If the destination file does not exist, RCP creates it for you.

RCP requires a client to send a remote username on each RCP request to a server. When you copy an image from the switch to a server by using RCP, the software sends the first valid username in this list:

- The username specified in the **archive download-sw** or **archive upload-sw** privileged EXEC command if a username is specified.
- The username set by the **ip rcmd remote-username** *username* global configuration command if the command is entered.
- The remote username associated with the current TTY (terminal) process. For example, if the user is connected to the router through Telnet and was authenticated through the **username** command, the switch software sends the Telnet username as the remote username.
- The switch host name.

For the RCP copy request to execute successfully, an account must be defined on the network server for the remote username. If the server has a directory structure, the image file is written to or copied from the directory associated with the remote username on the server. For example, if the image file resides in the home directory of a user on the server, specify that user's name as the remote username.

Before you begin downloading or uploading an image file by using RCP, do these tasks:

- Ensure that the workstation acting as the RCP server supports the remote shell (rsh).
- Ensure that the switch has a route to the RCP server. The switch and the server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the RCP server by using the **ping** command.
- If you are accessing the switch through the console or a Telnet session and you do not have a valid username, make sure that the current RCP username is the one that you want to use for the RCP download. You can enter the **show users** privileged EXEC command to view the valid username. If you do not want to use this username, create a new RCP username by using the **ip rcmd remote-username** *username* global configuration command to be used during all archive operations. The new username is stored in NVRAM. If you are accessing the switch through a Telnet session and you have a valid username, this username is used, and there is no need to set the RCP username. Include the username in the **archive download-sw** or **archive upload-sw** privileged EXEC command if you want to specify a username only for that operation.

- When you upload an image to the RCP to the server, it must be properly configured to accept the RCP write request from the user on the switch. For UNIX systems, you must add an entry to the .rhosts file for the remote user on the RCP server. For example, suppose the switch contains these configuration lines:

  ```
  hostname Switch1
  ip rcmd remote-username User0
  ```

  If the switch IP address translates to *Switch1.company.com*, the .rhosts file for User0 on the RCP server should contain this line:

  ```
  Switch1.company.com Switch1
  ```

For more information, see the documentation for your RCP server.

Downloading an Image File By Using RCP

You can download a new image file and replace or keep the current image.

Beginning in privileged EXEC mode, follow Steps 1 through 6 to download a new image from an RCP server and overwrite the existing image. To keep the current image, omit Step 6.

	Command	Purpose
Step 1		Verify that the RCP server is properly configured by referring to the "Preparing to Download or Upload an Image File By Using RCP" section on page B-28.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**configure terminal**	Enter global configuration mode. This step is required only if you override the default remote username (see Steps 4 and 5).
Step 4	**ip rcmd remote-username** *username*	(Optional) Specify the remote username.
Step 5	**end**	Return to privileged EXEC mode.

	Command	Purpose
Step 6	**archive download-sw /overwrite /reload rcp:**[[[**//**[*username*@]*location*]**/***directory*]**/***image-name***.tar**]	Download the image file from the RCP server to the switch, and overwrite the current image. • The **/overwrite** option overwrites the software image in flash with the downloaded image. • The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. • For **//***username, s*pecify the username. For the RCP copy request to execute successfully, an account must be defined on the network server for the remote username. For more information, see the "Preparing to Download or Upload an Image File By Using RCP" section on page B-28. • For @*location*, specify the IP address of the RCP server. • For **/***directory***/***image-name***.tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.
Step 7	**archive download-sw /leave-old-sw /reload rcp:**[[[**//**[*username*@]*location*]**/***directory*]**/***image-name***.tar**]	Download the image file from the RCP server to the switch, and keep the current image. • The **/leave-old-sw** option keeps the old software version after a download. • The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. • For **//***username, s*pecify the username. For the RCP copy request to execute, an account must be defined on the network server for the remote username. For more information, see the "Preparing to Download or Upload an Image File By Using RCP" section on page B-28. • For @*location*, specify the IP address of the RCP server. • **For /***directory***]/***image-name***.tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.

The download algorithm verifies that the image is appropriate for the switch model and that enough DRAM is present, or it stops the process and reports an error. If you specify the **/overwrite** option, the download algorithm removes the existing image on the flash device whether or not it is the same as the new one, downloads the new image, and then reloads the software.

Note If the flash device has sufficient space to hold two images and you want to overwrite one of these images with the same version, you must specify the **/overwrite** option.

If you specify the **/leave-old-sw**, the existing files are not removed. If there is not enough room to install the new image an keep the running image, the download process stops, and an error message appears.

The algorithm installs the downloaded image onto the system board flash device (flash:). The image is placed in a new directory named with the software version string, and the BOOT environment variable is updated to point to the newly installed image.

If you kept the old software during the download process (you specified the **/leave-old-sw** keyword), you can remove it by entering the **delete /force /recursive** *filesystem***:/***file-url* privileged EXEC command. For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the directory name of the old software image. All the files in the directory and the directory are removed.

Caution For the download and upload algorithms to operate properly, do *not* rename image names.

Uploading an Image File By Using RCP

You can upload an image from the switch to an RCP server. You can later download this image to the same switch or to another switch of the same type.

The upload feature should be used only if the HTML pages associated with the device manager have been installed with the existing image.

Beginning in privileged EXEC mode, follow these steps to upload an image to an RCP server:

	Command	Purpose
Step 1		Verify that the RCP server is properly configured by referring to the "Preparing to Download or Upload an Image File By Using RCP" section on page B-28.
Step 2		Log into the switch through the console port or a Telnet session.
Step 3	**configure terminal**	Enter global configuration mode. This step is required only if you override the default remote username (see Steps 4 and 5).
Step 4	**ip rcmd remote-username** *username*	(Optional) Specify the remote username.

	Command	Purpose
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**archive upload-sw rcp:**[[[**//**[*username*@]*location*]**/***directory*]**/***image-name***.tar**]	Upload the currently running switch image to the RCP server. • For **//***username,* specify the username; for the RCP copy request to execute, an account must be defined on the network server for the remote username. For more information, see the "Preparing to Download or Upload an Image File By Using RCP" section on page B-28. • For @*location*, specify the IP address of the RCP server. • For **/***directory*]**/***image-name***.tar**, specify the directory (optional) and the name of the software image to be uploaded. Directory and image names are case sensitive. • The *image-name***.tar** is the name of software image to be stored on the server.

The **archive upload-sw** privileged EXEC command builds an image file on the server by uploading these files in order: info, the Cisco IOS image, the HTML files, and info.ver. After these files are uploaded, the upload algorithm creates the tar file format.

Caution For the download and upload algorithms to operate properly, do *not* rename image names.

APPENDIX **C**

Unsupported CLI Commands in Cisco IOS Release 12.2(25)SEE

This appendix lists some of the command-line interface (CLI) commands that appear when you enter the question mark (?) at the Catalyst 3550 switch prompt but are not supported in this release, either because they are not tested, or because of Catalyst 3550 hardware limitations. This is not a complete list. The unsupported commands are listed by software feature and command mode.

Access Control Lists

Unsupported Privileged EXEC Commands

access-enable [**host**] [**timeout** *minutes*]

access-template [*access-list-number* | *name*] [*dynamic-name*] [*source*] [*destination*] [**timeout** *minutes*]

clear access-template [*access-list-number* | *name*] [*dynamic-name*] [*source*] [*destination*].

ARP Commands

Unsupported Global Configuration Commands

arp *ip-address hardware-address* **smds**

arp *ip-address hardware-address* **srp-a**

arp *ip-address hardware-address* **srp-b**

Unsupported Interface Configuration Commands

arp probe

ip probe proxy

Boot Loader Commands

Unsupported Global Configuration Commands

boot buffersize

FallBack Bridging

Unsupported Privileged EXEC Commands

clear bridge [*bridge-group*] **multicast** [**router-ports** | **groups** | **counts**] [*group-address*] [*interface-unit*] [**counts**]

clear vlan statistics

show bridge [*bridge-group*] **circuit-group** [*circuit-group*] [*src-mac-address*] [*dst-mac-address*]

show bridge [*bridge-group*] **multicast** [**router-ports** | **groups**] [*group-address*]

show bridge vlan

show interfaces crb

show interfaces {**ethernet** | **fastethernet**} [*interface* | *slot/port*] **irb**

show subscriber-policy *range*

Unsupported Global Configuration Commands

bridge *bridge-group* **acquire**

bridge *bridge-group* **bitswap_l3_addresses**

bridge *bridge-group* **bridge ip**

bridge *bridge-group* **circuit-group** *circuit-group* **pause** *milliseconds*

bridge *bridge-group* **circuit-group** *circuit-group* **source-based**

bridge cmf

bridge crb

bridge *bridge-group* **domain** *domain-name*

bridge irb

bridge *bridge-group* **mac-address-table limit** *number*

bridge *bridge-group* **multicast-source**

bridge *bridge-group* **route** *protocol*

bridge *bridge-group* **subscriber policy** *policy*

subscriber-policy *policy* [[**no** | **default**] *packet* [**permit** | **deny**]]

Unsupported Interface Configuration Commands

bridge-group *bridge-group* **cbus-bridging**

bridge-group *bridge-group* **circuit-group** *circuit-number*

bridge-group *bridge-group* **input-address-list** *access-list-number*

bridge-group *bridge-group* **input-lat-service-deny** *group-list*

bridge-group *bridge-group* **input-lat-service-permit** *group-list*

bridge-group *bridge-group* **input-lsap-list** *access-list-number*

bridge-group *bridge-group* **input-pattern-list** *access-list-number*

bridge-group *bridge-group* **input-type-list** *access-list-number*

bridge-group *bridge-group* **lat-compression**

bridge-group *bridge-group* **output-address-list** *access-list-number*

bridge-group *bridge-group* **output-lat-service-deny** *group-list*

bridge-group *bridge-group* **output-lat-service-permit** *group-list*

bridge-group *bridge-group* **output-lsap-list** *access-list-number*

bridge-group *bridge-group* **output-pattern-list** *access-list-number*

bridge-group *bridge-group* **output-type-list** *access-list-number*

bridge-group *bridge-group* **sse**

bridge-group *bridge-group* **subscriber-loop-control**

bridge-group *bridge-group* **subscriber-trunk**

bridge *bridge-group* **lat-service-filtering**

frame-relay map bridge *dlci* **broadcast**

interface bvi *bridge-group*

x25 map bridge *x.121-address* **broadcast** [*options-keywords*]

HSRP

Unsupported Global Configuration Commands

interface Async

interface BVI

interface Dialer

interface Group-Async

interface Lex

interface Multilink

interface Virtual-Template

interface Virtual-Tokenring

Unsupported Interface Configuration Commands

mtu

standby mac-refresh *seconds*

standby use-bia

Interface Configuration Commands

interface tunnel

switchport broadcast *level*

switchport multicast *level*

switchport unicast *level*

Note These commands were replaced in Cisco IOS release 12.1(8)EA1 by the **storm-control** {**broadcast** | **multicast** | **unicast**} **level** *level* [*.level*] interface configuration command.

transmit-interface *type number*

IP Multicast Routing

Unsupported Privileged EXEC Commands

clear ip rtp header-compression [*type number*]

The **debug ip packet** command displays packets received by the switch CPU. It does not display packets that are hardware-switched.

The **debug ip mcache** command affects packets received by the switch CPU. It does not display packets that are hardware-switched.

The **debug ip mpacket** [**detail**] [*access-list-number* [*group-name-or-address*] command affects only packets received by the switch CPU. Because most multicast packets are hardware-switched, use this command only when you know that the route will forward the packet to the CPU.

debug ip pim atm

show frame-relay ip rtp header-compression [**interface** *type number*]

The **show ip mcache** command displays entries in the cache for those packets that are sent to the switch CPU. Because most multicast packets are switched in hardware without CPU involvement, you can use this command, but multicast packet information is not displayed.

The **show ip mpacket** commands are supported but are only useful for packets received at the switch CPU. If the route is hardward-switched, the command has no effect because the CPU does not receive the packet and cannot display it.

show ip pim vc [*group-address* | *name*] [*type number*]

show ip rtp header-compression [*type number*] [**detail**]

Unsupported Global Configuration Commands

ip multicast-routing vrf *vrf-name*

ip pim accept-rp {*address* | **auto-rp**} [*group-access-list-number*]

ip pim message-interval *seconds*

ip pim register-rate-limit

Unsupported Interface Configuration Commands

frame-relay ip rtp header-compression [**active** | **passive**]

frame-relay map ip *ip-address dlci* [**broadcast**] **compress**

frame-relay map ip *ip-address dlci* **rtp header-compression** [**active** | **passive**]

ip igmp helper-address *ip-address*

ip multicast helper-map {*group-address* | **broadcast**} {*broadcast-address* | *multicast-address*} *extended-access-list-number*

ip multicast rate-limit {**in** | **out**} [**video** | **whiteboard**] [**group-list** *access-list*] [**source-list** *access-list*] *kbps*

ip multicast use-functional

ip pim minimum-vc-rate *pps*

ip pim multipoint-signalling

ip pim nbma-mode

ip pim vc-count *number*

ip rtp compression-connections *number*

ip rtp header-compression [**passive**]

IP Unicast Routing

Unsupported Privileged EXEC or User EXEC Commands

clear ip accounting [**checkpoint**]

clear ip bgp *address* **flap-statistics**

clear ip bgp prefix-list

set ip default next-hop

show cef [**drop** | **not-cef-switched**]

show ip accounting [**checkpoint**] [**output-packets** | **access-violations**]

show ip bgp dampened-paths

show ip bgp flap-statistics

show ip bgp inconsistent-as

show ip bgp regexp *regular expression*

Unsupported Global Configuration Commands

ip accounting-list ***ip-address wildcard***

ip accounting-transits *count*

ip cef accounting [**per-prefix**] [**non-recursive**]

ip cef traffic-statistics [**load-interval** *seconds*] [**update-rate** *seconds*]]

ip flow-aggregation

ip flow-cache

ip flow-export

ip gratituitous-arps

ip local

ip reflexive-list

router egp

router-isis

router iso-igrp

router mobile

router odr

router static

Unsupported Interface Configuration Commands

ip accounting precedence {**input** | **output**}

ip load-sharing [**per-packet**]

ip mtu *bytes*

ip verify

ip unnumbered *type number*

All **ip security** commands

Unsupported BGP Router Configuration Commands

address-family vpnv4

default-information originate

neighbor advertise-map

neighbor allowas-in

neighbor default-originate

neighbor description

network backdoor

table-map

Unsupported VPN Configuration Commands

All

Note The switch does support multi-VPN routing/forwarding (multi-VRF) commands shown in the command reference for this release.

Unsupported Route Map Commands

match length

route-map *map-tag* **deny**

set automatic-tag

set dampening *half-life reuse suppress max-suppress-time*

set default interface

set interface

set ip default next-hop

set ip default next-hop verify-availability

set ip destination *ip-address mask*

set ip df

set ip precedence *value*

set ip *qos-group*

set tag *tag-value*

set ip tos

Miscellaneous

Privileged EXEC Commands

file verify auto

MSDP

Unsupported Privileged EXEC Commands

show access-expression

show exception

show location

show pm LINE

show smf [*interface-id*]

show subscriber-policy [*policy-number*]

show template [*template-name*]

Unsupported Global Configuration Commands

ip msdp default-peer *ip-address* | *name* [**prefix-list** *list*] (Because BGP/MBGP is not supported, use the **ip msdp peer** command instead of this command.)

NetFlow Commands

Unsupported Global Configuration Commands

ip flow-aggregation cache

ip flow-cache entries

ip flow-export

Network Address Translation (NAT) commands

Unsupported User EXEC Commands

clear ip nat translation

show ip nat statistics

show ip nat translations

Unsupported Global Configuration Commands

ip nat inside destination

ip nat inside source

ip nat outside source

ip nat pool

Unsupported Interface Configuration Commands

ip nat

QoS

Unsupported Global Configuration Commands

policy-map *policy-map-name* **bandwidth**

policy-map *policy-map-name* **set mpls**

priority-list

Unsupported Interface Configuration Command

priority-group

Unsupported Policy-Map Configuration Command

class class-default where **class-default** is the *class-map-name*.

Unsupported Class-Map Configuration Commands

match any

match destination-address

match input-interface

match mpls

match not

match protocol

match source-address

RADIUS

Unsupported Global Configuration Commands

aaa nas port extended

aaa authentication *feature* **default enable**

aaa authentication *feature* **default line**

radius-server attribute nas-port

radius-server configure

radius-server extended-portnames

SNMP

Unsupported Global Configuration Commands

snmp-server enable informs

snmp-server enable traps flash insertion

snmp-server enable traps flash removal

snmp-server ifindex persist

Spanning Tree

Unsupported Global Configuration Command

spanning-tree etherchannel guard misconfig

VLAN

Unsupported User EXEC Commands

ifindex

private-vlan

show interfaces *vlan-id* **counters**

INDEX

Numerics

144-bit Layer 3 TCAM 6-26, 31-65

A

AAA down policy, NAC Layer 2 IP validation 1-6
abbreviating commands 2-4
ABRs 31-24
AC (command switch) 5-10
access-class command 29-22
access control entries
  See ACEs
access-denied response, VMPS 11-27
access groups
  IP 29-22
  Layer 3 29-22
accessing
  clusters, switch 5-13
  command switches 5-11
  member switches 5-13
  switch clusters 5-13
access lists
  See ACLs
access ports
  and Layer 2 protocol tunneling 14-10
  defined 9-2
  in switch clusters 5-9
accounting
  with IEEE 802.1x 8-8, 8-31
  with RADIUS 7-28
  with TACACS+ 7-11, 7-17
ACEs
  and QoS 28-7
  defined 29-2
  Ethernet 29-2
  IP 29-2
ACLs
  ACEs 29-2
  and logging 29-7
  any keyword 29-14
  applying
    on bridged packets 29-38
    on multicast packets 29-40
    on routed packets 29-39
    on switched packets 29-38
    time ranges to 29-18
    to Layer 2 and Layer 3 interfaces 29-22
    to QoS 28-7
  classifying traffic for QoS 28-36
  comments in 29-20
  compatibility on the same switch 29-2
  compiling 29-23
  configuration conflict examples 29-44
  configuring with VLAN maps 29-37
  defined 29-1
  examples, not fitting in hardware 29-45
  examples of 28-36, 29-23
  extended IP
    configuring for QoS classification 28-37
    creating 29-11
    matching criteria 29-8
  feature manager 29-42
  hardware and software handling 29-6
  hardware support for 29-6

ACLs (continued)

host keyword 29-14
input router ACL configuration guidelines 29-7
IP
applying to interface 29-21
creating 29-8
defined 29-8
fragments and QoS guidelines 28-28
implicit deny 29-10, 29-16, 29-18
implicit masks 29-10
matching criteria 29-8
matching criteria for port ACLs 29-4
matching criteria for router ACLs 29-3
named 29-16
options and QoS guidelines 28-28
undefined 29-23
violations, logging 29-18
virtual terminal lines, setting on 29-21
limiting actions 29-37
logging messages 29-11
log keyword 29-18
MAC extended 28-39, 29-27
matching 29-8, 29-22, 29-30
merge failure examples 29-46
monitoring 29-41
named 29-16
not fitting in hardware 29-44
number per QoS class map 28-28
numbers 29-9
policy maps and QoS classification 28-28
port
and voice VLAN 29-4
defined 29-2
limitations 29-4
preventing excessive TCAM usage 29-7
QoS 28-7, 28-36
resequencing entries 29-16
router 29-2

ACLs (continued)

standard IP
configuring for QoS classification 28-36
creating 29-10
matching criteria 29-8
support for 1-5
time ranges 29-18
undefined 29-30
unsupported features 29-8
using router ACLs with VLAN maps 29-36
VLAN maps
configuration guidelines 29-31
configuring 29-30
defined 29-4
active links 18-2
active router 32-1
active traffic monitoring, IP SLAs 1-1
addresses
displaying the MAC address table 6-26
dynamic
accelerated aging 15-8
changing the aging time 6-21
default aging 15-8
defined 6-19
learning 6-20
preventing frame forwarding 5-5
removing 6-21
filtering frames by MAC address 5-6
MAC, discovering 6-29
multicast
group address range 3-1, 3-3
STP address management 15-8
static
adding and removing 6-23
defined 6-19
address resolution 6-29, 31-8
Address Resolution Protocol
See ARP
See ARP table

adjacency tables, with CEF 31-72
administrative distances
defined 31-82
OSPF 31-30
routing protocol defaults 31-74
advertisements
CDP 22-1
RIP 31-19
VTP 11-18, 12-3
aggregate addresses, BGP 31-57
aggregated ports
See EtherChannel
aggregate policers 28-50
aggregate policing 1-7
aging, accelerating 15-8
aging time
accelerated
for MSTP 16-23
for STP 15-8, 15-21
bridge table for fallback bridging 5-5
MAC address table 6-21
maximum
for MSTP 16-23, 16-24
for STP 15-22, 15-23
alarms, RMON 25-3
allowed-VLAN list 11-21
area border routers
See ABRs
ARP
configuring 31-9
defined 31-8
encapsulation 31-10
static cache configuration 31-9
support for 1-3
ARP table
address resolution 6-29
managing 6-29
ASBRs 31-24
AS-path filters, BGP 31-51
asymmetrical links, and IEEE 802.1Q tunneling 14-4
attributes, RADIUS
vendor-proprietary 7-31
vendor-specific 7-29
authentication
EIGRP 31-39
HSRP 32-8
local mode with AAA 7-36
NTP associations 6-4
RADIUS
defined 7-18
key 7-21
login 7-23
TACACS+
defined 7-11
key 7-13
login 7-14
See also port-based authentication
authentication failed VLAN
See restricted VLAN
authentication keys, and routing protocols 31-83
authoritative time source, described 6-2
authorization
with RADIUS 7-27
with TACACS+ 7-11, 7-16
authorized ports with IEEE 802.1x 8-7
autoconfiguration 3-3
automatic discovery
considerations
beyond a noncandidate device 5-7
brand new switches 5-9
connectivity 5-4
different VLANs 5-6
management VLANs 5-7
non-CDP-capable devices 5-6
noncluster-capable devices 5-6
routed ports 5-8
in switch clusters 5-4
See also CDP

automatic QoS
See QoS
automatic recovery, clusters 5-10
See also HSRP
autonegotiation
duplex mode 1-2
interface configuration guidelines 9-15
mismatches 6-10
autonomous system boundary routers
See ASBRs
autonomous systems, in BGP 31-45
Auto-RP, described 3-5
autosensing, port speed 1-2
auxiliary VLAN
See voice VLAN

B

BackboneFast
described 17-9
enabling 17-19
support for 1-4
backup interfaces
See Flex Links
backup links 18-2
bandwidth for QoS
allocating 28-64
described 28-13
banners
configuring
login 6-18
message-of-the-day login 6-18
default configuration 6-17
when displayed 6-17
BGP
aggregate addresses 31-57
aggregate routes, configuring 31-57
CIDR 31-57
clear commands 31-60

BGP (continued)
community filtering 31-54
configuring neighbors 31-55
default configuration 31-42
described 31-42
enabling 31-45
monitoring 31-60
multipath support 31-49
neighbors, types of 31-45
path selection 31-49
peers, configuring 31-55
prefix filtering 31-53
resetting sessions 31-48
route dampening 31-59
route maps 31-51
route reflectors 31-58
routing domain confederation 31-58
routing session with multi-VRF CE 31-67
show commands 31-60
supernets 31-57
support for 1-7
Version 4 31-42
binding cluster group and HSRP group 32-10
binding database
address, DHCP server
See DHCP, Cisco IOS server database
DHCP snooping
See DHCP snooping binding database
binding database, DHCP snooping
See DHCP snooping binding database
bindings
address, Cisco IOS DHCP server 18-7
DHCP snooping database 18-7
IP source guard 18-18
binding table, DHCP snooping
See DHCP snooping binding database
blocking packets 21-6

booting
- boot loader, function of 3-2
- boot process 3-1
- manually 3-12
- specific image 3-13

boot loader
- accessing 3-13
- described 3-2
- environment variables 3-13
- prompt 3-13
- trap-door mechanism 3-2

bootstrap router (BSR), described 3-5

Border Gateway Protocol
- See BGP

BPDU
- error-disabled state 17-2
- filtering 17-3
- RSTP format 16-12

BPDU filtering
- described 17-3
- enabling 17-16
- support for 1-4

BPDU guard
- described 17-2
- enabling 17-15
- support for 1-4

bridged packets, ACLs on 29-38

bridge groups
- See fallback bridging

bridge protocol data unit
- See BPDU

broadcast flooding 31-16

broadcast packets
- directed 31-13
- flooded 31-13

broadcast storm control
- See storm control

broadcast storm-control command 21-4

broadcast storms 31-13

C

cables, monitoring for unidirectional links 23-1

cache engines, redirecting traffic to 2-1

CAMs, ACLs not loading in 29-45

candidate switch
- automatic discovery 5-4
- defined 5-3
- requirements 5-3
- See also command switch, cluster standby group, and member switch

CA trustpoint
- configuring 7-44
- defined 7-42

CDP
- and trusted boundary 28-33
- automatic discovery in switch clusters 5-4
- configuring 22-2
- default configuration 22-2
- described 22-1
- disabling for routing device 22-3, 22-4
- enabling and disabling
  - on an interface 22-4
  - on a switch 22-3
- Layer 2 protocol tunneling 14-7
- monitoring 22-4
- overview 22-1
- power negotiation extensions 9-6
- support for 1-3
- transmission timer and holdtime, setting 22-2
- updates 22-2

CEF 31-72

CGMP
- as IGMP snooping learning method 20-8
- clearing cached group entries 3-51
- enabling se
- rver support 3-32
- joining multicast group 20-3
- overview 3-8

CGMP (continued)
server support only 3-8
switch support of 1-2
CIDR 31-57
CipherSuites 7-43
Cisco Discovery Protocol
See CDP
Cisco Express Forwarding
See CEF
Cisco Group Management Protocol
See CGMP
Cisco Intelligence Engine 2100 Series Configuration Registrar
See IE2100
Cisco intelligent power management 9-6
Cisco IOS DHCP server
See DHCP, Cisco IOS DHCP server
Cisco IOS File System
See IFS
Cisco IOS IP SLAs 1-1
Cisco Network Assistant
See Network Assistant
CiscoWorks 2000 1-9, 27-4
classless interdomain routing
See CIDR
classless routing 31-7
class maps for QoS
configuring per physical port 28-40
configuring per-port per-VLAN 28-41
described 28-7
displaying 28-69
class of service
See CoS
clearing interfaces 9-22
CLI
abbreviating commands 2-4
command modes 2-1
configuration logging 2-5
described 1-9

CLI (continued)
editing features
enabling and disabling 2-7
keystroke editing 2-7
wrapped lines 2-9
error messages 2-5
filtering command output 2-10
getting help 2-3
history
changing the buffer size 2-6
described 2-6
disabling 2-7
recalling commands 2-6
no and default forms of commands 2-4
client mode, VTP 12-3
clock
See system clock
clusters, switch
accessing 5-13
automatic discovery 5-4
automatic recovery 5-10
benefits 1-10
compatibility 5-4
described 5-1
managing
through SNMP 5-15
planning 5-4
planning considerations
automatic discovery 5-4
automatic recovery 5-10
host names 5-13
IP addresses 5-13
passwords 5-13
RADIUS 5-14
SNMP 5-14, 5-15
TACACS+ 5-14
See also candidate switch, command switch, cluster standby group, member switch, and standby command switch

cluster standby group
and HSRP group 32-10
automatic recovery 5-12
considerations 5-11
defined 5-2
requirements 5-3
virtual IP address 5-11
See also HSRP
CNS
Configuration Engine
configID, deviceID, hostname 4-3
configuration service 4-2
described 4-1
event service 4-3
embedded agents
described 4-5
enabling automated configuration 4-6
enabling configuration agent 4-9
enabling event agent 4-7
for upgrading 4-14
Coarse Wave Division Multiplexer GBIC modules
See CWDM GBIC modules
command-line interface
See CLI
command modes 2-1
commands
abbreviating 2-4
no and default 2-4
setting privilege levels 7-8
command switch
accessing 5-11
active (AC) 5-10
configuration conflicts 6-10
defined 5-2
passive (PC) 5-10
password privilege levels 5-14
priority 5-10
command switch (continued)
recovery
from command-switch failure 5-10
from failure 6-6
from lost member connectivity 6-10
redundant 5-10
replacing
with another switch 6-8
with cluster member 6-7
requirements 5-3
standby (SC) 5-10
See also candidate switch, cluster standby group, member switch, and standby command switch
community list, BGP 31-54
community strings
configuring 5-14, 27-8
for cluster switches 27-4
in clusters 5-14
overview 27-4
SNMP 5-14
config.text 3-11
configurable leave timer, IGMP 20-5
configuration conflicts
ACL, displaying 29-43
recovering from lost member connectivity 6-10
configuration examples, network 1-10
configuration files
clearing the startup configuration B-18
creating using a text editor B-9
default name 3-11
deleting a stored configuration B-18
described B-7
downloading
automatically 3-11
preparing B-10, B-12, B-15
reasons for B-8
using FTP B-13
using RCP B-16
using TFTP B-10

configuration files (continued)
guidelines for creating and using B-8
invalid combinations when copying B-5
limiting TFTP server access 27-16
obtaining with DHCP 3-7
password recovery disable considerations 7-5
specifying the filename 3-11
system contact and location information 27-15
types and location B-9
uploading
preparing B-10, B-12, B-15
reasons for B-8
using FTP B-14
using RCP B-17
using TFTP B-11
VMPS database 11-28
configuration guidelines, multi-VRF CE 31-64
configuration logging 2-5
configuration settings, saving 3-10
configure terminal command 9-9
Configuring a Restricted VLAN 8-33
configuring PoE 9-16
config-vlan mode 2-2, 11-6
conflicts, configuration 6-10
congestion-avoidance techniques 28-12
congestion-management techniques 28-12, 28-15
connections, secure remote 7-38
connectivity problems 6-11
consistency checks in VTP version 2 12-4
console port, connecting to 2-10
content-routing technology
See WCCP
control protocol, IP SLAs 1-3
CoS
in Layer 2 frames 28-2
override priority 13-5
trust priority 13-5
CoS-to-DSCP map for QoS 28-52
CoS-to-egress-queue map 28-59
counters, clearing interface 9-22
CPU q, in show forward command output 6-20
crashinfo file 6-20
critical authentication, IEEE 802.1x 8-34
cross-stack UplinkFast, STP
connecting stack ports 17-8
described 17-5
enabling 17-18
fast-convergence events 17-7
Fast Uplink Transition Protocol 17-6
limitations 17-8
normal-convergence events 17-7
Stack Membership Discovery Protocol 17-6
support for 1-4
cryptographic software image
Kerberos 7-32
SSL 7-41
customer edge devices 31-62
CWDM GBIC modules, network example 1-18
CWDM OADM modules 1-18

D

daylight saving time 6-13
debugging
enabling all system diagnostics 6-17
enabling for a specific feature 6-17
redirecting error message output 6-18
using commands 6-16
default commands 2-4
default configuration
auto-QoS 28-17
banners 6-17
BGP 31-42
booting 3-11
CDP 22-2
DHCP 18-9
DHCP option 82 18-9
DHCP snooping 18-9

default configuration (continued)
DHCP snooping binding database 18-9
DNS 6-16
dynamic ARP inspection 19-5
EIGRP 31-35
EtherChannel 30-8
fallback bridging 5-3
Flex Links 18-5
HSRP 32-4
IEEE 802.1Q tunneling 14-4
IEEE 802.1x 8-20
IGMP 3-27
IGMP filtering 20-22
IGMP snooping 20-7
IGMP throttling 20-23
initial switch information 3-3
IP addressing, IP routing 31-4
IP multicast routing 3-9
IP SLAs 1-6
IP source guard 18-19
Layer 2 interfaces 9-14
Layer 2 protocol tunneling 14-10
MAC address table 6-20
MAC address-table move update 18-5
MSDP 4-4
MSTP 16-14
multi-VRF CE 31-64
MVR 20-18
NTP 6-4
optional spanning-tree features 17-14
OSPF 31-25
password and privilege level 7-2
port security 21-9
RADIUS 7-20
RIP 31-19
RMON 25-3
RSPAN 24-8
SNMP 27-6
SPAN 24-8

default configuration (continued)
SSL 7-44
standard QoS 28-26
storm control 21-3
STP 15-11
system message logging 26-3
system name and prompt 6-15
TACACS+ 7-13
UDLD 23-4
VLAN, Layer 2 Ethernet interfaces 11-18
VLANs 11-7
VMPS 11-28
voice VLAN 13-2
VTP 12-6
WCCP 2-4
default gateway 3-10, 31-11
default networks 31-75
default routes 31-75
default routing 31-2
deleting VLANs 11-10
denial-of-service attack 21-1
description command 9-19
designing your network, examples 1-10
destination addresses, in ACLs 29-13
detecting indirect link failures, STP 17-10
device B-18
device discovery protocol 22-1
device manager
described 1-1, 1-9
upgrading a switch B-18
DHCP
Cisco IOS server database
configuring 18-17
default configuration 18-9
described 18-7

DHCP-based autoconfiguration
client request message exchange 3-4
configuring
client side 3-3
DNS 3-6
relay device 3-6
server-side 3-5, 18-11
TFTP server 3-6
example 3-8
lease options
for IP address information 3-5
for receiving the configuration file 3-5
overview 3-3
relationship to BOOTP 3-4
relay support 1-8
support for 1-3
DHCP binding database
See DHCP snooping binding database
DHCP binding table
See DHCP snooping binding database
DHCP option 82
circuit ID suboption 18-5
configuration guidelines 18-10
default configuration 18-9
displaying 18-18
enabling
relay agent 18-11
relay agent information option 18-11
forwarding address, specifying 18-13
helper address 18-13
overview 18-3
packet format
circuit ID suboption 18-5
remote ID suboption 18-5
policy for reforwarding 18-12
reforwarding policy 18-12
remote ID suboption 18-5
support for 1-3
validating 18-12
DHCP relay agent 18-11
DHCP server 18-11
DHCP snooping
accepting untrusted packets form edge switch 18-3, 18-15
and private VLANs 18-16
binding database
See DHCP snooping binding database
configuration guidelines 18-10
default configuration 18-9
displaying binding tables 18-18
displaying configuration 18-18
message exchange process 18-4
option 82 data insertion 18-3
trusted interface 18-2
untrusted interface 18-2
untrusted messages 18-2
DHCP snooping binding database
adding bindings 18-17
binding file
format 18-8
bindings 18-7
clearing agent statistics 18-18
configuring 18-17
default configuration 18-9
deleting
binding file 18-18
bindings 18-18
database agent 18-18
described 18-2, 18-7
displaying 18-18
status and statistics 18-18
enabling 18-17
entries 18-2
entry 18-7
renewing database 18-18
resetting
delay value 18-18
timeout value 18-18

DHCP snooping binding table
See DHCP snooping binding database
Differentiated Services architecture, QoS 28-2
Differentiated Services Code Point 28-2
Diffusing Update Algorithm (DUAL) 31-34
directed unicast requests 1-3
directories
changing B-3
creating and removing B-4
displaying the working B-3
discovery, clusters
See automatic discovery
Distance Vector Multicast Routing Protocol
See DVMRP
distance-vector protocols 31-2
distribute-list command 31-82
DNS
and DHCP-based autoconfiguration 3-6
default configuration 6-16
displaying the configuration 6-17
overview 6-15
setting up 6-16
support for 1-3
domain names
DNS 6-15
VTP 12-8
Domain Name System
See DNS
dot1q-tunnel switchport mode 11-17
double-tagged packets
IEEE 802.1Q tunneling 14-2
Layer 2 protocol tunneling 14-10
downloading
configuration files
preparing B-10, B-12, B-15
reasons for B-8
using FTP B-13
using RCP B-16
using TFTP B-10
downloading (continued)
image files
deleting old image B-22
preparing B-21, B-24, B-28
reasons for B-18
using CMS 1-2
using FTP B-25
using HTTP 1-2, B-18
using Network Assistant 1-2
using RCP B-29
using TFTP B-21
using the device manager or Network Assistant B-18
drop threshold for Layer 2 protocol packets 14-10
DSCP 1-6, 1-7, 28-2
DSCP-to-CoS map for QoS 28-55
DSCP-to-DSCP-mutation map for QoS 28-57
DSCP-to-threshold map for QoS 28-61
DTP 1-4, 11-16
DUAL finite state machine, EIGRP 31-35
duplex mode, configuring 9-15
DVMRP
autosummarization
configuring a summary address 3-48
disabling 3-50
connecting PIM domain to DVMRP router 3-40
enabling unicast routing 3-43
interoperability
with Cisco devices 3-38
with IOS software 3-7
mrinfo requests, responding to 3-43
neighbors
advertising the default route to 3-42
discovery with Probe messages 3-38
displaying information 3-43
prevent peering with nonpruning 3-46
rejecting nonpruning 3-44
overview 3-7

DVMRP (continued)

routes

adding a metric offset 3-50

advertising all 3-50

advertising the default route to neighbors 3-42

caching DVMRP routes learned in report messages 3-44

changing the threshold for syslog messages 3-47

deleting 3-51

displaying 3-52

favoring one over another 3-50

limiting the number injected into MBONE 3-47

limiting unicast route advertisements 3-38

routing table 3-8

source distribution tree, building 3-8

support for 1-8

tunnels

configuring 3-40

displaying neighbor information 3-43

dynamic access ports

characteristics 11-3

configuring 11-30

defined 9-3

dynamic addresses

See addresses

dynamic ARP inspection

ARP cache poisoning 19-1

ARP requests, described 19-1

ARP spoofing attack 19-1

clearing

log buffer 19-15

statistics 19-14

configuration guidelines 19-6

configuring

ACLs for non-DHCP environments 19-8

in DHCP environments 19-7

log buffer 19-12

rate limit for incoming ARP packets 19-4, 19-10

default configuration 19-5

dynamic ARP inspection (continued)

denial-of-service attacks, preventing 19-10

described 19-1

DHCP snooping binding database 19-2

displaying

ARP ACLs 19-14

configuration and operating state 19-14

log buffer 19-15

statistics 19-14

trust state and rate limit 19-14

error-disabled state for exceeding rate limit 19-4

function of 19-2

interface trust states 19-3

log buffer

clearing 19-15

configuring 19-12

displaying 19-15

logging of dropped packets, described 19-4

man-in-the middle attack, described 19-2

network security issues and interface trust states 19-3

priority of ARP ACLs and DHCP snooping entries 19-4

rate limiting of ARP packets

configuring 19-10

described 19-4

error-disabled state 19-4

statistics

clearing 19-14

displaying 19-14

validation checks, performing 19-11

dynamic desirable trunking mode 11-17

Dynamic Host Configuration Protocol

See DHCP-based autoconfiguration

dynamic port VLAN membership

described 11-27

reconfirming 11-31

troubleshooting 11-32

types of connections 11-30

VMPS database configuration file 11-28

dynamic routing 31-2
Dynamic Trunking Protocol
See DTP

E

EBGP 31-41
editing features
enabling and disabling 2-7
keystrokes used 2-7
wrapped lines 2-9
egress q, in show forward command output 6-20
EIGRP
authentication 31-39
components 31-34
configuring 31-37
default configuration 31-35
definition 31-34
interface parameters, configuring 31-38
monitoring 31-40
support for 1-7
enable password 7-3
enable secret password 7-3
encryption, CipherSuite 7-43
encryption for passwords 7-3
Enhanced IGRP
See EIGRP
environment variables
function of 3-15
location in Flash 3-13
equal-cost routing 1-8, 31-73
error messages
during command entry 2-5
setting the display destination device 26-5
severity levels 26-8
system message format 26-2
EtherChannel
automatic creation of 30-3
channel groups
binding physical and logical interfaces 30-3
numbering of 30-3
configuration guidelines 30-8
configuring
Layer 2 interfaces 30-9
Layer 3 physical interfaces 30-13
Layer 3 port-channel logical interfaces 30-12
default configuration 30-8
destination MAC address forwarding 30-6
displaying status 30-18
forwarding methods 30-15
interaction
with STP 30-8
with VLANs 30-9
LACP, support for 1-2
Layer 3 interface 31-3
load balancing 30-6, 30-15
logical interfaces, described 30-3
number of interfaces per 30-2
overview 30-1
PAgP
aggregate-port learners 30-5
compatibility with Catalyst 1900 30-15
displaying status 30-18
interaction with other features 30-5
learn method and priority configuration 30-15
modes 30-4
overview 30-3
silent mode 30-5
support for 1-2
port-channel interfaces
described 30-3
numbering of 30-3
port groups 9-5
source MAC address forwarding 30-6
support for 1-2

EtherChannel guard
described 17-11
enabling 17-20
Ethernet VLANs
adding 11-8
defaults and ranges 11-8
modifying 11-8
events, RMON 25-3
examples
network configuration 1-10
expedite queue for QoS
10/100 Ethernet ports
allocating bandwidth 28-68
configuring 28-67
described 28-15
Gigabit-capable Ethernet ports
allocating bandwidth 28-64
configuring 28-63
described 28-12
Express Setup
overview 1-1
See also getting started guide
extended-range VLANs
configuration guidelines 11-12
configuring 11-11
creating 11-12, 11-13
defined 11-1
extended system ID
MSTP 16-17
STP 15-4, 15-15
Extensible Authentication Protocol over LAN 8-1
external BGP
See EBGP
external neighbors, BGP 31-45

F

fallback bridging
and protected ports 5-4
bridge groups
creating 5-3
described 5-1
displaying 5-12
function of 5-2
number supported 5-3
removing 5-4
bridge table
changing the aging time 5-5
clearing 5-12
displaying 5-12
configuration guidelines 5-3
connecting interfaces with 9-9
default configuration 5-3
described 5-1
frame forwarding
filtering by MAC address 5-6
flooding packets 5-2
for static addresses 5-5
forwarding packets 5-2
preventing for dynamically learned stations 5-5
to static addresses 5-5
overview 5-1
protocol, unsupported 5-3
STP
disabling on an interface 5-12
forward-delay interval 5-10
hello BPDU interval 5-10
interface priority 5-8
maximum-idle interval 5-11
path cost 5-9
switch priority 5-8
VLAN-bridge STP 5-1, 5-2
support for 1-8
SVIs and routed ports 5-1

fallback bridging (continued)
unsupported protocols 5-3
VLAN-bridge STP 15-10
fallback VLAN name 11-28
Fast Uplink Transition Protocol 17-6
feature manager, ACL 29-42
FIB 31-72
fiber-optic, detecting unidirectional links 23-1
files
basic crashinfo
description 6-20
location 6-20
copying B-4
crashinfo
description 6-20
deleting B-5
displaying the contents of B-7
extended crashinfo
description 6-21
location 6-21
tar
creating B-5
displaying the contents of B-6
extracting B-7
image file format B-19
file system
displaying available file systems B-2
displaying file information B-3
local file system names B-1
network file system names B-4
setting the default B-3
filtering
in a VLAN 29-30
non-IP traffic 29-27
show and more command output 2-10
with fallback bridging 5-6
filters, IP
See ACLs, IP
flash device, number of B-1
Flex Links
configuration guidelines 18-4
configuring 18-5, 18-6
default configuration 18-5
description 18-1
monitoring 18-9
flooded traffic, blocking 21-6
flow-based packet classification 1-7
flowcharts
QoS classification 28-6
QoS policing and marking 28-10
QoS queueing and scheduling
10/100 ports 28-15
Gigabit-capable ports 28-12
flow control 1-2, 9-17
forward-delay time
MSTP 16-23
STP 15-5, 15-21
Forwarding Information Base
See FIB
forwarding non-routable protocols 5-1
FTP
accessing MIB files A-3
configuration files
downloading B-13
overview B-12
preparing the server B-12
uploading B-14
image files
deleting old image B-26
downloading B-25
preparing the server B-24
uploading B-26

G

GBIC modules
See GBICs
GBICs
1000BASE-LX/LH module 1-14
1000BASE-SX module 1-14
1000BASE-T module 1-14
1000BASE-ZX module 1-14
CWDM module 1-18
GigaStack module 1-12
security and identification 6-10
get-bulk-request operation 27-3
get-next-request operation 27-3, 27-4
get-request operation 27-3, 27-4
get-response operation 27-3
Gigabit Interface Converters
See GBICs
GigaStack GBIC
fast transition of redundant link 17-5
See also GBICs
global configuration mode 2-2
global leave, IGMP 20-12
guide mode 1-10
GUIs
See device manager and Network Assistant 1-9

H

hardware, determining ACL configuration fit 29-44
hello time
MSTP 16-22
STP 15-21
help, for the command line 2-3
history
changing the buffer size 2-6
described 2-6
disabling 2-7
recalling commands 2-6
history table, level and number of syslog messages 26-10
host names
in clusters 5-13
hosts, limit on dynamic ports 11-32
Hot Standby Router Protocol
See HSRP
HP OpenView 1-9
HSRP
authentication string 32-8
automatic cluster recovery 5-12
binding to cluster group 32-10
cluster standby group considerations 5-11
command-switch redundancy 1-3
default configuration 32-4
definition 32-1
monitoring 32-10
overview 32-1
priority 32-6
routing redundancy 1-7
timers 32-8
tracking 32-7
See also clusters, cluster standby group, and standby command switch
HTTP over SSL
see HTTPS
HTTPS 7-42
configuring 7-45
self-signed certificate 7-42
HTTP secure server 7-42

I

IBPG 31-41
ICMP
redirect messages 31-11
support for 1-8
time exceeded messages 6-13
traceroute and 6-13

ICMP (continued)
unreachable messages 29-6
unreachables and ACLs 29-7
ICMP Echo operation
configuring 1-11
IP SLAs 1-10
ICMP ping
executing 6-11
overview 6-11
ICMP Router Discovery Protocol
See IRDP
IDS, using with SPAN and RSPAN 24-2
IE2100
described 1-9
support for 1-2
IEEE 802.1D
See STP
IEEE 802.1p 13-1
IEEE 802.1Q
and trunk ports 9-3
configuration limitations 11-18
encapsulation 11-15
native VLAN for untagged traffic 11-22
tunneling
compatibility with other features 14-5
defaults 14-4
described 14-1
tunnel ports and ACLs 29-3
tunnel ports with other features 14-6
IEEE 802.1s
See MSTP
IEEE 802.1w
See RSTP
IEEE 802.1x
See port-based authentication
IEEE 802.3af
See PoE
IEEE 802.3x flow control 9-17
ifIndex values, SNMP 27-5
IFS 1-3
IGMP
configurable leave timer, procedures 20-11
configuring the switch
as a member of a group 3-27
statically connected member 3-31
controlling access to groups 3-28
default configuration 3-27
deleting cache entries 3-52
displaying groups 3-52
fast switching 3-31
flooded multicast traffic
controlling the length of time 20-12
disabling on an interface 20-13
global leave 20-12
query solicitation 20-12
recovering from flood mode 20-12
host-query interval, modifying 3-29
joining multicast group 20-3
join messages 20-3
leave processing, enabling 20-10
leaving multicast group 20-4
multicast reachability 3-27
overview 3-2
queries 20-3
report suppression
described 20-5
disabling 20-13
support for 1-2
throttling action 20-22
Version 1
changing to Version 2 3-29
described 3-3
Version 2
changing to Version 1 3-29
described 3-3
maximum query response time value 3-31
pruning groups 3-31
query timeout value 3-30

IGMP configurable leave timer, described 20-5
IGMP filtering
configuring 20-23
default configuration 20-22
described 20-22
monitoring 20-27
IGMP groups
configuring the throttling action 20-25
setting the maximum number 20-25
IGMP profile
applying 20-24
configuration mode 20-23
configuring 20-23
IGMP snooping
configuring 20-6
default configuration 20-7
definition 20-1
enabling and disabling 20-7
global configuration 20-7
Immediate Leave 20-5
method 20-8
monitoring 20-14
support for 1-2
VLAN configuration 20-7
IGMP throttling
configuring 20-25
default configuration 20-23
described 20-22
displaying action 20-27
IGP 31-24
Immediate-Leave, IGMP 20-5
inaccessible authentication bypass 8-13
interface
number 9-9
range macros 9-12
interface command 9-9
interface configuration mode 2-3
interfaces
configuration guidelines 9-15
configuring 9-9
configuring duplex mode 9-15
configuring speed 9-15
counters, clearing 9-22
described 9-19
descriptive name, adding 9-19
displaying information about 9-21
flow control 9-17
management 1-9
monitoring 9-21
naming 9-19
physical, identifying 9-9
range of 9-10
restarting 9-22
shutting down 9-22
supported 9-9
types of 9-1
interfaces range macro command 9-12
Interior Gateway Protocol
See IGP
internal BGP
See IBGP
internal neighbors, BGP 31-45
Internet Control Message Protocol
See ICMP
Internet Group Management Protocol
See IGMP
Inter-Switch Link
See ISL
inter-VLAN routing 1-7, 31-2
Intrusion Detection System
See IDS
IOS File System
See IFS
ip access-group command 29-22

IP ACLs
applying to an interface 29-21
extended, creating 29-11
for QoS classification 28-7
implicit deny 29-10, 29-16, 29-18
implicit masks 29-10
logging 29-18
named 29-16
standard, creating 29-10
undefined 29-23
virtual terminal lines, setting on 29-21
IP addresses
candidate or member 5-3, 5-13
classes of 31-5
cluster access 5-2
command switch 5-3, 5-11, 5-13
default configuration 31-4
discovering 6-29
for IP routing 31-4
MAC address association 31-8
monitoring 31-17
redundant clusters 5-11
standby command switch 5-11, 5-13
See also IP information
IP broadcast address 31-15
ip cef command 31-72
IP directed broadcasts 31-13
ip igmp profile command 20-23
IP information
assigned
manually 3-10
through DHCP-based autoconfiguration 3-3
default configuration 3-3
IP multicast routing
addresses
all-hosts 3-1, 3-3
all-multicast-routers 3-1, 3-3
host group address range 3-1, 3-3
administratively-scoped boundaries, described 3-36

IP multicast routing (continued)

and IGMP snooping 20-1, 20-6
Auto-RP
adding to an existing sparse-mode cloud 3-14
benefits of 3-14
clearing the cache 3-52
configuration guidelines 3-10
filtering incoming RP announcement messages 3-16
overview 3-5
preventing candidate RP spoofing 3-16
preventing join messages to false RPs 3-16
setting up in a new internetwork 3-14
using with BSR 3-22
bootstrap router
configuration guidelines 3-10
configuring candidate BSRs 3-20
configuring candidate RPs 3-21
defining the IP multicast boundary 3-19
defining the PIM domain border 3-18
overview 3-5
using with Auto-RP 3-22
Cisco implementation 3-2
configuring
basic multicast routing 3-10
IP multicast boundary 3-36
TTL threshold 3-34
default configuration 3-9
enabling
multicast forwarding 3-11
PIM mode 3-11
group-to-RP mappings
Auto-RP 3-5
BSR 3-5
MBONE
deleting sdr cache entries 3-52
described 3-33
displaying sdr cache 3-53
enabling sdr listener support 3-34

IP multicast routing (continued)
MBONE
limiting DVMRP routes advertised 3-47
limiting sdr cache entry lifetime 3-34
SAP packets for conference session announcement 3-33
Session Directory (sdr) tool, described 3-33
monitoring
packet rate loss 3-53
peering devices 3-53
tracing a path 3-53
multicast forwarding, described 3-6
PIMv1 and PIMv2 interoperability 3-9
protocol interaction 3-2
reverse path check (RPF) 3-6
routing table
deleting 3-52
displaying 3-52
RP
assigning manually 3-12
configuring Auto-RP 3-14
configuring PIMv2 BSR 3-18
monitoring mapping information 3-23
using Auto-RP and BSR 3-22
statistics, displaying system and network 3-52
TTL thresholds, described 3-34
See also CGMP
See also DVMRP
See also IGMP
See also PIM
IP phones
and IEEE 802.1x authentication 8-15
and QoS 13-1
automatic classification and queueing 28-17
configuring 13-3
trusted boundary for QoS 28-32
IP precedence 28-2
IP-precedence-to-DSCP map for QoS 28-53
IP protocols
in ACLs 29-13
routing 1-7
IP routes, monitoring 31-84
IP routing
connecting interfaces with 9-8
enabling 31-18
IP Service Level Agreements
See IP SLAs
IP service levels, analyzing 1-1
IP SLAs
benefits 1-2
configuration guidelines 1-6
Control Protocol 1-3
default configuration 1-6
definition 1-1
ICMP echo operation 1-10
measuring network performance 1-2
monitoring 1-13
multioperations scheduling 1-5
operation 1-2
responder
described 1-3
enabling 1-7
response time 1-4
scheduling 1-5
SNMP support 1-2
supported metrics 1-2
threshold monitoring 1-5
UDP jitter operation 1-8
IP source guard
and 802.1x 18-20
and DHCP snooping 18-18
and EtherChannels 18-20
and port security 18-20
and private VLANs 18-20
and routed ports 18-20
and TCAM entries 18-20
and trunk interfaces 18-20

IP source guard (continued)
and VRF 18-20
binding configuration
automatic 18-18
manual 18-18
binding table 18-18
configuration guidelines 18-20
default configuration 18-19
described 18-18
disabling 18-21
displaying
bindings 18-22
configuration 18-22
enabling 18-20
filtering
source IP address 18-19
source IP and MAC address 18-19
source IP address filtering 18-19
source IP and MAC address filtering 18-19
static bindings
adding 18-20
deleting 18-21
IP traceroute
executing 6-13
overview 6-13
IP unicast routing
address resolution 31-8
administrative distances 31-74, 31-82
ARP 31-8
assigning IP addresses to Layer 3 interfaces 31-6
authentication keys 31-83
broadcast
address 31-15
flooding 31-16
packets 31-13
storms 31-13
classless routing 31-7
configuring static routes 31-74

IP unicast routing (continued)
default
addressing configuration 31-4
gateways 31-11
networks 31-75
routes 31-75
routing 31-2
directed broadcasts 31-13
dynamic routing 31-2
enabling 31-18
EtherChannel Layer 3 interface 31-3
IGP 31-24
inter-VLAN 31-2
IP addressing
classes 31-5
configuring 31-4
IRDP 31-12
Layer 3 interfaces 31-3
MAC address and IP address 31-8
passive interfaces 31-81
protocols
distance-vector 31-2
dynamic 31-2
link-state 31-2
proxy ARP 31-8
redistribution 31-76
reverse address resolution 31-8
routed ports 31-3
static routing 31-2
steps to configure 31-3
subnet mask 31-5
subnet zero 31-6
supernet 31-7
UDP 31-15
with SVIs 31-3
See also BGP
See also EIGRP
See also OSPF
See also RIP

ip unreachables command 29-6
IRDP
configuring 31-12
definition 31-12
support for 1-8
ISL
and trunk ports 9-3
encapsulation 1-4, 11-15
trunking with IEEE 802.1 tunneling 14-4

J

join messages, IGMP 20-3

K

KDC
described 7-32
See also Kerberos
Kerberos
authenticating to
boundary switch 7-34
KDC 7-34
network services 7-35
configuration examples 7-32
configuring 7-35
credentials 7-32
cryptographic software image 7-32
described 7-32
KDC 7-32
operation 7-34
realm 7-33
server 7-33
switch as trusted third party 7-32
terms 7-33
TGT 7-34
tickets 7-32
key distribution center
See KDC

L

l2protocol-tunnel command 14-12
LACP
Layer 2 protocol tunneling 14-9
See EtherChannel
Layer 2 frames, classification with CoS 28-2
Layer 2 interfaces, default configuration 9-14
Layer 2 protocol tunneling
configuring 14-9
configuring for EtherChannels 14-13
default configuration 14-10
defined 14-8
guidelines 14-11
Layer 2 traceroute
and ARP 6-15
and CDP 6-15
described 6-14
IP addresses and subnets 6-15
MAC addresses and VLANs 6-15
multicast traffic 6-15
multiple devices on a port 6-15
unicast traffic 6-14
usage guidelines 6-15
Layer 3 features 1-7
Layer 3 interfaces
assigning IP addresses to 31-6
changing from Layer 2 mode 31-6
types of 31-3
Layer 3 packets, classification methods 28-2
LDAP 4-2
leave processing, IGMP 20-10
Lightweight Directory Access Protocol
See LDAP
line configuration mode 2-3

Link Aggregation Control Protocol
See EtherChannel
Link Failure
detecting unidirectional 16-7
link redundancy
See Flex Links
links, unidirectional 23-1
link state advertisements (LSAs) 31-28
link-state protocols 31-2
logging messages, ACL 29-11
login authentication
with RADIUS 7-23
with TACACS+ 7-14
login banners 6-17
log messages
See system message logging
long-distance, high-bandwidth transport configuration example 1-18
Long-Reach Ethernet (LRE) technology 1-12
loop guard
described 17-13
enabling 17-21
support for 1-4

M

mac access-group command 29-29
MAC ACLs and Layer 2 interfaces 29-29
MAC addresses
aging time 6-21
and VLAN association 6-20
building the address table 6-20
default configuration 6-20
discovering 6-29
displaying 6-26
displaying in DHCP snooping binding table 18-18
displaying in the IP source binding table 18-22

MAC addresses (continued)
dynamic
learning 6-20
removing 6-21
in ACLs 29-27
IP address association 31-8
static
adding 6-24
allowing 6-25
characteristics of 6-23
dropping 6-25
removing 6-24
sticky secure, adding 21-8
MAC address multicast entries, monitoring 20-15
MAC address-table move update
configuration guidelines 18-5
configuring 18-7
default configuration 18-5
description 18-3
monitoring 18-9
MAC address-to-VLAN mapping 11-27
MAC extended access lists 28-5, 28-39, 29-27
macros
See Smartports macros
magic packet 8-16
manageability features 1-2
management options
benefits
clustering 1-10
Network Assistant 1-10
CLI 2-1
CNS 4-1
overview 1-9
management VLAN
considerations in switch clusters 5-7
discovery through different management VLANs 5-7

MANs
CWDM configuration example 1-18
long-distance, high-bandwidth transport configuration example 1-18
mapping tables for QoS
configuring
CoS-to-DSCP 28-52
CoS-to-egress-queue 28-59
DSCP 28-52
DSCP-to-CoS 28-55
DSCP-to-DSCP-mutation 28-57
DSCP-to-threshold 28-61
IP-precedence-to-DSCP 28-53
policed-DSCP 28-54
described 28-10
marking
action in policy map 28-44
action with aggregate policers 28-50
described 28-4, 28-8
matching, ACLs 29-8
maximum aging time
MSTP 16-23
STP 15-22, 15-23
maximum hop count, MSTP 16-24
maximum-paths command 31-49, 31-73
membership mode, VLAN port 11-3
member switch
automatic discovery 5-4
defined 5-2
passwords 5-13
recovering from lost connectivity 6-10
requirements 5-3
See also candidate switch, cluster standby group, and standby command switch
memory, optimizing 6-26
messages
logging ACL violations 29-18
to users through banners 6-17
metrics, in BGP 31-49
metric translations, between routing protocols 31-78
metropolitan-area networks
See MANs
metro tags 14-2
MIBs
accessing files with FTP A-3
location of files A-3
overview 27-1
SNMP interaction with 27-4
supported A-1
minimum-reserve levels
assigning to a queue 28-15, 28-67
configuring the buffer size 28-15, 28-66
default size 28-15
mini-point-of-presence
See POP
mirroring traffic for analysis 24-1
mismatches, autonegotiation 6-10
modules, GBIC
1000BASE-LX/LH 1-14
1000BASE-SX 1-14
1000BASE-T 1-14
1000BASE-ZX 1-14
CWDM 1-18
GigaStack 1-12
monitoring
access groups 29-41
ACL
configuration 29-41
configuration conflicts 29-43
fit in hardware 29-44
information 29-40
BGP 31-60
cables for unidirectional links 23-1
CDP 22-4
CEF 31-73
EIGRP 31-40
fallback bridging 5-12
features 1-8

monitoring (continued)
Flex Links 18-9
HSRP 32-10
IEEE 802.1Q tunneling 14-17
IGMP
filters 20-27
snooping 20-14
interfaces 9-21
IP
address tables 31-17
multicast routing 3-51
routes 31-84
IP SLAs operations 1-13
Layer 2 protocol tunneling 14-17
MAC address-table move update 18-9
MSDP peers 4-17
multicast router ports 20-15
multi-VRF CE 31-71
MVR 20-21
network traffic for analysis with probe 24-1
OSPF 31-33
port blocking 21-17
port protection 21-17
RP mapping information 3-23
source-active messages 4-17
speed and duplex mode 9-16
traffic flowing among switches 25-1
traffic suppression 21-17
tunneling 14-17
VLAN
filters 29-42
maps 29-42
VLANs 11-14
VMPS 11-32
VTP 12-15
MSDP
and dense-mode regions
sending SA messages to 4-16
specifying the originating address 4-17

MSDP (continued)
benefits of 4-3
clearing MSDP connections and statistics 4-18
controlling source information
forwarded by switch 4-11
originated by switch 4-8
received by switch 4-13
default configuration 4-4
filtering
incoming SA messages 4-13
SA messages to a peer 4-12
SA requests from a peer 4-10
join latency, defined 4-6
meshed groups
configuring 4-15
defined 4-15
originating address, changing 4-17
overview 4-1
peer-RPF flooding 4-2
peers
configuring a default 4-4
monitoring 4-17
peering relationship, overview 4-1
requesting source information from 4-8
shutting down 4-15
source-active messages
caching 4-6
clearing cache entries 4-18
defined 4-2
filtering from a peer 4-10
filtering incoming 4-13
filtering to a peer 4-12
limiting data with TTL 4-13
monitoring 4-17
restricting advertised sources 4-9

MSTP
boundary ports
configuration guidelines 16-15
BPDU filtering
described 17-3
enabling 17-16
BPDU guard
described 17-2
enabling 17-15
CIST, described 16-3
configuration guidelines 16-15, 17-14
configuring
forward-delay time 16-23
hello time 16-22
link type for rapid convergence 16-24
maximum aging time 16-23
maximum hop count 16-24
MST region 16-16
neighbor type 16-25
path cost 16-21
port priority 16-20
root switch 16-17
secondary root switch 16-19
switch priority 16-21
CST
defined 16-3
operations between regions 16-3
default configuration 16-14
default optional feature configuration 17-14
described 16-2
displaying status 16-26
enabling the mode 16-16
EtherChannel guard
described 17-11
enabling 17-20
extended system ID
effects on root switch 16-17
effects on secondary root switch 16-19
unexpected behavior 16-18

MSTP (continued)
IEEE 802.1s
implementation 16-6
instances supported 15-9
interface state, blocking to forwarding 17-2
interoperability and compatibility among modes 15-10
interoperability with IEEE 802.1D
described 16-8
restarting migration process 16-26
IST
defined 16-2
master 16-3
operations within a region 16-3
loop guard
described 17-13
enabling 17-21
mapping VLANs to MST instance 16-16
MST region
described 16-2
hop-count mechanism 16-5
supported spanning-tree instances 16-2
optional features supported 1-4
Port Fast
described 17-2
enabling 17-14
preventing root switch selection 17-12
root guard
described 17-12
enabling 17-20
root switch
configuring 16-18
effects of extended system ID 16-17
unexpected behavior 16-18
shutdown Port Fast-enabled port 17-2
multicast groups
and IGMP snooping 20-6
Immediate Leave 20-5
joining 20-3

multicast groups (continued)
leaving 20-4
static joins 20-9
multicast packets
ACLs on 29-40
multicast packets, blocking 21-6
multicast router ports
adding 20-9
monitoring 20-15
Multicast Source Discovery Protocol
See MSDP
multicast storm control
See storm control
multicast storm-control command 21-4
Multicast VLAN Registration
See MVR
multioperations scheduling, IP SLAs 1-5
Multiple Spanning Tree Protocol
See MSTP
multiple VPN routing/forwarding in customer edge devices
See multi-VRF CE
multi-VRF CE
configuration example 31-67
configuration guidelines 31-64
configuring 31-64
default configuration 31-64
defined 31-61
displaying 31-71
monitoring 31-71
network components 31-64
packet-forwarding process 31-63
support for 1-7
MVR
configuring interfaces 20-20
default configuration 20-18
described 20-15
modes 20-19
monitoring 20-21

MVR (continued)
setting global parameters 20-19
support for 1-2

N

NAC
AAA down policy 1-6
critical authentication 8-13, 8-34
IEEE 802.1x authentication using a RADIUS server 8-38
IEEE 802.1x validation using RADIUS server 8-38
inaccessible authentication bypass 1-6, 8-34
Layer 2 IEEE 802.1x validation 1-6, 8-38
Layer 2 IP validation 1-6
named IP ACLs 29-16
NameSpace Mapper
See NSM
native VLAN
and IEEE 802.1Q tunneling 14-4
configuring 11-22
default 11-22
neighbor discovery/recovery, EIGRP 31-34
neighbors, BGP 31-55
Network Admission Control
See NAC
Network Admission Control Software Configuration Guide 8-40, 8-41
Network Assistant
described 1-2, 1-9
downloading image files 1-2
upgrading a switch B-18
network configuration examples
increasing network performance 1-11
large network 1-15
long-distance, high-bandwidth transport 1-18
providing network services 1-11
small to medium-sized network 1-14

network design
performance 1-11
services 1-11
network management
CDP 22-1
RMON 25-1
SNMP 27-1
network performance, measuring with IP SLAs 1-2
Network Time Protocol
See NTP
no commands 2-4
non-IP traffic filtering 29-27
nontrunking mode 11-17
normal-range VLANs
configuration modes 11-6
defined 11-1
no switchport command 9-5
not-so-stubby areas
See NSSA
NSM 4-3
NSSA, OSPF 31-28
NTP
associations
authenticating 6-4
defined 6-2
enabling broadcast messages 6-6
peer 6-5
server 6-5
default configuration 6-4
displaying the configuration 6-11
overview 6-2
restricting access
creating an access group 6-8
disabling NTP services per interface 6-10
source IP address, configuring 6-10
stratum 6-2
support for 1-3
synchronizing devices 6-5

NTP (continued)
time
services 6-2
synchronizing 6-2

O

OADM modules
See CWDM OADM modules
Open Shortest Path First
See OSPF
optical add/drop multiplexer modules
See CWDM OADM modules
optimizing system resources 6-26
options, management 1-9
OSPF
area parameters, configuring 31-28
configuring 31-26
default configuration
metrics 31-30
route 31-30
settings 31-25
described 31-24
interface parameters, configuring 31-27
LSA group pacing 31-32
monitoring 31-33
router IDs 31-32
route summarization 31-30
support for 1-7
virtual links 31-30
out-of-profile markdown 1-7
output interface, getting information about 6-20

P

packet modification, with QoS 28-16
PAgP
  Layer 2 protocol tunneling 14-9
  See EtherChannel
parallel paths, in routing tables 31-73
passive interfaces
  configuring 31-81
  OSPF 31-30
pass-through mode 28-34
passwords
  default configuration 7-2
  disabling recovery of 7-5
  encrypting 7-3
  for security 1-5
  in clusters 5-13
  overview 7-1
  setting
    enable 7-3
    enable secret 7-3
    Telnet 7-6
    with usernames 7-6
  VTP domain 12-8
path cost
  MSTP 16-21
  STP 15-18
PBR
  defined 31-79
  enabling 31-80
  fast-switched policy-based routing 31-80
  local policy-based routing 31-80
  support for 1-8
PC (passive command switch) 5-10
peers, BGP 31-55
performance, network design 1-11
performance features 1-2
persistent self-signed certificate 7-42
per-VLAN spanning-tree plus
  See PVST+
PE to CE routing, configuring 31-67
physical ports 9-2
PIM
  default configuration 3-9
  dense mode
    overview 3-4
    rendezvous point (RP), described 3-5
    RPF lookups 3-7
  displaying neighbors 3-52
  enabling a mode 3-11
  overview 3-4
  router-query message interval, modifying 3-26
  shared tree and source tree, overview 3-23
  shortest path tree, delaying the use of 3-25
  sparse mode
    join messages and shared tree 3-5
    overview 3-5
    prune messages 3-5
    RPF lookups 3-7
  support for 1-8
  versions
    interoperability 3-9
    troubleshooting interoperability problems 3-23
    v2 improvements 3-4
PIM-DVMRP, as snooping method 20-8
ping
  character output description 6-12
  executing 6-11
  overview 6-11
PoE
  auto mode 9-7
  CDP with power consumption, described 9-6
  CDP with power negotiation, described 9-6
  Cisco intelligent power management 9-6
  configuring 9-16
  devices supported 9-5

PoE (continued)

high-power devices operating in low-power mode 9-6

powered-device detection and initial power allocation 9-6

power management modes 9-7

power negotiation extensions to CDP 9-6

standards supported 9-6

troubleshooting 6-16

policed-DSCP map for QoS 28-54

policers

configuring

for each matched traffic class 28-44

for more than one traffic class 28-50

described 28-4

displaying 28-69

number of 1-7, 28-9

types of 28-8

policing

described 28-4

token bucket algorithm 28-8

policy-based routing

See PBR

policy maps for QoS

characteristics of 28-44

configuring 28-44

described 28-7

displaying 28-70

POP 1-17

port ACLs

and voice VLAN 29-4

defined 29-2

limitations 29-4

Port Aggregation Protocol

See EtherChannel

port-based authentication

accounting 8-8

accounting services 1-5

port-based authentication (continued)

authentication server

defined 8-2

RADIUS server 8-2

client, defined 8-2

configuration guidelines 8-21

configuring

guest VLAN 8-32

host mode 8-27

IEEE 802.1x accounting 8-31

IEEE 802.1x authentication 8-24

inaccessible authentication bypass 8-34

manual re-authentication of a client 8-28

periodic re-authentication 8-27

quiet period 8-28

RADIUS server 8-26

RADIUS server parameters on the switch 8-25

restricted VLAN 8-33

switch-to-client frame-retransmission number 8-30

switch-to-client retransmission time 8-29

default configuration 8-20

described 8-1

device roles 8-2

displaying statistics 8-42

EAPOL-start frame 8-5

EAP-request/identity frame 8-5

EAP-response/identity frame 8-5

enabling

IEEE 802.1x with guest VLAN 8-11

IEEE 802.1x with per-user ACLs 8-10

IEEE 802.1x with port security 8-15

IEEE 802.1x with restricted VLAN 8-13

IEEE 802.1x with VLAN assignment 8-9

IEEE 802.1x with voice VLAN 8-15

encapsulation 8-3

guest VLAN

configuration guidelines 8-12, 8-13

host mode 8-7

port-based authentication (continued)
inaccessible authentication bypass
configuring 8-34
described 8-13
guidelines 8-23
initiation and message exchange 8-5
magic packet 8-16
method lists 8-24
multiple-hosts mode, described 8-8
per-user ACLs, AAA authorization 8-24
ports
authorization state and dot1x port-control command 8-7
authorized and unauthorized 8-7
critical 8-13
port security, multiple-hosts mode 8-8
resetting to default values 8-42
software upgrade changes 8-23
support for 1-5
switch
as proxy 8-3
RADIUS client 8-3
upgrading from a previous release 28-21
VLAN assignment, AAA authorization 8-24
wake-on-LAN, described 8-16
port blocking 1-2, 21-6
port-channel
See EtherChannel
Port Fast
described 17-2
enabling 17-14
mode, spanning tree 11-29
support for 1-4
port membership modes, VLAN 11-3
port priority
MSTP 16-20
STP 15-17
ports
access 9-2
blocking 21-6
dynamic access 11-3
forwarding, resuming 21-7
IEEE 802.1Q tunnel 11-3
protected 21-5
routed 9-4
secure 21-7
static-access 11-3, 11-10
switch 9-2
trunks 11-3, 11-15
VLAN assignments 11-10
port security
aging 21-15
and QoS trusted boundary 28-32
configuration guidelines 21-10
configuring 21-11
default configuration 21-9
described 21-7
displaying 21-17
on trunk ports 21-12
sticky learning 21-8
violations 21-8
with other features 21-10
port-shutdown response, VMPS 11-27
Power over Ethernet
See PoE
preemption
default configuration 18-5
preemption delay
default configuration 18-5
preferential treatment of traffic
See QoS
prefix lists, BGP 31-53
preventing unauthorized access 7-1
primary links 18-2

priority
HSRP 32-6
overriding CoS 13-5
trusting CoS 13-5
private VLAN edge ports
See protected ports
privileged EXEC mode 2-2
privilege levels
changing the default for lines 7-9
command switch 5-14
exiting 7-9
logging into 7-9
mapping on member switches 5-14
overview 7-2, 7-7
setting a command with 7-8
protected ports 1-5, 21-5
protocol-dependent modules, EIGRP 31-35
Protocol-Independent Multicast Protocol
See PIM
provider edge devices 31-62
proxy ARP
configuring 31-10
definition 31-8
with IP routing disabled 31-11
pruning, VTP
enabling 12-13
enabling on a port 11-22
examples 12-5
overview 12-4
pruning-eligible list
changing 11-22
for VTP pruning 12-4
VLANs 12-14
PVST+
described 15-9
IEEE 802.1Q trunking interoperability 15-10
instances supported 15-9

Q

QoS
and MQC commands 28-1
auto-QoS
categorizing traffic 28-17
configuration and defaults display 28-23
configuration guidelines 28-21
described 28-17
displaying 28-23
effects on NVRAM configuration 28-20
egress queue defaults 28-18
enabling for VoIP 28-22
generated commands 28-19
basic model 28-4
classification
class maps, described 28-7
defined 28-4
flowchart 28-6
forwarding treatment 28-3
in frames and packets 28-3
IP ACLs, described 28-5, 28-7
MAC ACLs, described 28-5, 28-7
pass-through mode, described 28-34
per physical port 28-40
per-port per-VLAN 28-41
policy maps, described 28-7
port default, described 28-5
trust DSCP, described 28-5
trusted CoS, described 28-5
trust IP precedence, described 28-5
types for IP traffic 28-5
types for non-IP traffic 28-5
class maps
configuring per physical port 28-40
configuring per-port per-VLAN 28-41
displaying 28-69

QoS (continued)

configuration examples
- distribution layer 28-73
- existing wiring closet 28-71
- intelligent wiring closet 28-72

configuration guidelines
- auto-QoS 28-21
- standard QoS 28-27

configuring
- aggregate policers 28-50
- auto-QoS 28-17
- default port CoS value 28-32
- DSCP maps 28-52
- DSCP trust states bordering another domain 28-34
- egress queues on 10/100 Ethernet ports 28-65
- egress queues on Gigabit-capable Ethernet ports 28-58
- IP extended ACLs 28-37
- IP standard ACLs 28-36
- MAC ACLs 28-39
- pass-through mode 28-34
- policy maps 28-44
- port trust states within the domain 28-30
- trusted boundary 28-32

default auto configuration 28-17

default standard configuration 28-26

displaying statistics 28-69

enabling globally 28-29

flowcharts
- classification 28-6
- policing and marking 28-10
- queueing and scheduling 28-12, 28-15

implicit deny 28-7

IP phones
- automatic classification and queueing 28-17
- detection and trusted settings 28-17, 28-32

QoS (continued)

mapping tables
- CoS-to-DSCP 28-52
- CoS-to-egress-queue 28-59
- displaying 28-70
- DSCP-to-CoS 28-55
- DSCP-to-DSCP-mutation 28-57
- DSCP-to-threshold 28-61
- IP-precedence-to-DSCP 28-53
- policed-DSCP 28-54
- types of 28-10

marked-down actions 28-47

marking, described 28-4, 28-8

overview 28-2

packet modification 28-16

pass-through mode 28-34

policers
- configuring 28-47, 28-50
- described 28-8
- displaying 28-69
- number of 28-9
- types of 28-8

policies, attaching to an interface 28-9

policing
- described 28-4, 28-8
- token bucket algorithm 28-8

policy maps
- characteristics of 28-44
- configuring 28-44
- displaying 28-70

queueing, defined 28-4

queues
- CoS-to-egress-queue map 28-59
- for 10/100 Ethernet ports 28-14
- high priority (expedite) 28-13, 28-63
- minimum-reserve levels 28-66
- serviced by WRR 28-13, 28-16
- size of 28-12, 28-15
- size ratios 28-59

QoS (continued)
queues
tail-drop threshold percentages 28-12, 28-60
WRED drop-percentage thresholds 28-12, 28-62
WRR scheduling 28-64
scheduling
allocating bandwidth on 10/100 Ethernet ports 28-68
allocating bandwidth on Gigabit-capable ports 28-64
defined 28-4
support for 1-6
tail drop
configuring drop threshold percentages 28-60
described 28-13
trust states
bordering another domain 28-34
described 28-5
trusted device 28-32
within the domain 28-30
WRED
configuring drop-percentage thresholds 28-62
described 28-13
WRR scheduling 28-64
quality of service
See QoS
queries, IGMP 20-3
query solicitation, IGMP 20-12

R

RADIUS
attributes
vendor-proprietary 7-31
vendor-specific 7-29
configuring
accounting 7-28
authentication 7-23
authorization 7-27

RADIUS (continued)
configuring
communication, global 7-21, 7-29
communication, per-server 7-20, 7-21
multiple UDP ports 7-20
default configuration 7-20
defining AAA server groups 7-25
described 7-18
displaying the configuration 7-31
identifying the server 7-20
in clusters 5-14
limiting the services to the user 7-27
method list, defined 7-19
operation of 7-19
suggested network environments 7-18
tracking services accessed by user 7-28
Random Early Detection, described 28-13
range
macro 9-12
of interfaces 9-10
rapid convergence 16-10
rapid per-VLAN spanning-tree plus
See rapid PVST+
rapid PVST+
described 15-9
IEEE 802.1Q trunking interoperability 15-10
instances supported 15-9
rapid-PVST+ 11-2
Rapid Spanning Tree Protocol
See RSTP
RARP 31-8
RCP
configuration files
downloading B-16
overview B-15
preparing the server B-15
uploading B-17

RCP (continued)
image files
deleting old image B-31
downloading B-29
preparing the server B-28
uploading B-31
reconfirmation interval, VMPS, changing 11-31
recovery procedures 6-1
redundancy
EtherChannel 30-2
features 1-3
HSRP 32-1
STP
backbone 15-8
multidrop backbone 17-5
path cost 11-25
port priority 11-23
redundant links and UplinkFast 17-17
reliable transport protocol, EIGRP 31-34
reloading software 3-16
Remote Authentication Dial-In User Service
See RADIUS
Remote Copy Protocol
See RCP
Remote Network Monitoring
See RMON
report suppression, IGMP
described 20-5
disabling 20-13
resequencing ACL entries 29-16
resets, in BGP 31-48
resetting a UDLD-shutdown interface 23-6
responder, IP SLAs
described 1-3
enabling 1-7
response time, measuring with IP SLAs 1-4
restricted VLAN
configuring 8-33
using with port-based authentication 8-13
restricting access
NTP services 6-8
overview 7-1
passwords and privilege levels 7-2
RADIUS 7-17
TACACS+ 7-9
retry count, VMPS, changing 11-31
reverse address resolution 31-8
Reverse Address Resolution Protocol
See RARP
RFC
1058, RIP 31-19
1112, IP multicast and IGMP 20-2
1157, SNMPv1 27-2
1163, BGP 31-40
1166, IP addresses 31-5
1253, OSPF 31-24
1267, BGP 31-40
1305, NTP 6-2
1587, NSSAs 31-24
1757, RMON 25-2
1771, BGP 31-40
1901, SNMPv2C 27-2
1902 to 1907, SNMPv2 27-2
2236, IP multicast and IGMP 20-2
2273-2275, SNMPv3 27-2
RIP
advertisements 31-19
authentication 31-22
configuring 31-20
default configuration 31-19
described 31-19
hop counts 31-19
split horizon 31-22
summary addresses 31-22
support for 1-7

RMON
default configuration 25-3
displaying status 25-6
enabling alarms and events 25-3
groups supported 25-2
overview 25-1
statistics
collecting group Ethernet 25-5
collecting group history 25-5
support for 1-8
root guard
described 17-12
enabling 17-20
support for 1-4
root switch
MSTP 16-17
STP 15-15
route calculation timers, OSPF 31-30
route dampening, BGP 31-59
routed packets, ACLs on 29-39
routed ports
configuring 31-3
defined 9-4
in switch clusters 5-8
IP addresses on 9-20, 31-3
route-map command for policy-based routing 31-80
route maps
BGP 31-51
policy-based routing, defined 31-79
router ACLs 29-2
route reflectors, BGP 31-58
router ID, OSPF 31-32
route selection, BGP 31-49
route summarization, OSPF 31-30
route targets, VPN 31-64
routing
default 31-2
dynamic 31-2
redistribution of information 31-76
static 31-2
routing domain confederation, BGP 31-58
Routing Information Protocol
See RIP
routing protocol administrative distances 31-74
RSPAN
configuration guidelines 24-16
default configuration 24-8
destination ports 24-5
displaying status 24-24
IDS 24-2
interaction with other features 24-7
monitored ports 24-4
monitoring ports 24-5
overview 1-8, 24-1
received traffic 24-3
reflector port 24-5
session limits 24-8
sessions
creating 24-17
defined 24-3
limiting source traffic to specific VLANs 24-23
monitoring VLANs 24-21
removing source (monitored) ports 24-21
specifying monitored ports 24-17
source ports 24-4
transmitted traffic 24-4
VLAN-based 24-6
RSTP
active topology, determining 16-9
BPDU
format 16-12
processing 16-13
designated port, defined 16-9
designated switch, defined 16-9

RSTP (continued)
interoperability with IEEE 802.1D
described 16-8
restarting migration process 16-26
topology changes 16-13
overview 16-8
port roles
described 16-9
synchronized 16-11
proposal-agreement handshake process 16-10
rapid convergence
described 16-10
edge ports and Port Fast 16-10
point-to-point links 16-10, 16-24
root ports 16-10
root port, defined 16-9
See also MSTP
running configuration, saving 3-10

S

SC (standby command switch) 5-10
scheduled reloads 3-16
scheduling, IP SLAs operations 1-5
SDM
configuring 6-29
described 6-26
templates
number of 6-26
resources used for Fast Ethernet switches 6-27
resources used for Gigabit Ethernet switches 6-27
sdm prefer extended-match command 31-65
secure HTTP client
configuring 7-47
displaying 7-48
secure HTTP server
configuring 7-46
displaying 7-48
secure ports, configuring 21-7
secure remote connections 7-38
Secure Shell
See SSH
Secure Socket Layer
See SSL
security, port 21-7
security features 1-5
sequence numbers in log messages 26-8
server mode, VTP 12-3
service-provider networks
and customer VLANs 14-2
and IEEE 802.1Q tunneling 14-1
Layer 2 protocols across 14-8
Layer 2 protocol tunneling for EtherChannels 14-9
MSTP and RSTP 16-1
set-request operation 27-4
setup program, failed command switch replacement 6-7, 6-8
severity levels, defining in system messages 26-8
show access-lists hw-summary command 29-7
show cdp traffic command 22-5
show configuration command 9-19
show fm command 29-42
show forward command 6-19
show interfaces command 9-16, 9-19
show l2protocol command 14-12, 14-14, 14-15
show mac access-group command 29-29
show running-config command
displaying ACLs 29-22, 29-32, 29-34
interface description in 9-19
show tcam command 29-43
shutdown command on interfaces 9-22
shutdown threshold for Layer 2 protocol packets 14-10
Simple Network Management Protocol
See SNMP

Smartports macros
applying Cisco-default macros 10-6
applying global parameter values 10-5, 10-6
applying macros 10-4
applying parameter values 10-5, 10-7
configuration guidelines 10-3
creating 10-4
default configuration 10-2
defined 10-1
displaying 10-8
tracing 10-3
website 10-2
SNAP 22-1
SNMP
accessing MIB variables with 27-4
agent
described 27-4
disabling 27-7
and IP SLAs 1-2
community strings
configuring 27-8
for cluster switches 27-4
overview 27-4
configuration examples 27-16
default configuration 27-6
groups 27-9
ifIndex values 27-5
in-band management 1-3
in clusters 5-14
informs
and trap keyword 27-11
described 27-5
differences from traps 27-5
enabling 27-14
limiting access by TFTP servers 27-16
limiting system log messages to NMS 26-10
manager functions 1-9, 27-3
managing clusters with 5-15
SNMP (continued)
MIBs
location of A-3
supported A-1
notifications 27-5
overview 27-1, 27-4
status, displaying 27-17
system contact and location 27-15
trap manager, configuring 27-13, 27-15
traps
described 27-3, 27-5
differences from informs 27-5
enabling 27-11, 27-14
enabling MAC address notification 6-21
overview 27-1, 27-4
types of 27-11
users 27-9
versions supported 27-2
snooping, IGMP 20-1
software images
location in flash B-19
recovery procedures 6-2
scheduling reloads 3-16
tar file format, described B-19
See also downloading and uploading
source addresses, in ACLs 29-13
SPAN
configuration guidelines 24-9
default configuration 24-8
destination ports 24-5
displaying status 24-24
IDS 24-2
interaction with other features 24-7
monitored ports 24-4
monitoring ports 24-5
overview 1-8, 24-1
ports, restrictions 21-11
received traffic 24-3
session limits 24-8

SPAN (continued)
sessions
creating 24-10
defined 24-3
limiting source traffic to specific VLANs 24-15
monitoring VLANs 24-14
removing destination (monitoring) ports 24-13
removing source (monitored) ports 24-13
specifying monitored ports 24-10
source ports 24-4
transmitted traffic 24-4
VLAN-based 24-6
spanning tree and native VLANs 11-18
Spanning Tree Protocol
See STP
speed, configuring on interfaces 9-15
split horizon, RIP 31-22
SSH
configuring 7-39
cryptographic software image 7-37
described 7-38
encryption methods 7-38
user authentication methods, supported 7-38
SSL
configuration guidelines 7-44
configuring a secure HTTP client 7-47
configuring a secure HTTP server 7-45
cryptographic software image 7-41
described 7-41
monitoring 7-48
Stack Membership Discovery Protocol 17-6
standby command switch
configuring
considerations 5-11
defined 5-2
priority 5-10
requirements 5-3
virtual IP address 5-11
See also cluster standby group and HSRP
standby group, cluster
See cluster standby group and HSRP
standby ip command 32-5
standby links 18-2
standby router 32-1
standby timers, HSRP 32-8
startup configuration
booting
manually 3-12
specific image 3-13
clearing B-18
configuration file
automatically downloading 3-11
specifying the filename 3-11
default boot configuration 3-11
static access ports
assigning to VLAN 11-10
defined 9-3, 11-3
static addresses
See addresses
static IP routing 1-8
static MAC addressing 1-5
static routes, configuring 31-74
static routing 31-2
static VLAN membership 11-2
statistics
CDP 22-4
IEEE 802.1x 8-42
interface 9-21
IP multicast routing 3-52
OSPF 31-33
QoS ingress and egress 28-69
RMON group Ethernet 25-5
RMON group history 25-5
SNMP input and output 27-17
VTP 12-15

sticky learning
configuration file 21-8
defined 21-8
disabling 21-8
enabling 21-8
saving addresses 21-8
storm control
configuring 21-3
default configuration 21-3
described 21-1
disabling 21-4
displaying 21-17
thresholds 21-1
STP
accelerating root port selection 17-4
BackboneFast
described 17-9
enabling 17-19
BPDU filtering
described 17-3
enabling 17-16
BPDU guard
described 17-2
enabling 17-15
BPDU message exchange 15-3
configuration guidelines 15-12, 17-14
configuring
forward-delay time 15-21
hello time 15-21
in cascaded stack 15-22
maximum aging time 15-22, 15-23
path cost 15-18
port priority 15-17
root switch 15-15
secondary root switch 15-16
spanning-tree mode 15-13
switch priority 15-19
counters, clearing 15-24

STP (continued)

cross-stack UplinkFast
described 17-5
enabling 17-18
default configuration 15-11
default optional feature configuration 17-14
designated port, defined 15-3
designated switch, defined 15-3
detecting indirect link failures 17-10
disabling 15-14
displaying status 15-24
EtherChannel guard
described 17-11
enabling 17-20
extended system ID
affects on root switch 15-15
affects on the secondary root switch 15-16
overview 15-4
unexpected behavior 15-15
features supported 1-3
inferior BPDU 15-3
instances supported 15-9
interface state, blocking to forwarding 17-2
interface states
blocking 15-5
disabled 15-7
forwarding 15-5, 15-6
learning 15-6
listening 15-6
overview 15-4
interoperability and compatibility among modes 15-10
Layer 2 protocol tunneling 14-7
limitations with IEEE 802.1Q trunks 15-10
load sharing
overview 11-23
using path costs 11-25
using port priorities 11-23

STP (continued)
loop guard
described 17-13
enabling 17-21
modes supported 15-9
multicast addresses, affect of 15-8
optional features supported 1-4
overview 15-2
path costs 11-25, 11-26
Port Fast
described 17-2
enabling 17-14
port priorities 11-24
preventing root switch selection 17-12
protocols supported 15-9
redundant connectivity 15-8
root guard
described 17-12
enabling 17-20
root port, defined 15-3
root switch
affects of extended system ID 15-4, 15-15
configuring 15-15
election 15-3
unexpected behavior 15-15
settings in a cascaded stack 15-22
shutdown Port Fast-enabled port 17-2
superior BPDU 15-3
timers, described 15-20
UplinkFast
described 17-3
enabling 17-17
VLAN-bridge 15-10
stratum, NTP 6-2
stub areas, OSPF 31-28
subnet mask 31-5
subnet zero 31-6
summer time 6-13
SunNet Manager 1-9
supernet 31-7
SVIs
and IP unicast routing 31-3
and router ACLs 29-3
connecting VLANs 9-8
defined 9-4
routing between VLANs 11-2
switch clustering technology 5-1
switch console port 1-3
switched packets, ACLs on 29-38
switched ports 9-2
switchport block multicast command 21-6
switchport block unicast command 21-7
switchport command 9-14
switchport mode dot1q-tunnel command 14-6
switchport protected command 21-6
switch priority
MSTP 16-21
STP 15-19
switch software features 1-1
switch virtual interfaces
See SVIs
synchronization, BGP 31-45
syslog
See system message logging
system clock
configuring
daylight saving time 6-13
manually 6-11
summer time 6-13
time zones 6-12
displaying the time and date 6-12
overview 6-1
See also NTP
System Database Management
See SDM

system message logging
default configuration 26-3
defining error message severity levels 26-8
disabling 26-4
displaying the configuration 26-12
enabling 26-4
facility keywords, described 26-12
level keywords, described 26-9
limiting messages 26-10
message format 26-2
overview 26-1
sequence numbers, enabling and disabling 26-8
setting the display destination device 26-5
synchronizing log messages 26-6
syslog facility 1-8
timestamps, enabling and disabling 26-7
UNIX syslog servers
configuring the daemon 26-11
configuring the logging facility 26-11
facilities supported 26-12
system MTU
IEEE 802.1Q tunneling 14-5
maximums 14-5
system name
default configuration 6-15
default setting 6-15
manual configuration 6-15
See also DNS
system prompt
default setting 6-14, 6-15
system resource templates 6-26

T

TACACS+
accounting, defined 7-11
authentication, defined 7-11
authorization, defined 7-11

TACACS+ (continued)
configuring
accounting 7-17
authentication key 7-13
authorization 7-16
login authentication 7-14
default configuration 7-13
displaying the configuration 7-17
identifying the server 7-13
in clusters 5-14
limiting the services to the user 7-16
operation of 7-12
overview 7-10
tracking services accessed by user 7-17
tagged packets
IEEE 802.1Q 14-3
Layer 2 protocol 14-7
tail drop
described 28-13
support for 1-7
tar files
creating B-5
displaying the contents of B-6
extracting B-7
image file format B-19
TCAMs
ACL regions 29-47
ACLs not loading in 29-45
allocations, monitoring 29-47
monitoring usage 29-47
Telnet
accessing management interfaces 2-10
number of connections 1-3
setting a password 7-6
templates, system resources 6-26
temporary self-signed certificate 7-42
Terminal Access Controller Access Control System Plus
See TACACS+
terminal lines, setting a password 7-6

ternary content addressable memory
See TCAM
TFTP
configuration files
downloading B-10
preparing the server B-10
uploading B-11
configuration files in base directory 3-6
configuring for autoconfiguration 3-6
image files
deleting B-22
downloading B-21
preparing the server B-21
uploading B-23
limiting access by servers 27-16
TFTP server 1-3
threshold, traffic level 21-2
threshold monitoring, IP SLAs 1-5
time
See NTP and system clock
time-range command 29-18
time ranges in ACLs 29-18
timestamps in log messages 26-7
time zones 6-12
Token Ring VLANs
support for 11-5
VTP support 12-4
TOS 1-6
traceroute, Layer 2
and ARP 6-15
and CDP 6-15
described 6-14
IP addresses and subnets 6-15
MAC addresses and VLANs 6-15
multicast traffic 6-15
multiple devices on a port 6-15
unicast traffic 6-14
usage guidelines 6-15
traceroute command 6-13
See also IP traceroute
traffic
blocking flooded 21-6
fragmented 29-5
unfragmented 29-5
traffic policing 1-7
traffic suppression 21-1
transparent mode, VTP 12-3, 12-12
trap-door mechanism 3-2
traps
configuring MAC address notification 6-21
configuring managers 27-11, 27-14
defined 27-3
enabling 6-21, 27-11, 27-14
notification types 27-11
overview 27-1, 27-4
troubleshooting
connectivity problems 6-11
detecting unidirectional links 23-1
determining packet disposition 6-19
displaying crash information 6-20
GBIC security and identification 6-10
PIMv1 and PIMv2 interoperability problems 3-23
PoE ports 6-16
show forward command 6-19
with CiscoWorks 27-4
with debug commands 6-16
with ping 6-11
with system message logging 26-1
with traceroute 6-13
trunking encapsulation 1-4
trunk ports
configuring 11-20
defined 9-3, 11-3
encapsulation 11-20, 11-24, 11-26

trunks
allowed-VLAN list 11-21
configuring 11-20, 11-24, 11-26
ISL 11-15
load sharing
setting STP path costs 11-25
using STP port priorities 11-23, 11-24
native VLAN for untagged traffic 11-22
parallel 11-25
pruning-eligible list 11-22
to non-DTP device 11-16
VLAN 1 minimization 11-21
trusted boundary for QoS 28-32
trustpoints, CA 7-42
tunneling
defined 14-1
IEEE 802.1Q 14-1
Layer 2 protocol 14-8
tunnel ports
defined 11-3
described 9-3, 14-1
IEEE 802.1Q, configuring 14-6
IEEE 802.1Q and ACLs 29-3
incompatibilities with other features 14-5
twisted-pair Ethernet, detecting unidirectional links 23-1
type of service
See TOS

U

UDLD
default configuration 23-4
echoing detection mechanism 23-3
enabling
globally 23-4
per interface 23-5
Layer 2 protocol tunneling 14-10
link-detection mechanism 23-1
neighbor database 23-2

UDLD (continued)
overview 23-1
resetting an interface 23-6
status, displaying 23-7
support for 1-3
UDP, configuring 31-15
UDP jitter, configuring 1-8
UDP jitter operation, IP SLAs 1-8
unauthorized ports with IEEE 802.1x 8-7
unicast MAC address filtering
and adding static addresses 6-25
and broadcast MAC addresses 6-24
and CPU packets 6-24
and multicast addresses 6-24
and router MAC addresses 6-24
configuration guidelines 6-24
described 6-24
unicast storm control
See storm control
unicast storm control command 21-4
unicast traffic, blocking 21-6
UniDirectional Link Detection protocol
See UDLD
UNIX syslog servers
daemon configuration 26-11
facilities supported 26-12
message logging configuration 26-11
unrecognized Type-Length-Value (TLV) support 12-4
upgrading software images
See downloading
upgrading with CNS 4-14
UplinkFast
described 17-3
enabling 17-17
support for 1-4

uploading
configuration files
preparing B-10, B-12, B-15
reasons for B-8
using FTP B-14
using RCP B-17
using TFTP B-11
image files
preparing B-21, B-24, B-28
reasons for B-18
using FTP B-26
using RCP B-31
using TFTP B-23
User Datagram Protocol
See UDP
user EXEC mode 2-2
username-based authentication 7-6

V

version-dependent transparent mode 12-4
virtual IP address
cluster standby group 5-11
command switch 5-11
Virtual Private Network
See VPN
virtual router 32-1, 32-3
vlan.dat file 11-4
VLAN 1 minimization, support for 1-4
VLAN ACLs
See VLAN maps
VLAN configuration
at bootup 11-7
saving 11-7
VLAN configuration mode 2-2, 11-6
VLAN database
and startup configuration file 11-7
and VTP 12-1
VLAN configuration saved in 11-7
VLANs saved in 11-4
vlan database command 11-6
vlan dot1q tag native command 14-4
vlan global configuration command 11-6
VLAN ID, discovering 6-29
VLAN management domain 12-2
VLAN Management Policy Server
See VMPS
VLAN map entries, order of 29-31
VLAN maps
applying 29-34
common uses for 29-34
configuration example 29-35
configuration guidelines 29-31
configuring 29-30
creating 29-31
defined 29-2
denying access example 29-36
denying and permitting packets 29-32
displaying 29-42
examples 29-35
support for 1-5
usage 29-4
VLAN membership
confirming 11-31
modes 11-3
VLAN Query Protocol
See VQP
VLANs
adding 11-8
adding to VLAN database 11-8
aging dynamic addresses 15-9
allowed on trunk 11-21
and spanning-tree instances 11-2, 11-6, 11-12
configuration guidelines, normal-range VLANs 11-5

VLANs (continued)
configuration options 11-6
configuring 11-1
configuring IDs 1006 to 4094 11-12
connecting through SVIs 9-8
creating in config-vlan mode 11-8
creating in VLAN configuration mode 11-9
customer numbering in service-provider networks 14-3
default configuration 11-7
deleting 11-10
described 9-2, 11-1
displaying 11-14
extended-range 11-1, 11-11
features 1-4
illustrated 11-2
internal 11-12
limiting source traffic with RSPAN 24-23
limiting source traffic with SPAN 24-15
modifying 11-8
monitoring with RSPAN 24-21
monitoring with SPAN 24-14
native, configuring 11-22
normal-range 11-1, 11-4
number supported 1-4
parameters 11-4
port membership modes 11-3
static-access ports 11-10
STP and IEEE 802.1Q trunks 15-10
supported 11-2
Token Ring 11-5
traffic between 11-2
trunks, VLAN 1 minimization 11-21
VLAN-bridge STP 15-10, 5-1
VTP modes 12-2
VLAN Trunking Protocol
See VTP
VLAN trunks 11-15
VMPS
administering 11-32
configuration example 11-33
configuration guidelines 11-28
default configuration 11-28
description 11-26
dynamic port membership
described 11-27
reconfirming 11-31
troubleshooting 11-32
entering server address 11-29
mapping MAC addresses to VLANs 11-27
monitoring 11-32
reconfirmation interval, changing 11-31
reconfirming membership 11-31
retry count, changing 11-31
voice VLAN
Cisco 7960 phone, port connections 13-1
configuration guidelines 13-3
configuring IP phones for data traffic
override CoS of incoming frame 13-5
trust CoS priority of incoming frame 13-5
configuring ports for voice traffic in
802.1p priority tagged frames 13-4
802.1Q frames 13-4
connecting to an IP phone 13-3
default configuration 13-2
described 13-1
displaying 13-6
VPN
configuring routing in 31-66
forwarding 31-64
in service provider networks 31-61
routes 31-62
VPN routing and forwarding table
See VRF
VQP 1-4, 11-26

VRF
defining 31-64
tables 31-61
VTP
adding a client to a domain 12-14
advertisements 11-18, 12-3
and extended-range VLANs 12-1
and normal-range VLANs 12-1
client mode, configuring 12-10
configuration
global configuration mode 12-7
guidelines 12-8
privileged EXEC mode 12-7
requirements 12-9
saving 12-7
VLAN configuration mode 12-7
configuration mode options 12-7
configuration requirements 12-9
configuration revision number
guideline 12-14
resetting 12-15
configuring
client mode 12-10
server mode 12-9
transparent mode 12-12
consistency checks 12-4
default configuration 12-6
described 12-1
disabling 12-12
domain names 12-8
domains 12-2
Layer 2 protocol tunneling 14-7
modes
client 12-3, 12-10
server 12-3, 12-9
transitions 12-2
transparent 12-3, 12-12
monitoring 12-15
passwords 12-8

VTP (continued)
pruning
disabling 12-14
enabling 12-13
examples 12-5
overview 12-4
support for 1-4
pruning-eligible list, changing 11-22
server mode, configuring 12-9
statistics 12-15
support for 1-4
Token Ring support 12-4
transparent mode, configuring 12-12
using 12-1
version, guidelines 12-8
version 1 12-4
version 2
configuration guidelines 12-8
disabling 12-13
enabling 12-12
overview 12-4

W

WCCP
authentication 2-3
configuration guidelines 2-4
default configuration 2-4
described 2-1
displaying 2-8
enabling 2-5
features unsupported 2-4
forwarding method 2-3
Layer-2 header rewrite 2-3
MD5 security 2-3
message exchange 2-2
monitoring and maintaining 2-8
negotiation 2-3
packet redirection 2-3

WCCP (continued)

packet-return method 2-3

redirecting traffic received from a client 2-5

setting the password 2-5

unsupported WCCPv2 features 2-4

web authentication

configuring 8-39 to 8-41

described 1-5

fallback for IEEE 802.1x 8-40

Web Cache Communication Protocol

See WCCP

Weighted Random Early Detection

See WRED

Weighted Round Robin

See WRR

weighted round robin, described 28-4

wizards 1-10

WRED 1-7, 28-13

WRR 1-7, 28-4

X

Xmodem protocol 6-2