

IBM Virtual Fabric 10Gb Switch Module for IBM BladeCenter®



# Application Guide



IBM Virtual Fabric 10Gb Switch Module for IBM BladeCenter®



# Application Guide

**Note:** Before using this information and the product it supports, read the general information in the *Safety information and Environmental Notices and User Guide* documents on the IBM *Documentation* CD and the *Warranty Information* document that comes with the product.

**First Edition (December 2013)**

**© Copyright IBM Corporation 2013**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

---

# Contents

<b>Preface</b> . . . . .	<b>19</b>
Who Should Use This Guide . . . . .	19
What You'll Find in This Guide . . . . .	19
Additional References. . . . .	21
Typographic Conventions . . . . .	22
How to Get Help . . . . .	22
 <b>Part 1:. Getting Started.</b> . . . . .	 <b>25</b>
 <b>Chapter 1. Switch Administration</b> . . . . .	 <b>27</b>
Administration Interfaces . . . . .	28
Management Module . . . . .	28
Command Line Interface . . . . .	28
Browser-Based Interface . . . . .	29
Establishing a Connection . . . . .	30
Using the BladeCenter Management Module . . . . .	30
Factory-Default vs. MM-Assigned IP Addresses . . . . .	30
Default Gateway . . . . .	31
Configuring the Management Module for Switch Access . . . . .	31
Using an External Switch Port . . . . .	32
Using Telnet. . . . .	33
Using Secure Shell . . . . .	33
Using a Web Browser . . . . .	34
Configuring HTTP Access to the BBI. . . . .	34
Configuring HTTPS Access to the BBI . . . . .	35
Browser-Based Interface Summary . . . . .	37
Using Simple Network Management Protocol . . . . .	39
BOOTP/DHCP Client IP Address Services. . . . .	40
Global BOOTP Relay Agent Configuration . . . . .	41
Domain-Specific BOOTP Relay Agent Configuration . . . . .	41
DHCP Option 82 . . . . .	41
DHCP Snooping . . . . .	42
Switch Login Levels . . . . .	43
Setup vs. the Command Line . . . . .	45
Boot Strict Mode . . . . .	46
Acceptable Cipher Suites. . . . .	48
Configuring Strict Mode . . . . .	49
Limitations . . . . .	49
 <b>Chapter 2. Initial Setup.</b> . . . . .	 <b>51</b>
Information Needed for Setup. . . . .	52
Default Setup Options. . . . .	53
Stopping and Restarting Setup Manually . . . . .	54
Setup Part 1: Basic System Configuration . . . . .	55
Setup Part 2: Port Configuration. . . . .	57
Setup Part 3: VLANs . . . . .	59

Setup Part 4: IP Configuration . . . . .	60
IP Interfaces . . . . .	60
Loopback Interfaces . . . . .	61
Using Loopback Interfaces for Source IP Addresses . . . . .	61
Loopback Interface Limitation . . . . .	62
Default Gateways . . . . .	62
IP Routing . . . . .	63
Setup Part 5: Final Steps . . . . .	64
Optional Setup for Telnet Support . . . . .	65
<b>Part 2:. Securing the Switch . . . . .</b>	<b>67</b>
<b>Chapter 3. Securing Administration . . . . .</b>	<b>69</b>
Changing the Switch Passwords . . . . .	70
Changing the Default Administrator Password . . . . .	70
Changing the Default User Password . . . . .	70
Secure Shell and Secure Copy . . . . .	71
Configuring SSH/SCP Features on the Switch . . . . .	71
Configuring the SCP Administrator Password. . . . .	72
Using SSH and SCP Client Commands . . . . .	72
SSH and SCP Encryption of Management Messages . . . . .	74
Generating RSA Host Key for SSH Access . . . . .	75
SSH/SCP Integration with RADIUS Authentication. . . . .	75
SSH/SCP Integration with LDAP Authentication . . . . .	75
SSH/SCP Integration with TACACS+ Authentication . . . . .	75
End User Access Control . . . . .	76
Considerations for Configuring End User Accounts . . . . .	76
Strong Passwords . . . . .	76
User Access Control Menu . . . . .	77
Listing Current Users . . . . .	79
Logging In to an End User Account. . . . .	79
Protected Mode . . . . .	80
<b>Chapter 4. Authentication &amp; Authorization Protocols . . . . .</b>	<b>81</b>
RADIUS Authentication and Authorization. . . . .	82
How RADIUS Authentication Works . . . . .	82
Configuring RADIUS on the Switch. . . . .	82
RADIUS Authentication Features in IBM N/OS . . . . .	83
Switch User Accounts . . . . .	84
RADIUS Attributes for IBM N/OS User Privileges . . . . .	84
TACACS+ Authentication . . . . .	86
How TACACS+ Authentication Works. . . . .	86
TACACS+ Authentication Features in IBM N/OS . . . . .	86
Authorization. . . . .	86
Accounting . . . . .	87
Command Authorization and Logging. . . . .	88
TACACS+ Password Change . . . . .	89
Configuring TACACS+ Authentication on the Switch . . . . .	89
LDAP Authentication and Authorization. . . . .	90
<b>Chapter 5. 802.1X Port-Based Network Access Control . . . . .</b>	<b>93</b>
Extensible Authentication Protocol over LAN . . . . .	94
EAPoL Authentication Process . . . . .	95

EAPoL Message Exchange . . . . .	96
EAPoL Port States . . . . .	97
Guest VLAN . . . . .	98
Supported RADIUS Attributes . . . . .	99
EAPoL Configuration Guidelines . . . . .	101
<b>Chapter 6. Access Control Lists . . . . .</b>	<b>103</b>
Summary of Packet Classifiers . . . . .	104
Summary of ACL Actions . . . . .	106
Assigning Individual ACLs to a Port . . . . .	107
ACL Order of Precedence . . . . .	108
ACL Groups . . . . .	109
Assigning ACL Groups to a Port. . . . .	110
ACL Metering and Re-Marking . . . . .	111
ACL Port Mirroring . . . . .	112
Viewing ACL Statistics . . . . .	113
ACL Configuration Examples . . . . .	114
VLAN Maps. . . . .	117
Management ACLs. . . . .	119
<b>Part 3:. Switch Basics . . . . .</b>	<b>121</b>
<b>Chapter 7. VLANs . . . . .</b>	<b>123</b>
VLANs Overview. . . . .	124
VLANs and Port VLAN ID Numbers . . . . .	125
VLAN Tagging/Trunk Mode . . . . .	127
Ingress VLAN Tagging/Trunk Mode. . . . .	130
Limitations . . . . .	131
VLAN Topologies and Design Considerations . . . . .	132
Protocol-Based VLANs . . . . .	135
Port-Based vs. Protocol-Based VLANs . . . . .	136
PVLAN Priority Levels . . . . .	136
PVLAN Tagging/Trunk Mode . . . . .	136
PVLAN Configuration Guidelines. . . . .	137
Configuring PVLAN . . . . .	137
Private VLANs . . . . .	139
Private VLAN Ports . . . . .	140
Configuration Guidelines . . . . .	140
Configuration Example. . . . .	140
<b>Chapter 8. Ports and Trunking . . . . .</b>	<b>143</b>
Trunking Overview . . . . .	144
Static Trunks . . . . .	145
Static Trunk Requirements . . . . .	145
Inter-Switch Link . . . . .	145
Static Trunk Group Configuration Rules . . . . .	147
Configuring a Static Port Trunk . . . . .	148
Link Aggregation Control Protocol . . . . .	150
LACP Overview . . . . .	150
Static LACP . . . . .	152
LACP Minimum Links Option . . . . .	153
Configuring LACP. . . . .	153
Configurable Trunk Hash Algorithm . . . . .	154

<b>Chapter 9. Spanning Tree Protocols</b>	<b>157</b>
Spanning Tree Protocol Modes	158
Global STP Control	159
PVRST Mode	160
Port States	160
Bridge Protocol Data Units	161
Bridge Protocol Data Units Overview	161
Determining the Path for Forwarding BPDUs	161
Simple STP Configuration	164
Per-VLAN Spanning Tree Groups	166
Using Multiple STGs to Eliminate False Loops	166
VLAN and STG Assignment	166
Manually Assigning STGs	167
Guidelines for Creating VLANs	168
Rules for VLAN Tagged/Trunk Mode Ports	168
Adding and Removing Ports from STGs	169
Switch-Centric Configuration	170
Configuring Multiple STGs	171
Rapid Spanning Tree Protocol	173
Port States	173
RSTP Configuration Guidelines	173
RSTP Configuration Example	173
Multiple Spanning Tree Protocol	174
MSTP Region	174
Common Internal Spanning Tree	174
MSTP Configuration Guidelines	174
MSTP Configuration Examples	175
Port Type and Link Type	178
Edge Port	178
Link Type	178
<b>Chapter 10. Quality of Service</b>	<b>179</b>
QoS Overview	180
Using ACL Filters	182
Summary of ACL Actions	182
ACL Metering and Re-Marking	182
Metering	183
Using DSCP Values to Provide QoS	184
Differentiated Services Concepts	184
Trusted/Untrusted Ports	184
Per-Hop Behavior	185
QoS Levels	186
DSCP Re-Marking and Mapping	186
DSCP Re-Marking Overview	186
DSCP Re-Marking Configuration Examples	187
Using 802.1p Priorities to Provide QoS	189
Queuing and Scheduling	190
Control Plane Protection	191

<b>Part 4:. Advanced Switching Features</b>	<b>.193</b>
<b>Chapter 11. Stacking</b>	<b>.195</b>
Stacking Overview	.196
Stacking Requirements	.196
Stacking Limitations	.197
Stack Membership	.198
The Master Switch	.198
Splitting and Merging One Stack	.198
Merging Independent Stacks	.199
Backup Switch Selection	.199
Master Failover	.200
Secondary Backup	.200
Master Recovery	.200
No Backup	.200
Stack Member Identification	.201
Configuring a Stack	.202
Configuration Overview	.202
Best Configuration Practices	.202
Stacking VLANs	.203
Configuring Each Switch in a Stack	.203
Additional Master Configuration	.204
Configuring an External IPv4 Address for the Stack	.204
Locating an External Stack Interface	.205
Viewing Stack Connections	.205
Binding Members to the Stack	.206
Assigning a Stack Backup Switch	.206
Managing a Stack	.207
Upgrading Software in an Existing Stack	.209
Replacing or Removing Stacked Switches	.211
Removing a Switch from the Stack	.211
Installing the New Switch or Healing the Topology	.211
Binding the New Switch to the Stack	.212
ISCLI Stacking Commands	.214
<b>Chapter 12. Virtualization.</b>	<b>.215</b>
<b>Chapter 13. Virtual NICs</b>	<b>.217</b>
vNIC IDs	.219
vNIC IDs on the Switch	.219
vNIC Interface Names on the Server	.220
vNIC Bandwidth Metering	.221
vNIC Uplink Modes	.222
LACP Trunks	.224
vNIC Groups	.225
vNIC Teaming Failover	.228
vNIC Configuration Example	.230
vNICs for iSCSI on Emulex Virtual Fabric Adapter	.232
vNICs for FCoE on Emulex Virtual Fabric Adapter	.233
vNIC Configuration Example - Shared Mode with LACP Trunk	.235
<b>Chapter 14. VMready</b>	<b>.237</b>
VE Capacity	.238

VM Group Types . . . . .	239
Local VM Groups . . . . .	240
Distributed VM Groups . . . . .	242
VM Profiles . . . . .	242
Initializing a Distributed VM Group . . . . .	242
Assigning Members . . . . .	243
Synchronizing the Configuration . . . . .	243
Removing Member VEs . . . . .	244
VMcheck . . . . .	245
Virtual Distributed Switch . . . . .	247
Prerequisites . . . . .	247
Guidelines . . . . .	247
Migrating to vDS . . . . .	247
Virtualization Management Servers . . . . .	249
Assigning a vCenter . . . . .	249
vCenter Scans . . . . .	249
Deleting the vCenter . . . . .	250
Exporting Profiles . . . . .	251
VMware Operational Commands . . . . .	251
Pre-Provisioning VEs . . . . .	252
VLAN Maps . . . . .	253
VM Policy Bandwidth Control . . . . .	254
VM Policy Bandwidth Control Commands . . . . .	254
Bandwidth Policies vs. Bandwidth Shaping . . . . .	255
VMready Information Displays . . . . .	256
VMready Configuration Example . . . . .	260
<b>Chapter 15. FCoE and CEE . . . . .</b>	<b>261</b>
Fibre Channel over Ethernet . . . . .	262
The FCoE Topology . . . . .	262
FCoE Requirements . . . . .	264
Port Trunking . . . . .	264
Connecting to a Bridge Module . . . . .	264
Bridge Module Location . . . . .	264
Defining the Bridge Connection . . . . .	266
Converged Enhanced Ethernet . . . . .	268
Turning CEE On or Off . . . . .	268
Effects on Link Layer Discovery Protocol . . . . .	268
Effects on 802.1p Quality of Service . . . . .	269
Effects on Flow Control . . . . .	269
FCoE Initialization Protocol Snooping . . . . .	271
Global FIP Snooping Settings . . . . .	271
FIP Snooping for Specific Ports . . . . .	271
Port FCF and ENode Detection . . . . .	272
FCoE Connection Timeout . . . . .	272
FCoE ACL Rules . . . . .	273
FCoE VLANs . . . . .	273
Viewing FIP Snooping Information . . . . .	274
Operational Commands . . . . .	274
FIP Snooping Configuration . . . . .	275
Priority-Based Flow Control . . . . .	277
Global vs. Port-by-Port Configuration . . . . .	278
PFC Configuration Example . . . . .	279

Enhanced Transmission Selection . . . . .	281
802.1p Priority Values . . . . .	281
Priority Groups . . . . .	282
PGID . . . . .	282
Assigning Priority Values to a Priority Group. . . . .	283
Deleting a Priority Group . . . . .	283
Allocating Bandwidth . . . . .	283
Configuring ETS . . . . .	284
Data Center Bridging Capability Exchange . . . . .	287
DCBX Settings . . . . .	287
Configuring DCBX . . . . .	289
<b>Chapter 16. Edge Virtual Bridging . . . . .</b>	<b>291</b>
EVB Operations Overview . . . . .	292
VSIDB Synchronization . . . . .	292
VLAN Behavior. . . . .	293
Manual Reflective Relay . . . . .	293
EVB Configuration . . . . .	294
Configuring EVB in Stacking Mode. . . . .	296
Limitations . . . . .	297
Unsupported features . . . . .	297
<b>Chapter 17. Static Multicast ARP . . . . .</b>	<b>299</b>
Configuring Static Multicast ARP . . . . .	300
Configuration Example. . . . .	300
Limitations . . . . .	301
<b>Chapter 18. Unified Fabric Port . . . . .</b>	<b>303</b>
UFP Limitations . . . . .	304
Virtual Ports Modes . . . . .	305
vPort-S-Tag Mapping. . . . .	305
vPort-VLAN Mapping. . . . .	305
UFP vPort Mode . . . . .	305
Tunnel Mode . . . . .	305
UFP Bandwidth Provisioning . . . . .	308
UFP Strict Bandwidth Provisioning Mode. . . . .	308
Using UFP with Other Virtual Fabric Switch Module Features . . . . .	309
Layer 2 Failover . . . . .	309
Increased VLAN Limits. . . . .	309
UFP Configuration Examples . . . . .	310
Example 1: Access Mode. . . . .	310
Example 2: Trunk Mode . . . . .	311
Example 3: Tunnel Mode . . . . .	312
Example 4: FCoE Mode . . . . .	313
<b>Part 5:. IP Routing. . . . .</b>	<b>315</b>
<b>Chapter 19. Basic IP Routing . . . . .</b>	<b>317</b>
IP Routing Benefits . . . . .	318
Routing Between IP Subnets . . . . .	319
Subnet Routing Example . . . . .	321
Using VLANs to Segregate Broadcast Domains . . . . .	322

BOOTP Relay Agent . . . . .	324
BOOTP Relay Agent Overview . . . . .	324
BOOTP Relay Agent Configuration . . . . .	324
Dynamic Host Configuration Protocol . . . . .	325
DHCP Relay Agent . . . . .	326
DHCP Relay Agent Configuration . . . . .	327
<b>Chapter 20. Internet Protocol Version 6 . . . . .</b>	<b>329</b>
IPv6 Limitations . . . . .	330
IPv6 Address Format . . . . .	331
IPv6 Address Types . . . . .	332
IPv6 Address Autoconfiguration . . . . .	333
IPv6 Interfaces . . . . .	334
Neighbor Discovery . . . . .	335
Supported Applications . . . . .	337
Configuration Guidelines . . . . .	339
IPv6 Configuration Examples . . . . .	340
<b>Chapter 21. Using IPsec with IPv6 . . . . .</b>	<b>341</b>
IPsec Protocols . . . . .	342
Using IPsec with the VFSM . . . . .	343
Setting up Authentication . . . . .	344
Creating an IKEv2 Proposal . . . . .	344
Importing an IKEv2 Digital Certificate . . . . .	345
Generating an IKEv2 Digital Certificate . . . . .	345
Enabling IKEv2 Preshared Key Authentication . . . . .	346
Setting Up a Key Policy . . . . .	346
Using a Manual Key Policy . . . . .	348
Using a Dynamic Key Policy . . . . .	350
<b>Chapter 22. Routing Information Protocol . . . . .</b>	<b>351</b>
Distance Vector Protocol . . . . .	352
Stability . . . . .	353
Routing Updates . . . . .	354
RIPv1 . . . . .	355
RIPv2 . . . . .	356
RIPv2 in RIPv1 Compatibility Mode . . . . .	357
RIP Features . . . . .	358
RIP Configuration Example . . . . .	360
<b>Chapter 23. Internet Group Management Protocol . . . . .</b>	<b>363</b>
IGMP Terms . . . . .	364
How IGMP Works . . . . .	365
VFSM Capacity and IGMP Default Values . . . . .	366

IGMP Snooping . . . . .	.367
IGMP Groups . . . . .	.367
IGMPv3 . . . . .	.367
IGMP Snooping Configuration Guidelines . . . . .	.368
IGMP Snooping Configuration Example . . . . .	.369
Advanced Configuration Example: IGMP Snooping. . . . .	.370
Prerequisites . . . . .	.371
Configuration . . . . .	.371
Switch B Configuration . . . . .	.372
Switch C Configuration . . . . .	.373
Troubleshooting . . . . .	.375
IGMP Relay . . . . .	.378
Configuration Guidelines . . . . .	.378
Configure IGMP Relay . . . . .	.379
Advanced Configuration Example: IGMP Relay . . . . .	.380
Prerequisites . . . . .	.380
Configuration . . . . .	.381
Troubleshooting . . . . .	.384
Additional IGMP Features . . . . .	.387
FastLeave. . . . .	.387
IGMP Filtering . . . . .	.387
Static Multicast Router . . . . .	.388
 <b>Chapter 24. Multicast Listener Discovery . . . . .</b>	 <b>.389</b>
MLD Terms . . . . .	.390
How MLD Works. . . . .	.391
MLD Querier. . . . .	.393
Dynamic M routers . . . . .	.393
MLD Capacity and Default Values . . . . .	.394
Configuring MLD. . . . .	.395
 <b>Chapter 25. Border Gateway Protocol . . . . .</b>	 <b>.397</b>
Internal Routing Versus External Routing . . . . .	.398
Forming BGP Peer Routers . . . . .	.399
Loopback Interfaces . . . . .	.400
What is a Route Map? . . . . .	.401
Incoming and Outgoing Route Maps . . . . .	.401
Precedence . . . . .	.402
Configuration Overview . . . . .	.402
Aggregating Routes . . . . .	.404
Redistributing Routes . . . . .	.405
BGP Attributes . . . . .	.406
Selecting Route Paths in BGP . . . . .	.407
BGP Failover Configuration . . . . .	.408
Default Redistribution and Route Aggregation Example . . . . .	.410

<b>Chapter 26. OSPF.</b>	<b>413</b>
OSPFv2 Overview	414
Types of OSPF Areas	414
Types of OSPF Routing Devices	415
Neighbors and Adjacencies	416
The Link-State Database	416
The Shortest Path First Tree	417
Internal Versus External Routing	417
OSPFv2 Implementation in IBM N/OS	418
Configurable Parameters	418
Defining Areas	419
Assigning the Area Index	419
Using the Area ID to Assign the OSPF Area Number	420
Attaching an Area to a Network	420
Interface Cost	421
Electing the Designated Router and Backup	421
Summarizing Routes	421
Default Routes	422
Virtual Links	422
Router ID	423
Authentication	423
Configuring Plain Text OSPF Passwords	424
Configuring MD5 Authentication	425
Host Routes for Load Balancing	425
Loopback Interfaces in OSPF	426
OSPF Features Not Supported in This Release	426
OSPFv2 Configuration Examples	427
Example 1: Simple OSPF Domain	428
Example 2: Virtual Links	430
Example 3: Summarizing Routes	433
Verifying the OSPF Configuration	434
OSPFv3 Implementation in IBM N/OS	435
OSPFv3 Differences from OSPFv2	435
OSPFv3 Requires IPv6 Interfaces	435
OSPFv3 Uses Independent Command Paths	435
OSPFv3 Identifies Neighbors by Router ID	435
Other Internal Improvements	436
OSPFv3 Limitations	436
OSPFv3 Configuration Example	436
Neighbor Configuration Example	438
 <b>Part 6:. High Availability Fundamentals</b>	 <b>441</b>
 <b>Chapter 27. Basic Redundancy</b>	 <b>443</b>
Trunking for Link Redundancy	444
Hot Links	445
Forward Delay	445
Preemption	445
FDB Update	445
Configuration Guidelines	445
Configuring Hot Links	446
Stacking for High Availability Topologies	447

<b>Chapter 28. Layer 2 Failover</b>	<b>.449</b>
Auto Monitoring Trunk Links	.450
VLAN Monitor	.450
Auto Monitor Configurations	.450
Setting the Failover Limit	.452
Manually Monitoring Port Links	.453
L2 Failover with Other Features	.454
LACP	.454
Spanning Tree Protocol	.454
Configuration Guidelines	.455
Auto Monitor Guidelines	.455
Manual Monitor Guidelines	.455
Configuring Layer 2 Failover	.456
Auto Monitor Example	.456
Manual Monitor Example	.456
 <b>Chapter 29. Virtual Router Redundancy Protocol</b>	 <b>.457</b>
VRRP Overview	.458
VRRP Components	.458
VRRP Operation	.459
Selecting the Master VRRP Router	.460
Failover Methods	.461
Active-Active Redundancy	.462
Hot-Standby Redundancy	.462
Virtual Router Group	.462
IBM N/OS Extensions to VRRP	.464
Virtual Router Deployment Considerations	.465
High Availability Configurations	.466
Active-Active Configuration	.466
Hot-Standby Configuration	.470
 <b>Part 7: Network Management</b>	 <b>.473</b>
 <b>Chapter 30. Link Layer Discovery Protocol</b>	 <b>.475</b>
LLDP Overview	.476
Enabling or Disabling LLDP	.477
Global LLDP Setting	.477
Transmit and Receive Control	.477
LLDP Transmit Features	.478
Scheduled Interval	.478
Minimum Interval	.478
Time-to-Live for Transmitted Information	.479
Trap Notifications	.479
Changing the LLDP Transmit State	.480
Types of Information Transmitted	.480
LLDP Receive Features	.482
Types of Information Received	.482
Viewing Remote Device Information	.482
Time-to-Live for Received Information	.484
LLDP Example Configuration	.486
 <b>Chapter 31. Simple Network Management Protocol</b>	 <b>.487</b>
SNMP Version 1 & Version 2	.488

SNMP Version 3 . . . . .	489
Configuring SNMP Trap Hosts . . . . .	493
SNMP MIBs . . . . .	496
Switch Images and Configuration Files . . . . .	501
Loading a New Switch Image . . . . .	502
Loading a Saved Switch Configuration . . . . .	502
Saving the Switch Configuration . . . . .	502
Saving a Switch Dump . . . . .	503
<b>Part 8:. Monitoring . . . . .</b>	<b>505</b>
<b>Chapter 32. Remote Monitoring . . . . .</b>	<b>507</b>
RMON Overview . . . . .	508
RMON Group 1—Statistics . . . . .	509
RMON Group 2—History . . . . .	510
History MIB Objects. . . . .	510
Configuring RMON History . . . . .	510
RMON Group 3—Alarms . . . . .	512
Alarm MIB Objects . . . . .	512
Configuring RMON Alarms . . . . .	512
RMON Group 9—Events . . . . .	514
<b>Chapter 33. sFLOW . . . . .</b>	<b>515</b>
sFlow Statistical Counters . . . . .	516
sFlow Network Sampling . . . . .	517
sFlow Example Configuration . . . . .	518
<b>Chapter 34. Port Mirroring . . . . .</b>	<b>519</b>
Port Mirroring Behavior . . . . .	520
<b>Part 9:. Appendices . . . . .</b>	<b>521</b>
<b>Appendix A. Glossary . . . . .</b>	<b>523</b>
<b>Appendix B. RADIUS Server Configuration Notes . . . . .</b>	<b>525</b>
<b>Appendix C. Getting help and technical assistance. . . . .</b>	<b>527</b>
Before you call . . . . .	527
Using the documentation . . . . .	527
Getting help and information on the World Wide Web . . . . .	527
Software service and support . . . . .	528
Hardware service and support . . . . .	528
IBM Taiwan product service . . . . .	528
<b>Appendix D. Notices . . . . .</b>	<b>529</b>
Trademarks . . . . .	529
Important Notes . . . . .	530
Particulate contamination . . . . .	531
Documentation format . . . . .	531

Electronic emission notices . . . . .	.532
Federal Communications Commission (FCC) statement . . . . .	.532
Industry Canada Class A emission compliance statement . . . . .	.532
Avis de conformité à la réglementation d'Industrie Canada . . . . .	.532
Australia and New Zealand Class A statement . . . . .	.532
European Union EMC Directive conformance statement. . . . .	.532
Germany Class A statement . . . . .	.533
Japan VCCI Class A statement . . . . .	.534
Korea Communications Commission (KCC) statement . . . . .	.534
Russia Electromagnetic Interference (EMI) Class A statement . . . . .	.534
People's Republic of China Class A electronic emission statement . . . . .	.534
Taiwan Class A compliance statement . . . . .	.535
<b>Index . . . . .</b>	<b>.537</b>



---

## Preface

The *IBM N/OS 7.8 Application Guide* describes how to configure and use the IBM Networking OS 7.8 software on the IBM Virtual Fabric 10Gb Switch Module for IBM BladeCenter (referred to as VFSM throughout this document). For documentation on installing the switch physically, see the *Installation Guide* for your VFSM.

---

## Who Should Use This Guide

This guide is intended for network installers and system administrators engaged in configuring and maintaining a network. The administrator should be familiar with Ethernet concepts, IP addressing, Spanning Tree Protocol, and SNMP configuration parameters.

---

## What You'll Find in This Guide

This guide will help you plan, implement, and administer IBM N/OS software. Where possible, each section provides feature overviews, usage examples, and configuration instructions. The following material is included:

### Part 1: Getting Started

This material is intended to help those new to N/OS products with the basics of switch management. This part includes the following chapters:

- [Chapter 1, "Switch Administration,"](#) describes how to access the VFSM in order to configure the switch and view switch information and statistics. This chapter discusses a variety of manual administration interfaces, including local management via the switch console, and remote administration via Telnet, a web browser, or via SNMP.
- [Chapter 2, "Initial Setup,"](#) describes how to use the built-in Setup utility to perform first-time configuration of the switch.

### Part 2: Securing the Switch

- [Chapter 3, "Securing Administration,"](#) describes methods for changing the default switch passwords, using Secure Shell and Secure Copy for administration connections, configuring end-user access control, and placing the switch in protected mode.
- [Chapter 4, "Authentication & Authorization Protocols,"](#) describes different secure administration for remote administrators. This includes using Remote Authentication Dial-in User Service (RADIUS), as well as TACACS+ and LDAP.
- [Chapter 5, "802.1X Port-Based Network Access Control,"](#) describes how to authenticate devices attached to a LAN port that has point-to-point connection characteristics. This feature prevents access to ports that fail authentication and authorization and provides security to ports of the VFSM that connect to blade servers.
- [Chapter 6, "Access Control Lists,"](#) describes how to use filters to permit or deny specific types of traffic, based on a variety of source, destination, and packet attributes.

### Part 3: Switch Basics

- [Chapter 7, “VLANs,”](#) describes how to configure Virtual Local Area Networks (VLANs) for creating separate network segments, including how to use VLAN tagging for devices that use multiple VLANs. This chapter also describes Protocol-based VLANs, and Private VLANs.
- [Chapter 8, “Ports and Trunking,”](#) describes how to group multiple physical ports together to aggregate the bandwidth between large-scale network devices.
- [Chapter 9, “Spanning Tree Protocols,”](#) discusses how Spanning Tree Protocol (STP) configures the network so that the switch selects the most efficient path when multiple paths exist. Covers Rapid Spanning Tree Protocol (RSTP), Per-VLAN Rapid Spanning Tree (PVRST), and Multiple Spanning Tree Protocol (MSTP).
- [Chapter 10, “Quality of Service,”](#) discusses Quality of Service (QoS) features, including IP filtering using Access Control Lists (ACLs), Differentiated Services, and IEEE 802.1p priority values.

### Part 4: Advanced Switching Features

- [Chapter 11, “Stacking,”](#) describes how to combine multiple switches into a single, aggregate switch entity.
- [Chapter 12, “Virtualization,”](#) provides an overview of allocating resources based on the logical needs of the data center, rather than on the strict, physical nature of components.
- [Chapter 13, “Virtual NICs,”](#) discusses using virtual NIC (vNIC) technology to divide NICs into multiple logical, independent instances.
- [Chapter 14, “VMready,”](#) discusses virtual machine (VM) support on the VFSM.
- [Chapter 15, “FCoE and CEE,”](#) discusses using various Converged Enhanced Ethernet (CEE) features such as Priority-based Flow Control (PFC), Enhanced Transmission Selection (ETS), and FIP Snooping for solutions such as Fibre Channel over Ethernet (FCoE).
- [Chapter 16, “Edge Virtual Bridging \(EVB\)”](#) discusses the IEEE 802.1Qbg—a standards-based protocol that defines how virtual Ethernet bridges exchange configuration information. EVB bridges the gap between physical and virtual network resources, thus simplifying network management.
- [Chapter 17, “Static Multicast ARP”](#) discusses the configuration of a static ARP entry with multicast MAC address for Microsoft’s Network Load Balancing (NLB) feature to function efficiently.
- [Chapter 18, “Unified Fabric Port”](#) describes how UFP logically subdivides a high-speed physical link connecting to a server NIC or to a Converged Network Adapter (CNA). UFP provides a switch fabric component to control the NIC.

### Part 5: IP Routing

- [Chapter 19, “Basic IP Routing,”](#) describes how to configure the VFSM for IP routing using IP subnets, BOOTP, and DHCP Relay.
- [Chapter 20, “Internet Protocol Version 6,”](#) describes how to configure the VFSM for IPv6 host management.
- [Chapter 21, “Using IPsec with IPv6,”](#) describes how to configure Internet Protocol Security (IPsec) for securing IP communications by authenticating and encrypting IP packets, with emphasis on Internet Key Exchange version 2, and authentication/confidentiality for OSPFv3.
- [Chapter 22, “Routing Information Protocol,”](#) describes how the N/OS software implements standard Routing Information Protocol (RIP) for exchanging TCP/IP route information with other routers.

- [Chapter 23, “Internet Group Management Protocol,”](#) describes how the N/OS software implements IGMP Snooping or IGMP Relay to conserve bandwidth in a multicast-switching environment.
- [Chapter 24, “Multicast Listener Discovery,”](#) describes how Multicast Listener Discovery (MLD) is used with IPv6 to support host users requests for multicast data for a multicast group.
- [Chapter 25, “Border Gateway Protocol,”](#) describes Border Gateway Protocol (BGP) concepts and features supported in N/OS.
- [Chapter 26, “OSPF,”](#) describes key Open Shortest Path First (OSPF) concepts and their implemented in N/OS, and provides examples of how to configure your switch for OSPF support.

## **Part 6: High Availability Fundamentals**

- [Chapter 27, “Basic Redundancy,”](#) describes how the VFSM supports redundancy through stacking, trunking, Active Multipass Protocol (AMP), and hotlinks.
- [Chapter 28, “Layer 2 Failover,”](#) describes how the VFSM supports high-availability network topologies using Layer 2 Failover.
- [Chapter 29, “Virtual Router Redundancy Protocol,”](#) describes how the VFSM supports high-availability network topologies using Virtual Router Redundancy Protocol (VRRP).

## **Part 7: Network Management**

- [Chapter 30, “Link Layer Discovery Protocol,”](#) describes how Link Layer Discovery Protocol helps neighboring network devices learn about each others’ ports and capabilities.
- [Chapter 31, “Simple Network Management Protocol,”](#) describes how to configure the switch for management through an SNMP client.

## **Part 8: Monitoring**

- [Chapter 32, “Remote Monitoring,”](#) describes how to configure the RMON agent on the switch, so that the switch can exchange network monitoring data.
- [Chapter 33, “sFLOW,”](#) described how to use the embedded sFlow agent for sampling network traffic and providing continuous monitoring information to a central sFlow analyzer.
- [Chapter 34, “Port Mirroring,”](#) discusses tools how copy selected port traffic to a monitor port for network analysis.

## **Part 9: Appendices**

- [Appendix A, “Glossary,”](#) describes common terms and concepts used throughout this guide.
- [Appendix B, “RADIUS Server Configuration Notes,”](#) discusses how to modify RADIUS configuration files for the Nortel Networks BaySecure Access Control RADIUS server, to provide authentication for users of the VFSM.

---

## **Additional References**

Additional information about installing and configuring the VFSM is available in the following guides:

- *IBM Virtual Fabric 10Gb Switch Module Installation Guide*
- *IBM Networking OS 7.8 ISCLI Reference Guide*
- *IBM Networking OS 7.8 BBI Quick Guide*

---

## Typographic Conventions

The following table describes the typographic styles used in this book.

Table 1. *Typographic Conventions*

Typeface or Symbol	Meaning	Example
ABC123	This type is used for names of commands, files, and directories used within the text.  It also depicts on-screen computer output and prompts.	View the <code>readme.txt</code> file.  Main#
<b>ABC123</b>	This bold type appears in command examples. It shows text that must be typed in exactly as shown.	Main# <b>sys</b>
<ABC123>	This italicized type appears in command examples as a parameter placeholder. Replace the indicated text with the appropriate real name or value when using the command. Do not type the brackets.  This also shows book titles, special terms, or words to be emphasized.	To establish a Telnet session, enter: host# <b>telnet</b> <IP address>  Read your <i>User's Guide</i> thoroughly.
[ ]	Command items shown inside brackets are optional and can be used or excluded as the situation demands. Do not type the brackets.	host# <b>ls</b> [-a]
	The vertical bar (   ) is used in command examples to separate choices where multiple options exist. Select only one of the listed options. Do not type the vertical bar.	host# <b>set left right</b>
<b>AaBbCc123</b>	This block type depicts menus, buttons, and other controls that appear in Web browsers and other graphical interfaces.	Click the <b>Save</b> button.

---

## How to Get Help

If you need help, service, or technical assistance, visit our web site at the following address:

<http://www.ibm.com/support>

The warranty card received with your product provides details for contacting a customer support representative. If you are unable to locate this information, please contact your reseller. Before you call, prepare the following information:

- Serial number of the switch unit
- Software release version number
- Brief description of the problem and the steps you have already taken
- Technical support dump information (>> Main# /maint/tsdmp)



# **Part 1: Getting Started**



---

## Chapter 1. Switch Administration

Your Virtual Fabric Switch Module (VFSM) is ready to perform basic switching functions right out of the box. Some of the more advanced features, however, require some administrative configuration before they can be used effectively.

The extensive IBM Networking OS switching software included in the VFSM provides a variety of options for accessing the switch to perform configuration, and to view switch information and statistics.

This chapter discusses the various methods that can be used to administer the switch.

---

## Administration Interfaces

IBM N/OS provides a variety of user-interfaces for administration. These interfaces vary in character and in the methods used to access them: some are text-based, and some are graphical; some are available by default, and some require configuration; some can be accessed by local connection to the switch, and others are accessed remotely using various client applications. For example, administration can be performed using any of the following:

- The BladeCenter management module tools for general chassis management
- A built-in, text-based command-line interface and menu system for access via serial-port connection or an optional Telnet or SSH session
- The built-in Browser-Based Interface (BBI) available using a standard web-browser
- SNMP support for access through network management software such as IBM Director or HP OpenView

The specific interface chosen for an administrative session depends on user preferences, as well as the switch configuration and the available client tools.

In all cases, administration requires that the switch hardware is properly installed and turned on. (see the *IBM Virtual Fabric 10Gb Switch Module Installation Guide*).

## Management Module

The VFSM is an integral subsystem within the overall BladeCenter system. The BladeCenter chassis also includes a management module as the central element for overall chassis management and control. Using the tools available through the management module, the administrator can configure many of the VFSM features and can also access other VFSM administration interfaces.

For details, see [“Using the BladeCenter Management Module” on page 30](#).

## Command Line Interface

The N/OS Command Line Interface (CLI) provides a simple, direct method for switch administration. Using a basic terminal, you are presented with an organized hierarchy of menus, each with logically-related sub-menus and commands. These allow you to view detailed information and statistics about the switch, and to perform any necessary configuration and switch software maintenance. For example:

```
[Main Menu]
info      - Information Menu
stats     - Statistics Menu
cfg       - Configuration Menu
oper      - Operations Command Menu
boot      - Boot Options Menu
maint     - Maintenance Menu
diff      - Show pending config changes [global command]
apply     - Apply pending config changes [global command]
save      - Save updated config to FLASH [global command]
revert    - Revert pending or applied changes [global command]
exit      - Exit [global command, always available]

>> #
```

You can establish a connection to the CLI in any of the following ways:

- Serial connection via the serial port on the VFSM (this option is always available)
- Telnet connection from the chassis management module or over the network
- SSH connection from the chassis management module or over the network

## **Browser-Based Interface**

The Browser-based Interface (BBI) provides access to the common configuration, management and operation features of the VFSM through your Web browser.

For more information, refer to the *BBI Quick Guide*.

---

## Establishing a Connection

The factory default settings permit initial switch administration through *only* the BladeCenter management module or the built-in serial port. All other forms of access require additional switch configuration before they can be used.

Remote access using the network requires the accessing terminal to have a valid, routable connection to the switch interface. The client IP address may be configured manually, or an IPv4 address can be provided automatically through the switch using a service such as DHCP or BOOTP relay (see [“BOOTP/DHCP Client IP Address Services” on page 40](#)), or an IPv6 address can be obtained using IPv6 stateless address configuration.

**Note:** Throughout this manual, *IP address* is used in places where either an IPv4 or IPv6 address is allowed. IPv4 addresses are entered in dotted-decimal notation (for example, 10.10.10.1), while IPv6 addresses are entered in hexadecimal notation (for example, 2001:db8:85a3::8a2e:370:7334). In places where only one type of address is allowed, *IPv4 address* or *IPv6 address* is specified.

## Using the BladeCenter Management Module

The BladeCenter VFSM is an integral subsystem within the overall BladeCenter system. The BladeCenter chassis includes a management module as the central element for overall chassis management and control.

The VFSM uses internal port 15 (MGT1) and port 16 (MGT2) to communicate with the management module(s). Even when the VFSM is in a factory default configuration, you can use the 100 Mbps Ethernet port on each BladeCenter management module to configure and manage the VFSM.

**Note:** Support for each management module is provided by a separate management port (MGT1 and MGT2). One port is active, and the other is used as a backup.

## Factory-Default vs. MM-Assigned IP Addresses

Each VFSM must be assigned its own Internet Protocol version 4 (IPv4) address, which is used for communication with an SNMP network manager or other transmission control protocol/Internet Protocol (TCP/IP) applications (for example, BOOTP or TFTP). The factory-default IPv4 address is 10.90.90.8 $x$ , where  $x$  is based on the number of the bay into which the VFSM is installed. For additional information, see the *Installation Guide*. The management module assigns an IPv4 address of 192.168.70.1 $xx$ , where  $xx$  is also based on the number of the bay into which each VFSM is installed, as shown in the following table:

Table 2. VFSM IPv4 addresses, by switch-module bay numbers

Bay Number	Factory-Default IPv4 Address	IPv4 Address Assigned by MM
Bay 7	10.90.90.80	192.168.70.133
Bay 8	10.90.90.82	192.168.70.134
Bay 9	10.90.90.81	192.168.70.135
Bay 10	10.90.90.83	192.168.70.136

**Note:** Before you install the VFSM in Bay 8 or Bay 10, confirm that your blade I/O Expansion adapter supports communication to these I/O bays.

## Default Gateway

The default Gateway IP address determines where packets with a destination address outside the current subnet should be sent. Usually, the default Gateway is a router or host acting as an IP gateway to handle connections to other subnets of other TCP/IP networks. If you want to access the VFSM from outside your local network, use the management module to assign a default Gateway address to the VFSM. For example:

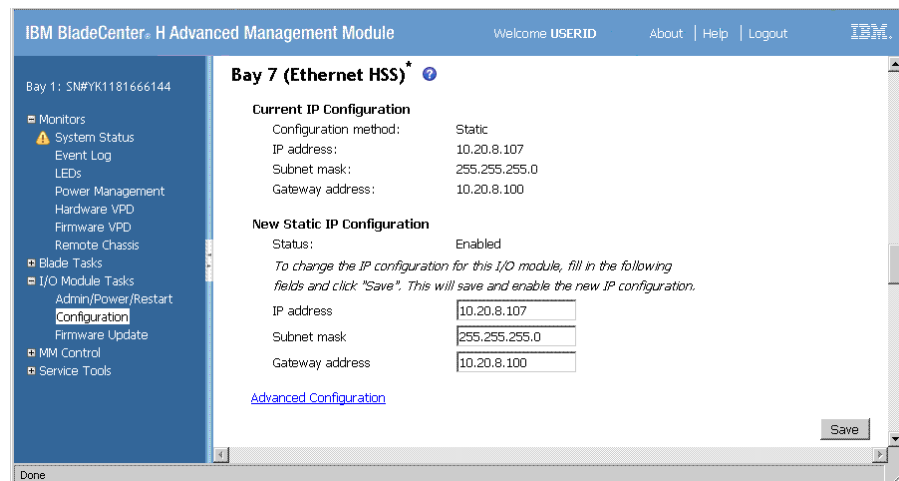
1. Choose I/O Module Tasks > Configuration from the navigation pane on the left.
2. Enter the default Gateway IP address (for example, 192.168.70.125).
3. Click Save.

## Configuring the Management Module for Switch Access

Complete the following initial configuration steps:

1. Connect the Ethernet port of the management module to a 10/100 Mbps network (with access to a management station) or directly to a management station.
2. Access and log on to the management module, as described in the *BladeCenter Management Module User's Guide*. The management module provides the appropriate IPv4 addresses for network access (see the applicable *BladeCenter Installation and User's Guide* publications for more information).
3. Select Configuration on the I/O Module Tasks menu on the left side of the BladeCenter Management Module window. See [Figure 1](#).

Figure 1. Switch Management on the BladeCenter Management Module



**Note:** The screen shown is an example only. Actual screen output may differ, depending on the type, version, and configuration of hardware and software installed in your systems.

4. You can use the default IPv4 addresses provided by the management module, or you can assign a new IPv4 or IPv6 address to the switch module through the management module. You can assign this IP address through one of the following methods:

- Manually through the BladeCenter management module
- Automatically through the IBM Director Configuration Wizard

**Note:** If you change the IP address of the VFSM, make sure that the switch module and the management module both reside on the same subnet.

5. Enable the following features in the management module:
  - External Ports (I/O Module Tasks > Admin/Power/Restart > Advanced Setup)
  - External management over all ports (Configuration > Advanced Configuration)
  - This setting is required if you want to access the management network through the external data ports (EXT<sub>x</sub>) on the VFSM.

The default value is `Disabled` for both features. If these features are not already enabled, change the value to `Enabled`, then click `Save`.

**Note:** In `Advanced Configuration > Advanced Setup`, enable “Preserve new IP configuration on all switch resets,” to retain the switch’s IP interface when you restore factory defaults. This setting preserves the management port’s IP address in the management module’s memory, so you maintain connectivity to the management module after a reset.

You can now start a Telnet or Secure Shell session to the switch CLI or ISCLI, or start a Web or secure HTTPS session to the switch BBI.

**Note:** Depending on your network requirements, BOOTP or DHCP may be required to be configured on the switch in order for remote clients to function. See [“BOOTP Relay Agent” on page 324](#) and [“Dynamic Host Configuration Protocol” on page 325](#) for more information.

## Using an External Switch Port

You can also use the external Ethernet ports (EXT<sub>x</sub>) on the VFSM for management and control of the switch. This feature must be enabled through the management module configuration utility program (shown in [Step 5 on page 32](#)).

See the applicable *BladeCenter Installation and User’s Guide* publications for more information.

## Using Telnet

A Telnet connection offers the convenience of accessing the switch from a workstation connected to the network. Telnet access provides the same options for user and administrator access as those available through the management module or console port.

By default, Telnet access is enabled. Use the following commands (available on the console only) to disable or re-enable Telnet access:

```
VFSM(config)# [no] access telnet enable
```

To use the management module to access the VFSM through Telnet, choose I/O Module Tasks > Configuration from the navigation pane on the left. Then select a bay number and click Advanced Configuration > Start Telnet/Web Session > Start Telnet Session. A Telnet window opens a connection to the Switch Module (requires Java 1.4 Plug-in).

Once the switch is configured with an IP address and gateway, you can use Telnet to access switch administration from any workstation connected to the management network.

To establish a Telnet connection with the switch, run the Telnet program on your workstation and issue the following Telnet command:

```
telnet <switch IPv4 or IPv6 address>
```

You will then be prompted to enter a password as explained [“Switch Login Levels” on page 43](#).

## Using Secure Shell

Although a remote network administrator can manage the configuration of a VFSM via Telnet, this method does not provide a secure connection. The Secure Shell (SSH) protocol enables you to securely log into another device over a network to execute commands remotely. As a secure alternative to using Telnet to manage switch configuration, SSH ensures that all data sent over the network is encrypted and secure.

The switch can do only one session of key/cipher generation at a time. Thus, a SSH/SCP client will not be able to login if the switch is doing key generation at that time. Similarly, the system will fail to do the key generation if a SSH/SCP client is logging in at that time.

The supported SSH encryption and authentication methods are listed below.

- Server Host Authentication: Client RSA-authenticates the switch when starting each connection
- Key Exchange: ecdh-sha2-nistp521, ecdh-sha2-nistp384, ecdh-sha2-nistp256, ecdh-sha2-nistp224, ecdh-sha2-nistp192, rsa2048-sha256, rsa1024-sha1, diffie-hellman-group-exchange-sha256, diffie-hellman-group-exchange-sha1, diffie-hellman-group14-sha1, diffie-hellman-group1-sha1
- Encryption: aes128-ctr, aes128-cbc, rijndael128-cbc, blowfish-cbc, 3des-cbc, arcfour256, arcfour128, arcfour

- MAC: hmac-sha1, hmac-sha1-96, hmac-md5, hmac-md5-96
- User Authentication: Local password authentication, RADIUS, TACACS+

IBM Networking OS implements the SSH version 2.0 standard and is confirmed to work with SSH version 2.0-compliant clients such as the following:

- OpenSSH\_5.4p1 for Linux
- Secure CRT Version 5.0.2 (build 1021)
- Putty SSH release 0.60

## Using SSH to Access the Switch

By default, the SSH feature is disabled. For information on enabling and using SSH for switch access, see [“Secure Shell and Secure Copy” on page 71](#).

Once the IP parameters are configured and the SSH service is enabled, you can access the command line interface using an SSH connection.

To establish an SSH connection with the switch, run the SSH program on your workstation by issuing the SSH command, followed by the switch IPv4 or IPv6 address:

```
# ssh <switch IP address>
```

You will then be prompted to enter a password as explained [“Switch Login Levels” on page 43](#).

## Using a Web Browser

The switch provides a Browser-Based Interface (BBI) for accessing the common configuration, management and operation features of the VFSM through your Web browser.

By default, BBI access via HTTP is enabled on the switch.

To use the management module to access the VFSM through a Web session, choose I/O Module Tasks > Configuration from the navigation pane on the left of the management tool. Next, select a bay number and click Advanced Configuration > Start Telnet/Web Session > Start Web Session. A browser window opens a connection to the Switch Module.

You can access the BBI directly from an open Web browser window. Enter the URL using the IP address of the switch interface (for example, `http://<IPv4 or IPv6 address>`).

## Configuring HTTP Access to the BBI

By default, BBI access via HTTP is enabled on the switch.

To disable or re-enable HTTP access to the switch BBI, use the following commands:

```
VFSM(config)# access http enable           (Enable HTTP access)

-or-

VFSM(config)# no access http enable        (Disable HTTP access)
```

The default HTTP web server port to access the BBI is port 80 (required by the management module). However, you can change the default Web server port with the following command:

```
VFSM(config)# access http port <TCP port number>
```

To access the BBI from a workstation, open a Web browser window and type in the URL using the IP address of the switch interface (for example, `http://<IPv4 or IPv6 address>`).

## Configuring HTTPS Access to the BBI

The BBI can also be accessed via a secure HTTPS connection.

### 1. Enable HTTPS.

By default, BBI access via HTTPS is disabled on the switch. To enable BBI Access via HTTPS, use the following command:

```
VFSM(config)# access http enable           (Enable HTTP access)
```

### 2. Set the HTTPS server port number (optional).

To change the HTTPS Web server port number from the default port 443, use the following command:

```
VFSM(config)# access http port <TCP port number>
```

### 3. Generate the HTTPS certificate.

Accessing the BBI via HTTPS requires that you generate a certificate to be used during the key exchange. A default certificate is created the first time HTTPS is enabled, but you can create a new certificate defining the information you want to be used in the various fields.

```
VFSM(config)# access https generate-certificate
Country Name (2 letter code) []: <country code>
State or Province Name (full name) []: <state>
Locality Name (eg, city) []: <city>
Organization Name (eg, company) []: <company>
Organizational Unit Name (eg, section) []: <org. unit>
Common Name (eg, YOUR name) []: <name>
Email (eg, email address) []: <email address>
Confirm generating certificate? [y/n]: y
Generating certificate. Please wait (approx 30 seconds)
restarting SSL agent
```

4. Save the HTTPS certificate.

The certificate is valid only until the switch is rebooted. In order to save the certificate so that it is retained beyond reboot or power cycles, use the following command:

```
VFSM(config)# access https save-certificate
```

When a client (e.g. web browser) connects to the switch, the client is asked to accept the certificate and verify that the fields match what is expected. Once BBI access is granted to the client, the BBI can be used as described in the *IBM Networking OS 7.8 BBI Quick Guide*.

## Browser-Based Interface Summary

The BBI is organized at a high level as follows:

**Context buttons**—These buttons allow you to select the type of action you wish to perform. The *Configuration* button provides access to the configuration elements for the entire switch. The *Statistics* button provides access to the switch statistics and state information. The *Dashboard* button allows you to display the settings and operating status of a variety of switch features.

**Navigation Window**—This window provides a menu list of switch features and functions:

- **System**—this folder provides access to the configuration elements for the entire switch.
  - General
  - User Table
  - Radius
  - TACACS+
  - LDAP
  - NTP
  - sFlow
  - Bridge Module
  - Syslog/Trap Features
  - Config/Image Control
  - Management Network
  - Transceiver
  - Protected Mode
  - Chassis
- **Switch Ports**—Configure each of the physical ports on the switch.
- **Port-Based Port Mirroring**—Configure port mirroring behavior.
- **Layer 2**—Configure Layer 2 features for the switch.
  - 802.1X
  - FDB
  - Virtual LANs
  - Spanning Tree Groups
  - MSTP/RSTP
  - LLDP
  - Failover
  - Hot Links
  - Trunk Groups
  - Trunk Hash
  - LACP
  - BPDU Guard
  - PVST+ compatibility
  - MAC Address Notification

- **RMON Menu**—Configure Remote Monitoring features for the switch.
- **Layer 3**—Configure Layer 3 features for the switch.
  - IP Interfaces
  - IP Loopback Interfaces
  - Network Routes
  - Network IPv6 Routes
  - Static IPMC Routes
  - ARP
  - NBR
  - Network Filters
  - Route Maps
  - Border Gateway Protocol
  - Default Gateways
  - IPv6 Default Gateways
  - IGMP
  - OSPF Routing Protocol
  - Routing Information Protocol
  - Virtual Router Redundancy Protocol
  - Domain Name System
  - Bootstrap Protocol Relay
  - General
- **QoS**—Configure Quality of Service features for the switch.
  - 802.1p
  - DSCP
- **Access Control**—Configure Access Control Lists to filter IP packets.
  - Access Control Lists
  - VLAN Map
  - Access Control List Groups
- **CEE** – Configure Converged Enhanced Ethernet (CEE).
- **FCoE** – Configure FibreChannel over Ethernet (FCoE).
- **Virtualization** – Configure vNICs and VMready for virtual machine (VM) support.

For information on using the BBI, refer to the *IBM Networking OS 7.8 BBI Quick Guide*.

## Using Simple Network Management Protocol

N/OS provides Simple Network Management Protocol (SNMP) version 1, version 2, and version 3 support for access through any network management software, such as IBM Director or HP-OpenView.

To access the SNMP agent on the VFSM, the read and write community strings on the SNMP manager should be configured to match those on the switch.

The read and write community strings on the switch can be changed using the following commands:

```
VFSM(config)# snmp-server read-community <1-32 characters>  
  
-and-  
  
VFSM(config)# snmp-server write-community <1-32 characters>
```

The SNMP manager should be able to reach any one of the IP interfaces on the switch.

For the SNMP manager to receive the SNMPv1 traps sent out by the SNMP agent on the switch, configure the trap host on the switch with the following commands:

```
VFSM(config)# snmp-server trap-source <trap source IP interface>  
VFSM(config)# snmp-server host <IPv4 address> <trap host community string>
```

For more information on SNMP usage and configuration, see [“Simple Network Management Protocol” on page 487](#).

---

## BOOTP/DHCP Client IP Address Services

For remote switch administration, the client terminal device must have a valid IP address on the same network as a switch interface. The IP address on the client device may be configured manually, or obtained automatically using IPv6 stateless address configuration, or an IPv4 address may be obtained automatically via BOOTP or DHCP relay as discussed below.

The VFSM can function as a relay agent for Bootstrap Protocol (BOOTP) or DHCP. This allows clients to be assigned an IPv4 address for a finite lease period, reassigning freed addresses later to other clients.

Acting as a relay agent, the switch can forward a client's IPv4 address request to up to four BOOTP/DHCP servers. In addition to the four global BOOTP/DHCP servers, up to four domain-specific BOOTP/DHCP servers can be configured for each of up to 10 VLANs.

When a switch receives a BOOTP/DHCP request from a client seeking an IPv4 address, the switch acts as a proxy for the client. The request is forwarded as a UDP Unicast MAC layer message to the BOOTP/DHCP servers configured for the client's VLAN, or to the global BOOTP/DHCP servers if no domain-specific BOOTP/DHCP servers are configured for the client's VLAN. The servers respond to the switch with a Unicast reply that contains the IPv4 default gateway and the IPv4 address for the client. The switch then forwards this reply back to the client.

DHCP is described in RFC 2131, and the DHCP relay agent supported on the VFSM is described in RFC 1542. DHCP uses UDP as its transport protocol. The client sends messages to the server on port 67 and the server sends messages to the client on port 68.

BOOTP and DHCP relay are collectively configured using the BOOTP commands and menus on the VFSM.

## Global BOOTP Relay Agent Configuration

To enable the VFSM to be a BOOTP (or DHCP) forwarder, enable the BOOTP relay feature, configure up to four global BOOTP server IPv4 addresses on the switch, and enable BOOTP relay on the interface(s) on which the client requests are expected.

Generally, you should configure BOOTP for the switch IP interface that is closest to the client, so that the BOOTP server knows from which IPv4 subnet the newly allocated IPv4 address should come.

In the VFSM implementation, there are no primary or secondary BOOTP servers. The client request is forwarded to all the global BOOTP servers configured on the switch (if no domain-specific servers are configured). The use of multiple servers provide failover redundancy. However, no health checking is supported.

1. Use the following commands to configure global BOOTP relay servers:

```
VFSM(config)# ip bootp-relay enable
VFSM(config)# ip bootp-relay server <1-5> address <IPv4 address>
```

2. Enable BOOTP relay on the appropriate IP interfaces.

BOOTP/DHCP Relay functionality may be assigned on a per-interface basis using the following commands:

```
VFSM(config)# interface ip <interface number>
VFSM(config-ip-if)# relay
VFSM(config-ip-if)# exit
```

## Domain-Specific BOOTP Relay Agent Configuration

Use the following commands to configure up to four domain-specific BOOTP relay agents for each of up to 10 VLANs:

```
VFSM(config)# ip bootp-relay bcast-domain <1-10> vlan <VLAN number>
VFSM(config)# ip bootp-relay bcast-domain <1-10> server <1-5> address <IPv4 address>
VFSM(config)# ip bootp-relay bcast-domain <1-10> enable
```

As with global relay agent servers, domain-specific BOOTP/DHCP functionality may be assigned on a per-interface basis (see [Step 2](#) in [page 41](#)).

## DHCP Option 82

DHCP Option 82 provides a mechanism for generating IP addresses based on the client device's location in the network. When you enable the DHCP relay agent option on the switch, it inserts the relay agent information option 82 in the packet, and sends a unicast BOOTP request packet to the DHCP server. The DHCP server uses the option 82 field to assign an IP address, and sends the packet, with the original option 82 field included, back to the relay agent. DHCP relay agent strips off the option 82 field in the packet and sends the packet to the DHCP client.

Configuration of this feature is optional. The feature helps resolve several issues where untrusted hosts access the network. See RFC 3046 for details.

Given below are the commands to configure DHCP Option 82:

```
VFSM(config)# [no] ip bootp-relay information ?  
enable Enable option 82  
policy Define reforwarding policy
```

## DHCP Snooping

DHCP snooping provides security by filtering untrusted DHCP packets and by building and maintaining a DHCP snooping binding table. This feature is applicable only to IPv4 and only works in non-stacking mode.

An untrusted interface is a port that is configured to receive packets from outside the network or firewall. A trusted interface receives packets only from within the network. By default, all DHCP ports are untrusted.

The DHCP snooping binding table contains the MAC address, IP address, lease time, binding type, VLAN number, and port number that correspond to the local untrusted interface on the switch; it does not contain information regarding hosts interconnected with a trusted interface.

By default, DHCP snooping is disabled on all VLANs. You can enable DHCP snooping on one or more VLANs. You must enable DHCP snooping globally. To enable this feature, enter the commands below:

```
VFSM(config)# ip dhcp snooping vlan <vlan number(s)>  
VFSM(config)# ip dhcp snooping
```

Given below is an example of DHCP snooping configuration, where the DHCP server and client are in VLAN 100, and the server connects using port 1.

```
VFSM(config)# ip dhcp snooping vlan 100  
VFSM(config)# ip dhcp snooping  
VFSM(config)# interface port 1  
VFSM(config-if)# ip dhcp snooping trust (Optional; Set port as trusted)  
VFSM(config-if)# ip dhcp snooping information option-insert (Optional; add DHCP option 82)  
VFSM(config-if)# ip dhcp snooping limit rate 100 (Optional; Set DHCP packet rate)
```

---

## Switch Login Levels

To enable better switch management and user accountability, three levels or *classes* of user access have been implemented on the VFSM. Levels of access to CLI, Web management functions, and screens increase as needed to perform various switch management tasks. Conceptually, access classes are defined as follows:

- User interaction with the switch is completely passive—nothing can be changed on the VFSM. Users may display information that has no security or privacy implications, such as switch statistics and current operational state information.
- Operators can only effect temporary changes on the VFSM. These changes will be lost when the switch is rebooted/reset. Operators have access to the switch management features used for daily switch operations. Because any changes an operator makes are undone by a reset of the switch, operators cannot severely impact switch operation.
- Administrators are the only ones that may make permanent changes to the switch configuration—changes that are persistent across a reboot/reset of the switch. Administrators can access switch functions to configure and troubleshoot problems on the VFSM. Because administrators can also make temporary (operator-level) changes as well, they must be aware of the interactions between temporary and permanent changes.

Access to switch functions is controlled through the use of unique surnames and passwords. Once you are connected to the switch via local Telnet, remote Telnet, or SSH, you are prompted to enter a password. The default user names/password for each access level are listed in the following table.

**Note:** It is recommended that you change default switch passwords after initial configuration and as regularly as required under your network security policies. For more information, see [“Changing the Switch Passwords” on page 70](#).

*Table 3. User Access Levels - Default Settings*

User Account	Password	Description and Tasks Performed	Status
user	user	The User has no direct responsibility for switch management. He or she can view all switch status information and statistics, but cannot make any configuration changes to the switch.	Disabled
oper	oper	The Operator manages all functions of the switch. The Operator can reset ports, except the management ports.	Disabled
admin	admin	The superuser Administrator has complete access to all menus, information, and configuration commands on the VFSM, including the ability to change both the user and administrator passwords.	Enabled

**Note:** Access to each user level (except admin account) can be disabled by setting the password to an empty value. To disable admin account, use the command:

VFSM(config)# no access user administrator-enable. Admin account can be disabled only if there is at least one user account enabled and configured with administrator privilege.

---

## Setup vs. the Command Line

Once the administrator password is verified, you are given complete access to the switch. If the switch is still set to its factory default configuration, the system will ask whether you wish to run Setup (see [“Initial Setup” on page 51](#)”), a utility designed to help you through the first-time configuration process. If the switch has already been configured, the command line is displayed instead.

---

## Boot Strict Mode

The implementations specified in this section are compliant with National Institute of Standards and Technology (NIST) Special Publication (SP) 800-131a. SP 800-131a

The Virtual Fabric Switch Module can operate in two boot modes:

- Compatibility mode (default): This is the default switch boot mode. This mode may use algorithms and key lengths that may not be allowed/acceptable by NIST SP 800-131a specification. This mode is useful in maintaining compatibility with previous releases and in environments that have lesser data security requirements.
- Strict mode: Encryption algorithms, protocols, and key lengths in strict mode are compliant NIST SP 800-131a specification.

When in boot strict mode, the switch uses Secure Sockets Layer (SSL)/Transport Layer Security (TLS) 1.2 protocols to ensure confidentiality of the data to and from the switch.

By default, HTTP, Telnet, and SNMPv1 and SNMPv2 are enabled on the VFSM.

Before enabling strict mode, ensure the following:

- The software version on all connected switches is IBM N/OS 7.8.
- The supported protocol versions and cryptographic cipher suites between clients and servers are compatible. For example: if using SSH to connect to the switch, ensure that the SSH client supports SSHv2 and a strong cipher suite that is compliant with the NIST standard.
- Compliant Web server certificate is installed on the switch, if using BBI.
- A new self-signed certificate is generated for the switch (VFSM(config)# access https generate-certificate). The new certificate is generated using 2048-bit RSA key and SHA-256 digest.
- Protocols that are not NIST SP 800-131A compliant must be disabled or not used.
- Only SSHv2 or higher is used.
- The current configuration, if any, must be saved in a location external to the switch. When the switch reboots, both the startup and running configuration are lost.
- Only protocols/algorithms compliant with NIST SP 800-131a specification are used/enabled on the switch. Please see the NIST SP 800-131a publication for details. The following table lists the acceptable protocols and algorithms:

*Table 4. Acceptable Protocols and Algorithms*

Protocol/Function	Strict Mode Algorithm	Compatibility Mode Algorithm
BGP	BGP does not comply with NIST SP 800-131a specification. When in strict mode, BGP is disabled. However, it can be enabled, if required.	Acceptable
Certificate Generation	RSA-2048 SHA-256	RSA 2048 SHA 256
Certificate Acceptance	RSA 2048 or higher SHA 224 or higher	RSA SHA, SHA2

Table 4. Acceptable Protocols and Algorithms

Protocol/Function	Strict Mode Algorithm	Compatibility Mode Algorithm
HTTPS	TLS 1.2 only See “Acceptable Cipher Suites” on page 48;	TLS 1.0, 1.1, 1.2 See “Acceptable Cipher Suites” on page 48;
IKE		
Key Exchange	DH Group 24	DH group 1, 2, 5, 14, 24
Encryption	3DES, AES-128-CBC	3DES, AES-128-CBC
Integrity	HMAC-SHA1	HMAC-SHA1, HMAC-MD5
IPSec		
AH	HMAC-SHA1	HMAC-SHA1, HMAC-MD5
ESP	3DES, AES-128-CBC, HMAC-SHA1	3DES, AES-128-CBC, HMAC-SHA1, HMAC-MD5
LDAP	LDAP does not comply with NIST SP 800-131a specification. When in strict mode, LDAP is disabled. However, it can be enabled, if required.	Acceptable
OSPF	OSPF does not comply with NIST SP 800-131a specification. When in strict mode, OSPF is disabled. However, it can be enabled, if required.	Acceptable
RADIUS	RADIUS does not comply with NIST SP 800-131a specification. When in strict mode, RADIUS is disabled. However, it can be enabled, if required.	Acceptable
Random Number Generator	NIST SP 800-90A AES CTR DRBG	NIST SP 800-90A AES CTR DRBG
Secure NTP	Secure NTP does not comply with NIST SP 800-131a specification. When in strict mode, secure NTP is disabled. However, it can be enabled, if required.	Acceptable
SLP	SHA-256 or higher RSA/DSA 2048 or higher	
SNMP	SNMPv3 only AES-128-CFB-128/SHA1	SNMPv1, SNMPv2, SNMPv3 DES/MD5, AES-128-CFB-128/SHA1
SSH/SFTP		
Host Key	SSH-RSA	SSH-RSA

Table 4. Acceptable Protocols and Algorithms

Protocol/Function	Strict Mode Algorithm	Compatibility Mode Algorithm
Key Exchange	ECDH-SHA2-NISTP521 ECDH-SHA2-NISTP384 ECDH-SHA2-NISTP256 ECDH-SHA2-NISTP224 RSA2048-SHA256 DIFFIE-HELL- MAN-GROUP-EXCHANGE-SHA256 DIFFIE-HELL- MAN-GROUP-EXCHANGE-SHA1	ECDH-SHA2-NISTP521 ECDH-SHA2-NISTP384 ECDH-SHA2-NISTP256 ECDH-SHA2-NISTP224 ECDH-SHA2-NISTP192 RSA2048-SHA256 RSA1024-SHA1 DIFFIE-HELL- MAN-GROUP-EXCHANGE-SHA 256 DIFFIE-HELL- MAN-GROUP-EXCHANGE-SHA 1 DIFFIE-HELL- MAN-GROUP14-SHA1 DIFFIE-HELL- MAN-GROUP1-SHA1
Encryption	AES128-CTR AES128-CBC 3DES-CBC	AES128-CTR AES128-CBC RIJNDAEL128-CBC BLOWFISH-CBC 3DES-CBC ARCFOUR256 ARCFOUR128 ARCFOUR
MAC	HMAC-SHA1 HMAC-SHA1-96	HMAC-SHA1 HMAC-SHA1-96 HMAC-MD5 HMAC-MD5-96
TACACS+	TACACS+ does not comply with NIST SP 800-131a specification. When in strict mode, TACACS+ is disabled. However, it can be enabled, if required.	Acceptable

## Acceptable Cipher Suites

The following cipher suites are acceptable (listed in the order of preference) when the Virtual Fabric Switch Module is in compatibility mode:

Table 5. List of Acceptable Cipher Suites in Compatibility Mode

Cipher ID	Key Exchange	Authentication	Encryption	MAC	Cipher Name
0xC027	ECDHE	RSA	AES_128_CBC	SHA256	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
0xC013	ECDHE	RSA	AES_128_CBC	SHA1	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
0xC012	ECDHE	RSA	3DES	SHA1	SSL_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA
0xC011	ECDHE	RSA	RC4	SHA1	SSL_ECDHE_RSA_WITH_RC4_128_SHA
0x002F	RSA	RSA	AES_128_CBC	SHA1	TLS_RSA_WITH_AES_128_CBC_SHA
0x003C	RSA	RSA	AES_128_CBC	SHA256	TLS_RSA_WITH_AES_128_CBC_SHA256
0x0005	RSA	RSA	RC4	SHA1	SSL_RSA_WITH_RC4_128_SHA
0x000A	RSA	RSA	3DES	SHA1	SSL_RSA_WITH_3DES_EDE_CBC_SHA
0x0033	DHE	RSA	AES-128_CBC	SHA1	TLS_DHE_RSA_WITH_AES_128_CBC_SHA
0x0067	DHE	RSA	AES_128_CBC	SHA256	TLS_DHE_RSA_WITH_AES_128_CBC_SHA256

*Table 5. List of Acceptable Cipher Suites in Compatibility Mode*

0x0016	DHE	RSA	3DES	SHA1	SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA
--------	-----	-----	------	------	-----------------------------------

The following cipher suites are acceptable (listed in the order of preference) when the Virtual Fabric Switch Module is in strict mode:

*Table 6. List of Acceptable Cipher Suites in Strict Mode*

Cipher ID	Key Exchange	Authentication	Encryption	MAC	Cipher Name
0xC027	ECDHE	RSA	AES_128_CBC	SHA256	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
0xC013	ECDHE	RSA	AES_128_CBC	SHA1	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
0xC012	ECDHE	RSA	3DES	SHA1	SSL_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA
0x0033	DHE	RSA	AES-128_CBC	SHA1	TLS_DHE_RSA_WITH_AES_128_CBC_SHA
0x0067	DHE	RSA	AES_128_CBC	SHA256	TLS_DHE_RSA_WITH_AES_128_CBC_SHA256
0x0016	DHE	RSA	3DES	SHA1	SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA
0x002F	RSA	RSA	AES_128_CBC	SHA1	TLS_RSA_WITH_AES_128_CBC_SHA
0x003C	RSA	RSA	AES_128_CBC	SHA256	TLS_RSA_WITH_AES_128_CBC_SHA256
0x000A	RSA	RSA	3DES	SHA1	SSL_RSA_WITH_3DES_EDE_CBC_SHA

## Configuring Strict Mode

To change the switch mode to boot strict mode, use the following command:

```
VFSM(config)# [no] boot strict enable
```

When strict mode is enabled, you will see the following message:

```
Warning, security strict mode limits the cryptographic algorithms used by secure
protocols on this switch. Please see the documentation for full details, and
verify that peer devices support acceptable algorithms before enabling this
mode. The mode change will take effect after reloading the switch and the
configuration will be wiped during the reload. System will enter security strict
mode with default factory configuration at next boot up.

Do you want SNMPV3 support old default users in strict mode (y/n)?
```

When strict mode is disabled, the following message is displayed:

```
Warning, disabling security strict mode. The mode change will take effect after
reloading the switch.
```

You must reboot the switch for the boot strict mode enable/disable to take effect.

## Limitations

In IBM N/OS 7.8, consider the following limitation/restrictions if you need to operate the switch in boot strict mode:

- Strict mode can be enabled on the VFSM only in the setup phase. If you enable strict mode on the switch after it has been configured, all the configuration will be lost when you reboot the switch.
- Power ITEs and High-Availability features do not comply with NIST SP 800-131A specification.

- The VFSM will not discover Platform agents/Common agents that are not in strict mode.
- Web browsers that do not use TLS 1.2 cannot be used.
- Limited functions of the switch managing Windows will be available.

---

## Chapter 2. Initial Setup

To help with the initial process of configuring your switch, the IBM Networking OS software includes a Setup utility. The Setup utility prompts you step-by-step to enter all the necessary information for basic configuration of the switch.

Whenever you log in as the system administrator under the factory default configuration, you are asked whether you wish to run the Setup utility. Setup can also be activated manually from the command line interface any time after login:

```
VFSM(config)# setup
```

---

## Information Needed for Setup

Setup requests the following information:

- Basic system information
  - Date & time
  - Whether to use Spanning Tree Group or not
- Optional configuration for each port
  - Speed, duplex, flow control, and negotiation mode (as appropriate)
  - Whether to use VLAN tagging/trunk mode or not (as appropriate)
- Optional configuration for each VLAN
  - Name of VLAN
  - Which ports are included in the VLAN
- Optional configuration of IP parameters
  - IP address/mask and VLAN for each IP interface
  - IP addresses for default gateway
  - Whether IP forwarding is enabled or not

---

## Default Setup Options

The Setup prompt appears automatically whenever you login as the system administrator under the factory default settings.

1. Connect to the switch.

After connecting, the login prompt will appear as shown below.

```
Enter Password:
```

2. Enter `admin` as the default administrator password.

If the factory default configuration is detected, the system prompts:

```
IBM Virtual Fabric 10Gb Switch Module
18:44:05 Wed Jan 3, 2013

The switch is booted with factory default configuration.
To ease the configuration of the switch, a "Set Up" facility which
will prompt you with those configuration items that are essential to the
operation of the switch is provided.
Would you like to run "Set Up" to configure the switch? [y/n]:
```

**Note:** If the default `admin` login is unsuccessful, or if the administrator Main Menu appears instead, the system configuration has probably been changed from the factory default settings. If desired, return the switch to its factory default configuration.

3. Enter `y` to begin the initial configuration of the switch, or `n` to bypass the Setup facility.

---

## Stopping and Restarting Setup Manually

### Stopping Setup

To abort the Setup utility, press <Ctrl-C> during any Setup question. When you abort Setup, the system will prompt:

```
Would you like to run from top again? [y/n]
```

Enter `n` to abort Setup, or `y` to restart the Setup program at the beginning.

### Restarting Setup

You can restart the Setup utility manually at any time by entering the following command at the administrator prompt:

```
VFSM(config)# setup
```

---

## Setup Part 1: Basic System Configuration

When Setup is started, the system prompts:

"Set Up" will walk you through the configuration of System Date and Time, Spanning Tree, Port Speed/Mode, VLANs, and IP interfaces. [type Ctrl-C to abort "Set Up"]

1. Enter *y* if you will be configuring VLANs. Otherwise enter *n*.

If you decide not to configure VLANs during this session, you can configure them later using the configuration menus, or by restarting the Setup facility. For more information on configuring VLANs, see the *IBM Networking OS Application Guide*.

Next, the Setup utility prompts you to input basic system information.

2. Enter the year of the current date at the prompt:

System Date:  
Enter year [2013]:

Enter the four-digits that represent the year. To keep the current year, press <Enter>.

3. Enter the month of the current system date at the prompt:

System Date:  
Enter month [1]:

Enter the month as a number from 1 to 12. To keep the current month, press <Enter>.

4. Enter the day of the current date at the prompt:

Enter day [3]:

Enter the date as a number from 1 to 31. To keep the current day, press <Enter>. The system displays the date and time settings:

System clock set to 18:55:36 Wed Jan 28, 2009.

5. Enter the hour of the current system time at the prompt:

System Time:  
Enter hour in 24-hour format [18]:

Enter the hour as a number from 00 to 23. To keep the current hour, press <Enter>.

6. Enter the minute of the current time at the prompt:

Enter minutes [55]:

Enter the minute as a number from 00 to 59. To keep the current minute, press <Enter>.

7. Enter the seconds of the current time at the prompt:

```
Enter seconds [37]:
```

Enter the seconds as a number from 00 to 59. To keep the current second, press <Enter>. The system then displays the date and time settings:

```
System clock set to 8:55:36 Wed Jan 28, 2013.
```

8. Turn Spanning Tree Protocol on or off at the prompt:

```
Spanning Tree:  
Current Spanning Tree Group 1 setting: ON  
Turn Spanning Tree Group 1 OFF? [y/n]
```

Enter **y** to turn off Spanning Tree, or enter **n** to leave Spanning Tree on.

---

## Setup Part 2: Port Configuration

**Note:** When configuring port options for your switch, some prompts and options may be different.

1. Select whether you will configure VLANs and VLAN tagging/trunk mode for ports:

```
Port Config:
Will you configure VLANs and VLAN Tagging/Trunk-Mode for ports? [y/n]
```

If you wish to change settings for VLANs, enter `y`, or enter `n` to skip VLAN configuration.

**Note:** The sample screens that appear in this document might differ slightly from the screens displayed by your system. Screen content varies based on the type of BladeCenter unit that you are using and the firmware versions and options that are installed.

2. Select the port to configure, or skip port configuration at the prompt:

If you wish to change settings for individual ports, enter the number of the port you wish to configure. To skip port configuration, press `<Enter>` without specifying any port and go to [“Setup Part 3: VLANs” on page 59](#).

3. Configure Gigabit Ethernet port flow parameters.

The system prompts:

```
Gig Link Configuration:
Port Flow Control:
Current Port EXT1 flow control setting:    both
Enter new value ["rx"/"tx"/"both"/"none"]:
```

Enter `rx` to enable receive flow control, `tx` for transmit flow control, `both` to enable both, or `none` to turn flow control off for the port. To keep the current setting, press `<Enter>`.

4. Configure Gigabit Ethernet port autonegotiation mode.

If you selected a port that has a Gigabit Ethernet connector, the system prompts:

```
Port Auto Negotiation:
Current Port EXT1 autonegotiation:        on
Enter new value ["on"/"off"]:
```

Enter `on` to enable port autonegotiation, `off` to disable it, or press `<Enter>` to keep the current setting.

5. If configuring VLANs, enable or disable VLAN tagging/trunk mode for the port.  
If you have selected to configure VLANs back in Part 1, the system prompts:

```
Port VLAN tagging/trunk-mode config (tagged/trunk-mode port can be a member of
multiple VLANs)
Current VLAN tag/trunk-mode support:                disabled
Enter new VLAN tag/trunk-mode support [d/e]:
```

Enter **d** to disable VLAN tagging/trunk mode for the port or enter **e** to enable VLAN tagging/trunk mode for the port. To keep the current setting, press <Enter>.

6. The system prompts you to configure the next port:

```
Enter port (INT1-14, MGT1-2, EXT1-11):
```

When you are through configuring ports, press <Enter> without specifying any port. Otherwise, repeat the steps in this section.

---

## Setup Part 3: VLANs

If you chose to skip VLANs configuration back in Part 2, skip to [“Setup Part 4: IP Configuration” on page 60](#).

1. Select the VLAN to configure, or skip VLAN configuration at the prompt:

```
VLAN Config:
Enter VLAN number from 2 to 4094, NULL at end:
```

If you wish to change settings for individual VLANs, enter the number of the VLAN you wish to configure. To skip VLAN configuration, press <Enter> without typing a VLAN number and go to [“Setup Part 4: IP Configuration” on page 60](#).

2. Enter the new VLAN name at the prompt:

```
Current VLAN name: VLAN 2
Enter new VLAN name:
```

Entering a new VLAN name is optional. To use the pending new VLAN name, press <Enter>.

3. Enter the VLAN port numbers:

```
Define Ports in VLAN:
Current VLAN 2: empty
Enter ports one per line, NULL at end:
```

Enter each port, by port number or port alias, and confirm placement of the port into this VLAN. When you are finished adding ports to this VLAN, press <Enter> without specifying any port.

4. Configure Spanning Tree Group membership for the VLAN:

```
Spanning Tree Group membership:
Enter new Spanning Tree Group index [1-127]:
```

5. The system prompts you to configure the next VLAN:

```
VLAN Config:
Enter VLAN number from 2 to 4094, NULL at end:
```

Repeat the steps in this section until all VLANs have been configured. When all VLANs have been configured, press <Enter> without specifying any VLAN.

---

## Setup Part 4: IP Configuration

The system prompts for IPv4 parameters.

Although the switch supports both IPv4 and IPv6 networks, the Setup utility permits only IPv4 configuration. For IPv6 configuration, see [“Internet Protocol Version 6” on page 329](#).

### IP Interfaces

IP interfaces are used for defining the networks to which the switch belongs.

Up to 128 IP interfaces can be configured on the Virtual Fabric Switch Module (VFSM). The IP address assigned to each IP interface provides the switch with an IP presence on your network. No two IP interfaces can be on the same IP network. The interfaces can be used for connecting to the switch for remote configuration, and for routing between subnets and VLANs (if used).

**Note:** Interface 128 is reserved for IPv4 switch management. If the IPv6 feature is enabled, interface 127 is also reserved.

1. Select the IP interface to configure, or skip interface configuration at the prompt:

```
IP Config:

IP interfaces:
Enter interface number: (1-128)
```

If you wish to configure individual IP interfaces, enter the number of the IP interface you wish to configure. To skip IP interface configuration, press <Enter> without typing an interface number and go to [“Default Gateways” on page 62](#).

**Note:** If you change the IP address of interfaces reserved for switch management, you can lose the connection to the management module. Use the management module to change the IP address of the VFSM.

2. For the specified IP interface, enter the IP address in IPv4 dotted decimal notation:

```
Current IP address:      0.0.0.0
Enter new IP address:
```

To keep the current setting, press <Enter>.

3. At the prompt, enter the IPv4 subnet mask in dotted decimal notation:

```
Current subnet mask:      0.0.0.0
Enter new subnet mask:
```

To keep the current setting, press <Enter>.

4. If configuring VLANs, specify a VLAN for the interface.

This prompt appears if you selected to configure VLANs back in Part 1:

```
Current VLAN:      1
Enter new VLAN [1-4094]:
```

Enter the number for the VLAN to which the interface belongs, or press <Enter> without specifying a VLAN number to accept the current setting.

5. At the prompt, enter *y* to enable the IP interface, or *n* to leave it disabled:

```
Enable IP interface? [y/n]
```

6. The system prompts you to configure another interface:

```
Enter interface number: (1-128)
```

Repeat the steps in this section until all IP interfaces have been configured. When all interfaces have been configured, press <Enter> without specifying any interface number.

## Loopback Interfaces

A loopback interface provides an IP address, but is not otherwise associated with a physical port or network entity. Essentially, it is a virtual interface that is perceived as being “always available” for higher-layer protocols to use and advertise to the network, regardless of other connectivity.

Loopback interfaces improve switch access, increase reliability, security, and provide greater flexibility in Layer 3 network designs. They can be used for many different purposes, but are most commonly for management IP addresses, router IDs for various protocols, and persistent peer IDs for neighbor relationships.

In IBM N/OS 7.8, loopback interfaces have been expanded for use with routing protocols such as OSPF and BGP. Loopback interfaces can also be specified as the source IP address for syslog, SNMP, RADIUS, TACACS+, NTP, and router IDs.

Loopback interfaces must be configured before they can be used in other features. Up to five loopback interfaces are currently supported. They can be configured using the following commands:

```
VFSM(config)# interface loopback <1-5>
VFSM(config-ip-loopback)# [no] ip address <IPv4 address> <mask> enable
VFSM(config-ip-loopback)# exit
```

## Using Loopback Interfaces for Source IP Addresses

The switch can use loopback interfaces to set the source IP addresses for a variety of protocols. This assists in server security, as the server for each protocol can be configured to accept protocol packets only from the expected loopback address block. It may also make it easier to locate or process protocol information, since packets have the source IP address of the loopback interface, rather than numerous egress interfaces.

Configured loopback interfaces can be applied to the following protocols:

- Syslogs

```
VFSM(config)# logging source-interface loopback <1-5>
```

- SNMP traps

```
VFSM(config)# snmp-server trap-source loopback <1-5>
```

- RADIUS

```
VFSM(config)# ip radius source-interface loopback <1-5>
```

- TACACS+

```
VFSM(config)# ip tacacs source-interface loopback <1-5>
```

- NTP

```
VFSM(config)# ntp source loopback <1-5>
```

## Loopback Interface Limitation

- ARP is not supported. Loopback interfaces will ignore ARP requests.
- Loopback interfaces cannot be assigned to a VLAN.

## Default Gateways

1. At the prompt, select an IP default gateway for configuration, or skip default gateway configuration:

```
IP default gateways:
Enter default gateway number: (1-132)
```

Enter the number for the IP default gateway to be configured. To skip default gateway configuration, press <Enter> without typing a gateway number and go to [“IP Routing” on page 63](#).

2. At the prompt, enter the IPv4 address for the selected default gateway:

```
Current IP address:      0.0.0.0
Enter new IP address:
```

Enter the IPv4 address in dotted decimal notation, or press <Enter> without specifying an address to accept the current setting.

3. At the prompt, enter *y* to enable the default gateway, or *n* to leave it disabled:

```
Enable default gateway? [y/n]
```

4. The system prompts you to configure another default gateway:

```
Enter default gateway number: (1-132)
```

Repeat the steps in this section until all default gateways have been configured. When all default gateways have been configured, press <Enter> without specifying any number.

## IP Routing

When IP interfaces are configured for the various IP subnets attached to your switch, IP routing between them can be performed entirely within the switch. This eliminates the need to send inter-subnet communication to an external router device. Routing on more complex networks, where subnets may not have a direct presence on the VFSM, can be accomplished through configuring static routes or by letting the switch learn routes dynamically.

This part of the Setup program prompts you to configure the various routing parameters.

At the prompt, enable or disable forwarding for IP Routing:

`Enable IP forwarding? [y/n]`

Enter `y` to enable IP forwarding. To disable IP forwarding, enter `n`. To keep the current setting, press `<Enter>`.

---

## Setup Part 5: Final Steps

1. When prompted, decide whether to restart Setup or continue:

`Would you like to run from top again? [y/n]`

Enter `y` to restart the Setup utility from the beginning, or `n` to continue.

2. When prompted, decide whether you wish to review the configuration changes:

`Review the changes made? [y/n]`

Enter `y` to review the changes made during this session of the Setup utility. Enter `n` to continue without reviewing the changes. We recommend that you review the changes.

3. Next, decide whether to apply the changes at the prompt:

`Apply the changes? [y/n]`

Enter `y` to apply the changes, or `n` to continue without applying. Changes are normally applied.

4. At the prompt, decide whether to make the changes permanent:

`Save changes to flash? [y/n]`

Enter `y` to save the changes to flash. Enter `n` to continue without saving the changes. Changes are normally saved at this point.

5. If you do not apply or save the changes, the system prompts whether to abort them:

`Abort all changes? [y/n]`

Enter `y` to discard the changes. Enter `n` to return to the “Apply the changes?” prompt.

**Note:** After initial configuration is complete, it is recommended that you change the default passwords as shown in [“Changing the Switch Passwords” on page 70](#).

---

## Optional Setup for Telnet Support

**Note:** This step is optional. Perform this procedure only if you are planning on connecting to the VFSM through a remote Telnet connection.

Telnet is enabled by default. To change the setting, use the following command:

```
VFSM(config)# no access telnet
```



# **Part 2: Securing the Switch**



---

## Chapter 3. Securing Administration

This chapter discusses different methods of securing local and remote administration on the Virtual Fabric Switch Module (VFSM):

- [“Changing the Switch Passwords” on page 70](#)
- [“Secure Shell and Secure Copy” on page 71](#)
- [“End User Access Control” on page 76](#)
- [“Protected Mode” on page 80](#)

---

## Changing the Switch Passwords

It is recommended that you change the administrator and user passwords after initial configuration and as regularly as required under your network security policies.

To change the administrator password, you must login using the administrator password.

**Note:** If you forget your administrator password, call your technical support representative for help using the password fix-up mode.

You can also change the default user names, if desired. The user name length can be up to 64 characters.

## Changing the Default Administrator Password

The administrator has complete access to all menus, information, and configuration commands, including the ability to change both the user and administrator passwords.

The default password for the administrator account is `admin`. To change the default password, follow this procedure:

1. Connect to the switch and log in using the `admin` password.
2. Use the following command to change the administrator password:

```
VFSM(config)# access user administrator-password <password>
```

## Changing the Default User Password

The user login has limited control of the switch. Through a user account, you can view switch information and statistics, but you can't make configuration changes.

The default password for the user account is `user`. This password can be changed from the user account. The administrator can change all passwords, as shown in the following procedure.

1. Connect to the switch and log in using the `admin` password.
2. Use the following command to change the user password:

```
VFSM(config)# access user user-password <password>
```

---

## Secure Shell and Secure Copy

Because using Telnet does not provide a secure connection for managing a VFSM, Secure Shell (SSH) and Secure Copy (SCP) features have been included for VFSM management. SSH and SCP use secure tunnels to encrypt and secure messages between a remote administrator and the switch.

**SSH** is a protocol that enables remote administrators to log securely into the VFSM over a network to execute management commands.

**SCP** is typically used to copy files securely from one machine to another. SCP uses SSH for encryption of data on the network. On a VFSM, SCP is used to download and upload the switch configuration via secure channels.

Although SSH and SCP are disabled by default, enabling and using these features provides the following benefits:

- Identifying the administrator using Name/Password
- Authentication of remote administrators
- Authorization of remote administrators
- Determining the permitted actions and customizing service for individual administrators
- Encryption of management messages
- Encrypting messages between the remote administrator and switch
- Secure copy support

IBM Networking OS implements the SSH version 2.0 standard and is confirmed to work with SSH version 2.0-compliant clients such as the following:

- OpenSSH\_5.4p1 for Linux
- Secure CRT Version 5.0.2 (build 1021)
- Putty SSH release 0.60

## Configuring SSH/SCP Features on the Switch

SSH and SCP are disabled by default. To change the setting, using the following procedures.

### To Enable or Disable the SSH Feature

Begin a Telnet session from the console port and enter the following commands:

VFSM(config)# ssh enable	<i>(Turn SSH on)</i>
VFSM(config)# no ssh enable	<i>(Turn SSH off)</i>

### To Enable or Disable SCP Apply and Save

Enter the following commands from the switch CLI to enable the SCP `putcfg_apply` and `putcfg_apply_save` commands:

VFSM(config)# [no] ssh scp-enable
-----------------------------------

## Configuring the SCP Administrator Password

To configure the SCP-only administrator password, enter the following command (the default password is admin):

```
VFSM(config)# [no] ssh scp-password
Changing SCP-only Administrator password; validation required...
Enter current administrator password: <password>
Enter new SCP-only administrator password: <new password>
Re-enter new SCP-only administrator password: <new password>
New SCP-only administrator password accepted.
```

## Using SSH and SCP Client Commands

This section shows the format for using some common client commands.

### To Log In to the Switch from the Client

Syntax:

```
>> ssh [-4|-6] <switch IP address>
      -or-
>> ssh [-4|-6] <login name>@<switch IP address>
```

**Note:** The -4 option (the default) specifies that an IPv4 switch address will be used. The -6 option specifies IPv6.

Example:

```
>> ssh scpadmin@205.178.15.157
```

### To Copy the Switch Configuration File to the SCP Host

Syntax:

```
>> scp [-4|-6] <username>@<switch IP address>:getcfg <local filename>
```

Example:

```
>> scp scpadmin@205.178.15.157:getcfg ad4.cfg
```

### To Load a Switch Configuration File from the SCP Host

Syntax:

```
>> scp [-4|-6] <local filename> <username>@<switch IP address>:putcfg
```

Example:

```
>> scp ad4.cfg scpadmin@205.178.15.157:putcfg
```

## To Apply and Save the Configuration

When loading a configuration file to the switch, the `apply` and `save` commands are still required, in order for the configuration commands to take effect. The `apply` and `save` commands may be entered manually on the switch, or by using SCP commands.

Syntax:

```
>> scp [-4|-6] <local filename> <username>@<switch IP address>:putcfg_apply
>> scp [-4|-6] <local filename> <username>@<switch IP address>:putcfg_apply_save
```

Example:

```
>> scp ad4.cfg scpadmin@205.178.15.157:putcfg_apply
>> scp ad4.cfg scpadmin@205.178.15.157:putcfg_apply_save
```

- The CLI `diff` command is automatically executed at the end of `putcfg` to notify the remote client of the difference between the new and the current configurations.
- `putcfg_apply` runs the `apply` command after the `putcfg` is done.
- `putcfg_apply_save` saves the new configuration to the flash after `putcfg_apply` is done.
- The `putcfg_apply` and `putcfg_apply_save` commands are provided because extra `apply` and `save` commands are usually required after a `putcfg`; however, an SCP session is not in an interactive mode.

## To Copy the Switch Image and Boot Files to the SCP Host

Syntax:

```
>> scp [-4|-6] <username>@<switch IP address>:getimg1 <local filename>
>> scp [-4|-6] <username>@<switch IP address>:getimg2 <local filename>
>> scp [-4|-6] <username>@<switch IP address>:getboot <local filename>
```

Example:

```
>> scp scpadmin@205.178.15.157:getimg1 6.1.0_os.img
```

## To Load Switch Configuration Files from the SCP Host

Syntax:

```
>> scp [-4|-6] <local filename> <username>@<switch IP address>:putimg1
>> scp [-4|-6] <local filename> <username>@<switch IP address>:putimg2
>> scp [-4|-6] <local filename> <username>@<switch IP address>:putboot
```

Example:

```
>> scp 6.1.0_os.img scpadmin@205.178.15.157:putimg1
```

## SSH and SCP Encryption of Management Messages

The following encryption and authentication methods are supported for SSH and SCP:

- Server Host Authentication: Client RSA authenticates the switch at the beginning of every connection
- Key Exchange: RSA
- Encryption: 3DES-CBC, DES
- User Authentication: Local password authentication or RADIUS

## Generating RSA Host Key for SSH Access

To support the SSH host feature, an RSA host key is required. The host key is 2048 bits and is used to identify the VFSM.

To configure RSA host, first connect to the VFSM through the console port (commands are not available via external Telnet connection), and enter the following command to generate it manually.

VFSM(config)# ssh generate-host-key	(Generates the host key)
-------------------------------------	--------------------------

When the switch reboots, it will retrieve the host key from the FLASH memory.

**Note:** The switch will perform only one session of key/cipher generation at a time. Thus, an SSH/SCP client will not be able to log in if the switch is performing key generation at that time. Also, key generation will fail if an SSH/SCP client is logging in at that time.

## SSH/SCP Integration with RADIUS Authentication

SSH/SCP is integrated with RADIUS authentication. After the RADIUS server is enabled on the switch, all subsequent SSH authentication requests will be redirected to the specified RADIUS servers for authentication. The redirection is transparent to the SSH clients.

## SSH/SCP Integration with LDAP Authentication

SSH/SCP is integrated with LDAP authentication. After the LDAP server is enabled on the switch, all subsequent SSH authentication requests will be redirected to the specified LDAP servers for authentication. The redirection is transparent to the SSH clients.

## SSH/SCP Integration with TACACS+ Authentication

SSH/SCP is integrated with TACACS+ authentication. After the TACACS+ server is enabled on the switch, all subsequent SSH authentication requests will be redirected to the specified TACACS+ servers for authentication. The redirection is transparent to the SSH clients.

---

## End User Access Control

N/OS allows an administrator to define end user accounts that permit end users to perform operation tasks via the switch CLI commands. Once end user accounts are configured and enabled, the switch requires username/password authentication.

For example, an administrator can assign a user, who can then log into the switch and perform operational commands (effective only until the next switch reboot).

### Considerations for Configuring End User Accounts

- A maximum of 20 user IDs are supported on the switch.
- N/OS supports end user support for Console, Telnet, BBI, and SSHv2 access to the switch.
- If RADIUS authentication is used, the user password on the Radius server will override the user password on the VFSM. Also note that the password change command modifies only the user switch password on the switch and has no effect on the user password on the Radius server. Radius authentication and user password cannot be used concurrently to access the switch.
- Passwords can be up to 128 characters in length for TACACS, RADIUS, Telnet, SSH, Console, and Web access.

### Strong Passwords

The administrator can require use of Strong Passwords for users to access the VFSM. Strong Passwords enhance security because they make password guessing more difficult.

The following rules apply when Strong Passwords are enabled:

- Minimum length: 8 characters; maximum length: 64 characters
- Must contain at least one uppercase alphabet
- Must contain at least one lowercase alphabet
- Must contain at least one number
- Must contain at least one special character:  
Supported special characters: ! " # % & ' ( ) ; < = > ? [ \ ] ^ \_ { | } ~
- Cannot be same as the username
- No consecutive four characters can be the same as in the old password

When strong password is enabled, users can still access the switch using the old password but will be advised to change to a strong password while attempting to log in.

Strong password requirement can be enabled using the following command:

```
VFSM(config)# access user strong-password enable
```

The administrator can choose the number of days allowed before each password expires. When a strong password expires, the user is allowed to log in one last time (last time) to change the password. A warning provides advance notice for users to change the password.

## User Access Control Menu

The end-user access control commands allow you to configure end-user accounts.

### Setting Up User IDs

Up to 20 user IDs can be configured in the User ID menu.

```
VFSM(config)# access user 1 name <1-8 characters>
VFSM(config)# access user 1 password

Changing user1 password; validation required:
Enter current admin password: <current administrator password>
Enter new user1 password: <new user password>
Re-enter new user1 password: <new user password>
New user1 password accepted.
```

### Defining a User's Access Level

The end user is by default assigned to the user access level (also known as class of service, or CoS). CoS for all user accounts have global access to all resources except for User CoS, which has access to view only resources that the user owns. For more information, see [Table 7 on page 84](#).

To change the user's level, enter the command:

```
VFSM(config)# access user 1 level {user|operator|administrator}
```

### Validating a User's Configuration

```
VFSM# show access user uid 1
```

### Enabling or Disabling a User

An end user account must be enabled before the switch recognizes and permits login under the account. Once enabled, the switch requires any user to enter both username and password.

```
VFSM(config)# [no] access user 1 enable
```

### Locking Accounts

To protect the switch from unauthorized access, the account lockout feature can be enabled. By default, account lockout is disabled. To enable this feature, ensure the strong password feature is enabled (See ["Strong Passwords" on page 76](#)). Then use the following command:

```
VFSM(config)# access user strong-password lockout
```

After multiple failed login attempts, the switch locks the user account if lockout has been enabled on the switch.

## Re-enabling Locked Accounts

The administrator can re-enable a locked account by reloading the switch or by using the following command:

```
VFSM(config)# access user strong-password clear local user lockout username  
<user name>
```

However, the above command cannot be used to re-enable an account disabled by the administrator.

To re-enable all locked accounts, use the following command:

```
VFSM(config)# access user strong-password clear local user lockout all
```

## Listing Current Users

The `cur` command displays defined user accounts and whether or not each user is currently logged into the switch.

```
VFSM# show access user

Usernames:
  user      - Enabled - offline
  oper      - Disabled - offline
  admin     - Always Enabled - online 1 session

Current User ID table:
  1: name jane      , ena, cos user      , password valid, online
  2: name john      , ena, cos user      , password valid, online
```

## Logging In to an End User Account

Once an end user account is configured and enabled, the user can login to the switch, using the username/password combination. The level of switch access is determined by the CoS established for the end user account.

---

## Protected Mode

Protected Mode settings allow the switch administrator to block the management module from making configuration changes that affect switch operation. The switch retains control over those functions.

The following management module functions are disabled when Protected Mode is turned on:

- External Ports: Enabled/Disabled
- External management over all ports: Enabled/Disabled
- Restore Factory Defaults
- New Static IP Configuration

In this release, configuration of the functions listed above are restricted to the local switch when you turn Protected Mode on. In future releases, individual control over each function may be added.

**Note:** Before you turn Protected Mode on, make sure that external management (Telnet) access to one of the switch's IP interfaces is enabled.

Use the following command to turn Protected Mode on:

```
VFSM(config)# protected-mode enable
```

If you lose access to the switch through the external ports, use the console port to connect directly to the switch, and configure an IP interface with Telnet access.

## Stacking Mode

When the switch is in stacking mode, Protected Mode is automatically enabled for three of the four Protected Mode functions, and the following module functions are disabled:

- External Ports (Enabled)
- External management over all ports (Enabled)
- Restore Factory Defaults

Stack members and stack Master can get their IP addresses from the advanced management module (AMM). Stack can be managed using external ports or using the AMM management port.

If required, the functionality of new static IP configuration can also be disabled by turning off Protected Mode (`VFSM(config)# no protected-mode`) and turning it back on (`VFSM(config)# protected-mode enable`).

---

## Chapter 4. Authentication & Authorization Protocols

Secure switch management is needed for environments that perform significant management functions across the Internet. The following are some of the functions for secured IPv4 management and device access:

- [“RADIUS Authentication and Authorization” on page 82](#)
- [“TACACS+ Authentication” on page 86](#)
- [“LDAP Authentication and Authorization” on page 90](#)

**Note:** IBM Networking OS 7.8 does not support IPv6 for RADIUS, TACACS+ or LDAP.

---

## RADIUS Authentication and Authorization

IBM N/OS supports the RADIUS (Remote Authentication Dial-in User Service) method to authenticate and authorize remote administrators for managing the switch. This method is based on a client/server model. The Remote Access Server (RAS)—the switch—is a client to the back-end database server. A remote user (the remote administrator) interacts only with the RAS, not the back-end server and database.

RADIUS authentication consists of the following components:

- A protocol with a frame format that utilizes UDP over IP (based on RFC 2138 and 2866)
- A centralized server that stores all the user authorization information
- A client, in this case, the switch

The VFSM—acting as the RADIUS client—communicates to the RADIUS server to authenticate and authorize a remote administrator using the protocol definitions specified in RFC 2138 and 2866. Transactions between the client and the RADIUS server are authenticated using a shared key that is not sent over the network. In addition, the remote administrator passwords are sent encrypted between the RADIUS client (the switch) and the back-end RADIUS server.

### How RADIUS Authentication Works

1. Remote administrator connects to the switch and provides user name and password.
2. Using Authentication/Authorization protocol, the switch sends request to authentication server.
3. Authentication server checks the request against the user ID database.
4. Using RADIUS protocol, the authentication server instructs the switch to grant or deny administrative access.

### Configuring RADIUS on the Switch

Use the following procedure to configure Radius authentication on your VFSM. For more information, see ["RADIUS Server Configuration Notes" on page 525](#).

1. Turn RADIUS authentication on, then configure the Primary and Secondary RADIUS servers.

```
VFSM(config)# radius-server primary-host 10.10.1.1
VFSM(config)# radius-server secondary-host 10.10.1.2
```

**Note:** You can use a configured loopback address as the source address so the RADIUS server accepts requests only from the expected loopback address block. Use the following command to specify the loopback interface:

```
VFSM(config)# ip radius source-interface loopback <1-5>
```

2. Configure the RADIUS secret.

```
VFSM(config)# radius-server primary-host 10.10.1.1 key <1-32 character secret>
VFSM(config)# radius-server secondary-host 10.10.1.2 key <1-32 character secret>
VFSM(config)# radius-server enable
```



**CAUTION:**

If you configure the RADIUS secret using any method other than through the console port or management module, the secret may be transmitted over the network as clear text.

3. If desired, you may change the default UDP port number used to listen to RADIUS.

The well-known port for RADIUS is 1645.

```
VFSM(config)# radius-server port <UDP port number>
```

4. Configure the number retry attempts for contacting the RADIUS server, and the timeout period.

```
VFSM(config)# radius-server retransmit 3
VFSM(config)# radius-server timeout 5
```

## RADIUS Authentication Features in IBM N/OS

N/OS supports the following RADIUS authentication features:

- Supports RADIUS client on the switch, based on the protocol definitions in RFC 2138 and RFC 2866.
- Allows a RADIUS secret password of up to 32 characters.
- Supports *secondary authentication server* so that when the primary authentication server is unreachable, the switch can send client authentication requests to the secondary authentication server. Use the following command to show the currently active RADIUS authentication server:

```
VFSM# show radius-server
```

- Supports user-configurable RADIUS server retry and time-out values:
  - Time-out value = 1-10 seconds
  - Retries = 1-3

The switch will time out if it does not receive a response from the RADIUS server within 1-10 seconds. The switch automatically retries connecting to the RADIUS server 1-3 times before it declares the server down.

- Supports user-configurable RADIUS application port. The default is UDP port 1645. UDP port 1812, based on RFC 2138, is also supported.
- Allows network administrator to define privileges for one or more specific users to access the switch at the RADIUS user database.

## Switch User Accounts

The user accounts listed in [Table 7](#) can be defined in the RADIUS server dictionary file.

Table 7. User Access Levels

User Account	Description and Tasks Performed	Password
User	The User has no direct responsibility for switch management. He/she can view all switch status information and statistics but cannot make any configuration changes to the switch.	user
Operator	In addition to User capabilities, the Operator has limited switch management access, including the ability to make temporary, operational configuration changes to some switch features, and to reset switch ports (other than management ports).	oper
Administrator	The super-user Administrator has complete access to all menus, information, and configuration commands on the switch, including the ability to change both the user and administrator passwords.	admin

## RADIUS Attributes for IBM N/OS User Privileges

When the user logs in, the switch authenticates his/her level of access by sending the RADIUS access request, that is, the client authentication request, to the RADIUS authentication server.

If the remote user is successfully authenticated by the authentication server, the switch will verify the *privileges* of the remote user and authorize the appropriate access. The administrator has two options: to allow *backdoor* access via Telnet, SSH, HTTP, or HTTPS; to allow *secure backdoor* access via Telnet, SSH, or BBI. Backdoor and secure backdoor provides access to the switch when the RADIUS servers cannot be reached.

The default VFSM setting for backdoor and secure backdoor access is `disabled`. Backdoor and secure backdoor access is always enabled on the console port.

Irrespective of backdoor/secure backdoor being enabled or not, you can always access the switch via the console port by using `noradius` as radius username. You can then enter the username and password configured on the switch. If you are trying to connect via SSH/Telnet/HTTP/HTTPS (not console port), there are two possibilities:

- Backdoor is enabled: The switch acts like it is connecting via console.
- Secure backdoor is enabled: You must enter the username: `noradius`. The switch checks if RADIUS server is reachable. If it is reachable, then you must authenticate via remote authentication server. Only if RADIUS server is not reachable, you will be prompted for local user/password to be authenticated against these local credentials.

All user privileges, other than those assigned to the Administrator, have to be defined in the RADIUS dictionary. RADIUS attribute 6 which is built into all RADIUS servers defines the administrator. The file name of the dictionary is RADIUS vendor-dependent. The following RADIUS attributes are defined for N/OS user privileges levels:

*Table 8. IBM N/OS-proprietary Attributes for RADIUS*

<b>User Name/Access</b>	<b>User-Service-Type</b>	<b>Value</b>
User	<i>Vendor-supplied</i>	255
Operator	<i>Vendor-supplied</i>	252
Admin	<i>Vendor-supplied</i>	6

---

## TACACS+ Authentication

N/OS supports authentication, authorization, and accounting with networks using the Cisco Systems TACACS+ protocol. The VFSM functions as the Network Access Server (NAS) by interacting with the remote client and initiating authentication and authorization sessions with the TACACS+ access server. The remote user is defined as someone requiring management access to the VFSM either through a data or management port.

TACACS+ offers the following advantages over RADIUS:

- TACACS+ uses TCP-based connection-oriented transport; whereas RADIUS is UDP-based. TCP offers a connection-oriented transport, while UDP offers best-effort delivery. RADIUS requires additional programmable variables such as re-transmit attempts and time-outs to compensate for best-effort transport, but it lacks the level of built-in support that a TCP transport offers.
- TACACS+ offers full packet encryption whereas RADIUS offers password-only encryption in authentication requests.
- TACACS+ separates authentication, authorization and accounting.

## How TACACS+ Authentication Works

TACACS+ works much in the same way as RADIUS authentication as described on [page 82](#).

1. Remote administrator connects to the switch and provides user name and password.
2. Using Authentication/Authorization protocol, the switch sends request to authentication server.
3. Authentication server checks the request against the user ID database.
4. Using TACACS+ protocol, the authentication server instructs the switch to grant or deny administrative access.

During a session, if additional authorization checking is needed, the switch checks with a TACACS+ server to determine if the user is granted permission to use a particular command.

## TACACS+ Authentication Features in IBM N/OS

Authentication is the action of determining the identity of a user, and is generally done when the user first attempts to log in to a device or gain access to its services. N/OS supports ASCII inbound login to the device. PAP, CHAP and ARAP login methods, TACACS+ change password requests, and one-time password authentication are not supported.

### Authorization

Authorization is the action of determining a user's privileges on the device, and usually takes place after authentication.

The default mapping between TACACS+ authorization levels and N/OS management access levels is shown in [Table 9](#). The authorization levels listed in this table must be defined on the TACACS+ server.

Table 9. Default TACACS+ Authorization Levels

N/OS User Access Level	TACACS+ Level
user	0
oper	3
admin	6

Alternate mapping between TACACS+ authorization levels and N/OS management access levels is shown in [Table 10](#). Use the `/cfg/sys/tacacs/cmap ena` command to use the alternate TACACS+ authorization levels.

```
VFSM(config)# tacacs-server privilege-mapping
```

Table 10. Alternate TACACS+ Authorization Levels

N/OS User Access Level	TACACS+ Level
user	0–1
oper	6–8
admin	14–15

You can customize the mapping between TACACS+ privilege levels and VFSM management access levels. Use the following command to manually map each TACACS+ privilege level (0-15) to a corresponding VFSM management access level: `VFSM(config)# tacacs-server user-mapping`

If the remote user is successfully authenticated by the authentication server, the switch verifies the *privileges* of the remote user and authorizes the appropriate access. The administrator has an option to allow *backdoor* access via Telnet (`VFSM(config)# tacacs-server backdoor`). The default value for Telnet access is disabled. The administrator also can enable *secure backdoor* (`VFSM(config)# tacacs-server secure-backdoor`), to allow access if both the primary and the secondary TACACS+ servers fail to respond.

**Note:** To obtain the TACACS+ backdoor password for your switch, contact your IBM Service and Support line.

## Accounting

Accounting is the action of recording a user's activities on the device for the purposes of billing and/or security. It follows the authentication and authorization actions. If the authentication and authorization is not performed via TACACS+, there are no TACACS+ accounting messages sent out.

You can use TACACS+ to record and track software login access, configuration changes, and interactive commands.

The VFSM supports the following TACACS+ accounting attributes:

- protocol (console/telnet/ssh/http)

- start\_time
- stop\_time
- elapsed\_time
- disc-cause

**Note:** When using the Browser-Based Interface, the TACACS+ Accounting Stop records are sent only if the **Quit** button on the browser is clicked.

## Command Authorization and Logging

When TACACS+ Command Authorization is enabled

(VFSM(config)# tacacs-server command-authorization), IBM N/OS configuration commands are sent to the TACACS+ server for authorization. When TACACS+ Command Logging is enabled (VFSM(config)# tacacs-server command-logging), IBM N/OS configuration commands are logged on the TACACS+ server.

The following examples illustrate the format of N/OS commands sent to the TACACS+ server:

```
authorization request, cmd=cfgtree, cmd-arg=/cfg/l3/if
accounting request, cmd=/cfg/l3/if, cmd-arg=1
authorization request, cmd=cfgtree, cmd-arg=/cfg/l3/if/ena
accounting request, cmd=/cfg/l3/if/ena
authorization request, cmd=cfgtree, cmd-arg=/cfg/l3/if/addr
accounting request, cmd=/cfg/l3/if/addr, cmd-arg=10.90.90.91

authorization request, cmd=apply
accounting request, cmd=apply
```

The following rules apply to TACACS+ command authorization and logging:

- Only commands from a Console, Telnet, or SSH connection are sent for authorization and logging. SNMP, BBI, or file-copy commands (for example, TFTP or sync) are not sent.
- Only leaf-level commands are sent for authorization and logging. For example, VFSM(config)# is not sent, but VFSM(config)# tacacs-server command-logging is sent.
- The full path of each command is sent for authorization and logging. For example: VFSM(config)# tacacs-server command-logging
- Command arguments are not sent for authorization.
- Only executed commands are logged.
- Invalid commands are checked by N/OS, and are not sent for authorization or logging.
- Authorization is performed on each leaf-level command separately. If the user issues multiple commands at once, each command is sent separately as a full path.
- Only the following global commands are sent for authorization and logging:

```
apply
diff
ping
revert
save
telnet
traceroute
```

## TACACS+ Password Change

N/OS supports TACACS+ password change. When enabled, users can change their passwords after successful TACACS+ authorization. Use the command `VFSM(config)# [no] tacacs-server password-change` to enable or disable this feature.

Use the following commands to change the password for the primary and secondary TACACS+ servers:

<code>VFSM(config)# tacacs-server chpassp</code>	<i>(Change primary TACACS+ password)</i>
<code>VFSM(config)# tacacs-server chpass</code>	<i>(Change secondary TACACS+ password)</i>

## Configuring TACACS+ Authentication on the Switch

1. Turn TACACS+ authentication on, then configure the Primary and Secondary TACACS+ servers, and enable TACACS authentication.

<code>VFSM(config)# tacacs-server primary-host 10.10.1.1</code>	<i>(Enter primary server IPv4 address)</i>
<code>VFSM(config)# tacacs-server primary-host mgt-port</code>	
<code>VFSM(config)# tacacs-server secondary-host 10.10.1.1</code>	<i>(Enter secondary server IPv4 address)</i>
<code>VFSM(config)# tacacs-server secondary-host data-port</code>	
<code>VFSM(config)# tacacs-server enable</code>	

2. Configure the TACACS+ secret and second secret.

<code>VFSM(config)# tacacs-server primary-host 10.10.1.1 key &lt;1-32 character secret&gt;</code>
<code>VFSM(config)# tacacs-server secondary-host 10.10.1.2 key &lt;1-32 character secret&gt;</code>



### CAUTION:

**If you configure the TACACS+ secret using any method other than a direct console connection or through a secure management module connection, the secret may be transmitted over the network as clear text.**

3. If desired, you may change the default TCP port number used to listen to TACACS+. The well-known port for TACACS+ is 49.

<code>VFSM(config)# tacacs-server port &lt;TCP port number&gt;</code>
---

4. Configure the number of retry attempts, and the timeout period.

<code>VFSM(config)# tacacs-server retransmit 3</code>
<code>VFSM(config)# tacacs-server timeout 5</code>

5. Configure custom privilege-level mapping (optional).

<code>VFSM(config)# tacacs-server user-mapping 2 user</code>
<code>VFSM(config)# tacacs-server user-mapping 3 user</code>
<code>VFSM(config)# tacacs-server user-mapping 4 user</code>
<code>VFSM(config)# tacacs-server user-mapping 5 oper</code>

---

## LDAP Authentication and Authorization

N/OS supports the LDAP (Lightweight Directory Access Protocol) method to authenticate and authorize remote administrators to manage the switch. LDAP is based on a client/server model. The switch acts as a client to the LDAP server. A remote user (the remote administrator) interacts only with the switch, not the back-end server and database.

LDAP authentication consists of the following components:

- A protocol with a frame format that utilizes TCP over IP
- A centralized server that stores all the user authorization information
- A client, in this case, the switch

Each entry in the LDAP server is referenced by its Distinguished Name (DN). The DN consists of the user-account name concatenated with the LDAP domain name. If the user-account name is John, the following is an example DN:

```
uid=John,ou=people,dc=domain,dc=com
```

### Configuring the LDAP Server

VFSM user groups and user accounts must reside within the same domain. On the LDAP server, configure the domain to include VFSM user groups and user accounts, as follows:

- User Accounts:  
Use the *uid* attribute to define each individual user account.
- User Groups:  
Use the *members* attribute in the *groupOfNames* object class to create the user groups. The first word of the common name for each user group must be equal to the user group names defined in the VFSM, as follows:
  - admin
  - oper
  - user

### Configuring LDAP Authentication on the Switch

1. Turn LDAP authentication on, then configure the Primary and Secondary LDAP servers.

```
VFSM(config)# ldap-server primary-host 10.10.1.1 (Enter primary server IPv4 address)
VFSM(config)# ldap-server primary-host 10.10.1.2 (Enter secondary server IPv4 address)
```

2. Configure the domain name.

```
VFSM(config)# ldap-server domain <ou=people,dc=my-domain,dc=com>
```

3. If desired, you may change the default TCP port number used to listen to LDAP. The well-known port for LDAP is 389.

```
VFSM(config)# ldap-server port <1-65000>
```

4. Configure the number of retry attempts for contacting the LDAP server, and the timeout period.

VFSM(config)# ldap-server retransmit 3	<i>(server retries)</i>
VFSM(config)# ldap-server timeout 10	<i>(Enter the timeout period in seconds)</i>



---

## Chapter 5. 802.1X Port-Based Network Access Control

Port-Based Network Access control provides a means of authenticating and authorizing devices attached to a LAN port that has point-to-point connection characteristics. It prevents access to ports that fail authentication and authorization. This feature provides security to ports of the Virtual Fabric Switch Module (VFSM) that connect to blade servers.

The following topics are discussed in this section:

- [“Extensible Authentication Protocol over LAN” on page 94](#)
- [“EAPoL Authentication Process” on page 95](#)
- [“EAPoL Port States” on page 97](#)
- [“Guest VLAN” on page 98](#)
- [“Supported RADIUS Attributes” on page 99](#)
- [“EAPoL Configuration Guidelines” on page 101](#)

---

## Extensible Authentication Protocol over LAN

IBM Networking OS can provide user-level security for its ports using the IEEE 802.1X protocol, which is a more secure alternative to other methods of port-based network access control. Any device attached to an 802.1X-enabled port that fails authentication is prevented access to the network and denied services offered through that port.

The 802.1X standard describes port-based network access control using Extensible Authentication Protocol over LAN (EAPoL). EAPoL provides a means of authenticating and authorizing devices attached to a LAN port that has point-to-point connection characteristics and of preventing access to that port in cases of authentication and authorization failures.

EAPoL is a client-server protocol that has the following components:

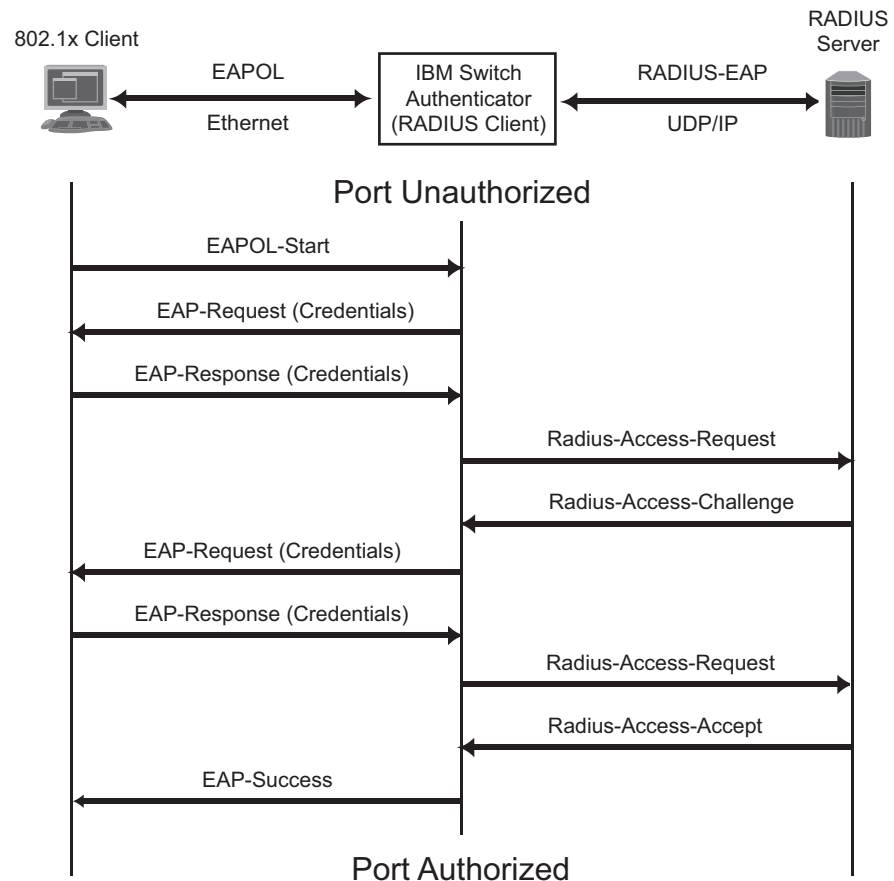
- **Supplicant or Client**  
The Supplicant is a device that requests network access and provides the required credentials (user name and password) to the Authenticator and the Authenticator Server.
- **Authenticator**  
The Authenticator enforces authentication and controls access to the network. The Authenticator grants network access based on the information provided by the Supplicant and the response from the Authentication Server. The Authenticator acts as an intermediary between the Supplicant and the Authentication Server: requesting identity information from the client, forwarding that information to the Authentication Server for validation, relaying the server's responses to the client, and authorizing network access based on the results of the authentication exchange. The VFSM acts as an Authenticator.
- **Authentication Server**  
The Authentication Server validates the credentials provided by the Supplicant to determine if the Authenticator should grant access to the network. The Authentication Server may be co-located with the Authenticator. The VFSM relies on external RADIUS servers for authentication.

Upon a successful authentication of the client by the server, the 802.1X-controlled port transitions from unauthorized to authorized state, and the client is allowed full access to services through the port. When the client sends an EAP-Logoff message to the authenticator, the port will transition from authorized to unauthorized state.

## EAPoL Authentication Process

The clients and authenticators communicate using Extensible Authentication Protocol (EAP), which was originally designed to run over PPP, and for which the IEEE 802.1X Standard has defined an encapsulation method over Ethernet frames, called EAP over LAN (EAPoL). Figure 2 shows a typical message exchange initiated by the client.

Figure 2. Authenticating a Port Using EAPoL



---

## EAPoL Message Exchange

During authentication, EAPoL messages are exchanged between the client and the VFSM authenticator, while RADIUS-EAP messages are exchanged between the VFSM authenticator and the RADIUS server.

Authentication is initiated by one of the following methods:

- The VFSM authenticator sends an EAP-Request/Identity packet to the client
- The client sends an EAPoL-Start frame to the VFSM authenticator, which responds with an EAP-Request/Identity frame.

The client confirms its identity by sending an EAP-Response/Identity frame to the VFSM authenticator, which forwards the frame encapsulated in a RADIUS packet to the server.

The RADIUS authentication server chooses an EAP-supported authentication algorithm to verify the client's identity, and sends an EAP-Request packet to the client via the VFSM authenticator. The client then replies to the RADIUS server with an EAP-Response containing its credentials.

Upon a successful authentication of the client by the server, the 802.1X-controlled port transitions from unauthorized to authorized state, and the client is allowed full access to services through the controlled port. When the client later sends an EAPoL-Logoff message to the VFSM authenticator, the port transitions from authorized to unauthorized state.

If a client that does not support 802.1X connects to an 802.1X-controlled port, the VFSM authenticator requests the client's identity when it detects a change in the operational state of the port. The client does not respond to the request, and the port remains in the unauthorized state.

**Note:** When an 802.1X-enabled client connects to a port that is not 802.1X-controlled, the client initiates the authentication process by sending an EAPoL-Start frame. When no response is received, the client retransmits the request for a fixed number of times. If no response is received, the client assumes the port is in authorized state, and begins sending frames, even if the port is unauthorized.

---

## EAPoL Port States

The state of the port determines whether the client is granted access to the network, as follows:

- **Unauthorized**  
While in this state the port discards all ingress and egress traffic except EAP packets.
- **Authorized**  
When the client is successfully authenticated, the port transitions to the authorized state allowing all traffic to and from the client to flow normally.
- **Force Unauthorized**  
You can configure this state that denies all access to the port.
- **Force Authorized**  
You can configure this state that allows full access to the port.

---

## Guest VLAN

The guest VLAN provides limited access to unauthenticated ports. The guest VLAN can be configured from the following menu:

```
VFSM(config)# dot1x guest-vlan ?
```

Client ports that have not received an EAPOL response are placed into the Guest VLAN, if one is configured on the switch. Once the port is authenticated, it is moved from the Guest VLAN to its configured VLAN.

When Guest VLAN enabled, the following considerations apply while a port is in the unauthenticated state:

- The port is placed in the guest VLAN.
- The Port VLAN ID (PVID)/Native VLAN is changed to the Guest VLAN ID.
- Port tagging/trunk mode is disabled on the port.

## Supported RADIUS Attributes

The 802.1X Authenticator relies on external RADIUS servers for authentication with EAP. [Table 11](#) lists the RADIUS attributes that are supported as part of RADIUS-EAP authentication based on the guidelines specified in Annex D of the 802.1X standard and RFC 3580.

Table 11. Support for RADIUS Attributes

#	Attribute	Attribute Value	A-R	A-A	A-C	A-R
1	User-Name	The value of the Type-Data field from the supplicant's EAP-Response/Identity message. If the Identity is unknown (i.e. Type-Data field is zero bytes in length), this attribute will have the same value as the Calling-Station-Id.	1	0-1	0	0
4	NAS-IP-Address	IPv4 address of the authenticator used for Radius communication.	1	0	0	0
5	NAS-Port	Port number of the authenticator port to which the supplicant is attached.	1	0	0	0
24	State	Server-specific value. This is sent unmodified back to the server in an Access-Request that is in response to an Access-Challenge.	0-1	0-1	0-1	0
30	Called-Station-ID	The MAC address of the authenticator encoded as an ASCII string in canonical format, such as 000D5622E3 9F.	1	0	0	0
31	Calling-Station-ID	The MAC address of the supplicant encoded as an ASCII string in canonical format, such as 00034B436206.	1	0	0	0
64	Tunnel-Type	Only VLAN (type 13) is currently supported (for 802.1X RADIUS VLAN assignment). The attribute must be untagged/access-mode (the Tag field must be 0).	0	0-1	0	0
65	Tunnel-Medium-Type	Only 802 (type 6) is currently supported (for 802.1X RADIUS VLAN assignment). The attribute must be untagged/access mode (the Tag field must be 0).	0	0-1	0	0

Table 11. Support for RADIUS Attributes (continued)

#	Attribute	Attribute Value	A-R	A-A	A-C	A-R
81	Tunnel-Private-Group-ID	VLAN ID (1-4094). When 802.1X RADIUS VLAN assignment is enabled on a port, if the RADIUS server includes the tunnel attributes defined in RFC 2868 in the Access-Accept packet, the switch will automatically place the authenticated port in the specified VLAN. Reserved VLANs (such as for management or stacking) may not be specified. The attribute must be untagged/access mode (the Tag field must be 0).	0	0-1	0	0
79	EAP-Message	Encapsulated EAP packets from the supplicant to the authentication server (Radius) and vice-versa. The authenticator relays the decoded packet to both devices.	1+	1+	1+	1+
80	Message-Authenticator	Always present whenever an EAP-Message attribute is also included. Used to integrity-protect a packet.	1	1	1	1
87	NAS-Port-ID	Name assigned to the authenticator port, e.g. Server1_Port3	1	0	0	0
<p><b>Legend:</b> RADIUS Packet Types: A-R (Access-Request), A-A (Access-Accept), A-C (Access-Challenge), A-R (Access-Reject)</p> <p>RADIUS Attribute Support:</p> <ul style="list-style-type: none"> <li>• 0 This attribute MUST NOT be present in a packet.</li> <li>• 0+ Zero or more instances of this attribute MAY be present in a packet.</li> <li>• 0-1 Zero or one instance of this attribute MAY be present in a packet.</li> <li>• 1 Exactly one instance of this attribute MUST be present in a packet.</li> <li>• 1+ One or more of these attributes MUST be present.</li> </ul>						

---

## EAPoL Configuration Guidelines

When configuring EAPoL, consider the following guidelines:

- The 802.1X port-based authentication is currently supported only in point-to-point configurations, that is, with a single supplicant connected to an 802.1X-enabled switch port.
- 802.1X authentication is not supported on ISL ports or on any port that is part of a trunk.
- When 802.1X is enabled, a port has to be in the authorized state before any other Layer 2 feature can be operationally enabled. For example, the STG state of a port is operationally disabled while the port is in the unauthorized state.
- The 802.1X supplicant capability is not supported. Therefore, none of its ports can successfully connect to an 802.1X-enabled port of another device, such as another switch, that acts as an authenticator, unless access control on the remote port is disabled or is configured in forced-authorized mode. For example, if a VFSM is connected to another VFSM, and if 802.1X is enabled on both switches, the two connected ports must be configured in force-authorized mode.
- Unsupported 802.1X attributes include Service-Type, Session-Timeout, and Termination-Action.
- RADIUS accounting service for 802.1X-authenticated devices or users is not currently supported.
- Configuration changes performed using SNMP and the standard 802.1X MIB will take effect immediately.



---

## Chapter 6. Access Control Lists

Access Control Lists (ACLs) are filters that permit or deny traffic for security purposes. They can also be used with QoS to classify and segment traffic in order to provide different levels of service to different traffic types. Each filter defines the conditions that must match for inclusion in the filter, and also the actions that are performed when a match is made.

IBM Networking OS 7.8 supports the following ACLs:

- IPv4 ACLs

Up to 256 ACLs are supported for networks that use IPv4 addressing. IPv4 ACLs are configured using the following CLI command:

```
VFSM(config)# access-control list <IPv4 ACL number>
```

- IPv6 ACLs

Up to 128 ACLs are supported for networks that use IPv6 addressing. IPv6 ACLs are configured using the following CLI command :

```
VFSM(config)# access-control list6 <IPv6 ACL number>
```

**Note:** IPv6 ACLs are not supported on ports where FCoE is enabled on the VFSM (see [“FCoE Initialization Protocol Snooping” on page 271](#)).

- VLAN Maps (VMaps)

Up to 128 VMaps are supported for attaching filters to VLANs rather than ports. See [“VLAN Maps” on page 117](#) for details.

```
VFSM(config)# access-control vmap <vmap number>
```

- Management ACLs (MACLs)

Up to 128 MACLs are supported for filtering traffic toward CPU. MACLs are configured using the following ISCLI command path:

```
VFSM(config)# access-control mac1 <MACL number>
```

---

## Summary of Packet Classifiers

ACLs allow you to classify packets according to a variety of content in the packet header (such as the source address, destination address, source port number, destination port number, and others). Once classified, packet flows can be identified for more processing.

IPv4 ACLs, IPv6 ACLs, and VMaps allow you to classify packets based on the following packet attributes:

- Ethernet header options (for IPv4 ACLs and VMaps only)
  - Source MAC address
  - Destination MAC address
  - VLAN number and mask
  - Ethernet type (ARP, IP, IPv6, MPLS, RARP, etc.)
  - Ethernet Priority (the IEEE 802.1p Priority)
- IPv4 header options (for IPv4 ACLs and VMaps only)
  - Source IPv4 address and subnet mask
  - Destination IPv4 address and subnet mask
  - Type of Service value
  - IP protocol number or name as shown in [Table 12](#):

*Table 12. Well-Known Protocol Types*

Number	Protocol Name
1	icmp
2	igmp
6	tcp
17	udp
89	ospf
112	vrrp

- IPv6 header options (for IPv6 ACLs only)
  - Source IPv6 address and prefix length
  - Destination IPv6 address and prefix length
  - Next Header value
  - Flow Label value
  - Traffic Class value

- TCP/UDP header options (for all ACLs)
  - TCP/UDP application source port as shown in [Table 13](#)

Table 13. Well-Known Application Ports

Port	TCP/UDP Application	Port	TCP/UDP Application	Port	TCP/UDP Application
20	ftp-data	79	finger	179	bgp
21	ftp	80	http	194	irc
22	ssh	109	pop2	220	imap3
23	telnet	110	pop3	389	ldap
25	smtp	111	sunrpc	443	https
37	time	119	nntp	520	rip
42	name	123	ntp	554	rtsp
43	whois	143	imap	1645/1812	Radius
53	domain	144	news	1813	Radius
69	tftp	161	snmp	1985	Accounting
70	gopher	162	snmptrap		hsrp

- TCP/UDP application destination port and mask as shown in [Table 13](#)
- TCP/UDP flag value as shown in [Table 14](#)

Table 14. Well-Known TCP flag values

Flag	Value
URG	0x0020
ACK	0x0010
PSH	0x0008
RST	0x0004
SYN	0x0002
FIN	0x0001

- Packet format (for IPv4 ACLs and VMaps only)
  - Ethernet format (eth2, SNAP, LLC)
  - Ethernet tagging format
  - IP format (IPv4, IPv6)
- Egress port packets (for all ACLs)

---

## Summary of ACL Actions

Once classified using ACLs, the identified packet flows can be processed differently. For each ACL, an *action* can be assigned. The action determines how the switch treats packets that match the classifiers assigned to the ACL. VFSM ACL actions include the following:

- Pass or Drop the packet
- Re-mark the packet with a new DiffServ Code Point (DSCP)
- Re-mark the 802.1p field
- Set the COS queue

---

## Assigning Individual ACLs to a Port

Once you configure an ACL, you must assign the ACL to the appropriate ports. Each port can accept multiple ACLs, and each ACL can be applied for multiple ports. ACLs can be assigned individually, or in groups.

To assign an individual ACLs to a port, use the following commands:

```
VFSM(config)# interface port <port>  
VFSM(config-if)# access-control list <IPv4 ACL number>
```

When multiple ACLs are assigned to a port, higher-priority ACLs are considered first, and their action takes precedence over lower-priority ACLs. ACL order of precedence is discussed in the next section.

To create and assign ACLs in groups, see [“ACL Groups” on page 109](#).

---

## ACL Order of Precedence

When multiple ACLs are assigned to a port, they are evaluated in numeric sequence, based on the ACL number. Lower-numbered ACLs take precedence over higher-numbered ACLs. For example, ACL 1 (if assigned to the port) is evaluated first and has top priority.

If multiple ACLs match the port traffic, only the action of the one with the lowest ACL number is applied. The others are ignored.

The ACL number is the sole factor in determining ACL order of precedence. The order in which ACLs are applied to a port does not affect the order of precedence, nor does the ACL Group number (see [“ACL Groups” on page 109](#)), the order in which an ACL is assigned to an ACL Group, or the order in which the ACL Group is assigned to a port.

If no assigned ACL matches the port traffic, no ACL action is applied.

---

## ACL Groups

To assist in organizing multiple ACLs and assigning them to ports, you can place ACLs into ACL Groups, thereby defining complex traffic profiles. ACLs and ACL Groups can then be assigned on a per-port basis. Any specific ACL can be assigned to multiple ACL Groups, and any ACL or ACL Group can be assigned to multiple ports. If, as part of multiple ACL Groups, a specific ACL is assigned to a port multiple times, only one instance is used. The redundant entries are ignored.

- **Individual ACLs**

The VFSM supports up to 256 ACLs. Each ACL defines one filter rule for matching traffic criteria. Each filter rule can also include an action (permit or deny the packet). For example:

<b>ACL 1:</b> VLAN = 1 SIP = 10.10.10.1 (255.255.255.0) Action = permit
--

- **Access Control List Groups**

An Access Control List Group (ACL Group) is a collection of ACLs. For example:

<b>ACL Group 1</b>
<b>ACL 1:</b> VLAN = 1 SIP = 10.10.10.1 (255.255.255.0) Action = permit
<b>ACL 2:</b> VLAN = 2 SIP = 10.10.10.2 (255.255.255.0) Action = deny
<b>ACL 3:</b> Priority = 7 DIP = 10.10.10.3 (255.255.255.0) Action = permit

ACL Groups organize ACLs into traffic profiles that can be more easily assigned to ports. The VFSM supports up to 256 ACL Groups.

**Note:** ACL Groups are used for convenience in assigning multiple ACLs to ports. ACL Groups have no effect on the order in which ACLs are applied (see [“ACL Order of Precedence” on page 108](#)). All ACLs assigned to the port (whether individually assigned or part of an ACL Group) are considered as individual ACLs for the purposes of determining their order of precedence.

---

## Assigning ACL Groups to a Port

To assign an ACL Group to a port, use the following command:

```
VFSM(config)# interface port <port number>  
VFSM(config-if)# access-control group <ACL group number>  
VFSM(config-if)# exit
```

---

## ACL Metering and Re-Marking

You can define a profile for the aggregate traffic flowing through the VFSM by configuring a QoS meter (if desired) and assigning ACLs to ports.

**Note:** When you add ACLs to a port, make sure they are ordered correctly in terms of precedence (see [“ACL Order of Precedence” on page 108](#)).

Actions taken by an ACL are called *In-Profile* actions. You can configure additional In-Profile and Out-of-Profile actions on a port. Data traffic can be metered, and re-marked to ensure that the traffic flow provides certain levels of service in terms of bandwidth for different types of network traffic.

### Metering

QoS metering provides different levels of service to data streams through user-configurable parameters. A meter is used to measure the traffic stream against a traffic profile which you create. Thus, creating meters yields In-Profile and Out-of-Profile traffic for each ACL, as follows:

- **In-Profile**—If there is no meter configured or if the packet conforms to the meter, the packet is classified as In-Profile.
- **Out-of-Profile**—If a meter is configured and the packet does not conform to the meter (exceeds the committed rate or maximum burst rate of the meter), the packet is classified as Out-of-Profile.

Using meters, you set a Committed Rate in Kbps (1000 bits per second in each Kbps). All traffic within this Committed Rate is In-Profile. Additionally, you can set a Maximum Burst Size that specifies an allowed data burst larger than the Committed Rate for a brief period. These parameters define the In-Profile traffic.

Meters keep the sorted packets within certain parameters. You can configure a meter on an ACL, and perform actions on metered traffic, such as packet re-marking.

### Re-Marking

Re-marking allows for the treatment of packets to be reset based on new network specifications or desired levels of service. You can configure the ACL to re-mark a packet as follows:

- Change the DSCP value of a packet, used to specify the service level that traffic should receive.
- Change the 802.1p priority of a packet.

---

## ACL Port Mirroring

For IPv4 ACLs and VMaps, packets that match the filter can be mirrored to another switch port for network diagnosis and monitoring.

The source port for the mirrored packets cannot be a portchannel, but may be a member of a portchannel.

The destination port to which packets are mirrored must be a physical port.

If the ACL or VMap has an action (permit, drop, etc.) assigned, it cannot be used to mirror packets for that ACL.

Use the following commands to add mirroring to an ACL:

- For IPv4 ACLs:

```
VFSM(config)# access-control list <ACL number> mirror port <destination port>
```

The ACL must be also assigned to its target ports as usual (see [“Assigning Individual ACLs to a Port” on page 107](#), or [“Assigning ACL Groups to a Port” on page 110](#)).

- For VMaps (see [“VLAN Maps” on page 117](#)):

```
VFSM(config)# access-control vmap <VMap number> mirror port <monitor destination port>
```

See the configuration example on [page 118](#).

---

## Viewing ACL Statistics

ACL statistics display how many packets have “hit” (matched) each ACL. Use ACL statistics to check filter performance or to debug the ACL filter configuration.

You must enable statistics for each ACL that you wish to monitor:

```
VFSM(config)# access-control list <ACL number> statistics
```

---

## ACL Configuration Examples

### ACL Example 1

Use this configuration to block traffic to a specific host. All traffic that ingresses on port EXT1 is denied if it is destined for the host at IP address 100.10.1.1

1. Configure an Access Control List.

```
VFSM(config)# access-control list 1 ipv4 destination-ip-address 100.10.1.1
VFSM(config)# access-control list 1 action deny
```

2. Add ACL 1 to port EXT1.

```
VFSM(config)# interface port EXT1
VFSM(config-if)# access-control list 1
VFSM(config-if)# exit
```

### ACL Example 2

Use this configuration to block traffic from a network destined for a specific host address. All traffic that ingresses in port EXT2 with source IP from class 100.10.1.0/24 and destination IP 200.20.2.2 is denied.

1. Configure an Access Control List.

```
VFSM(config)# access-control list 2 ipv4 source-ip-address 100.10.1.0
255.255.255.0
VFSM(config)# access-control list 2 ipv4 destination-ip-address 200.20.2.2
255.255.255.255
VFSM(config)# access-control list 2 action deny
```

2. Add ACL 2 to port EXT2.

```
VFSM(config)# interface port EXT2
VFSM(config-if)# access-control list 2
VFSM(config-if)# exit
```

### ACL Example 3

Use this configuration to block traffic from a specific IPv6 source address. All traffic that ingresses port EXT2 with source IP from class 2001:0:0:5:0:0:0:2/128 is denied.

1. Configure an Access Control List.

```
VFSM(config)# access-control list 4 ipv4 source-ip-address 100.10.1.0
255.255.255.0
VFSM(config)# access-control list 4 egress-port 3
VFSM(config)# access-control list 4 action deny
```

2. Add ACL 2 to port EXT2.

```
VFSM(config)# interface port EXT1
VFSM(config-if)# access-control list 4
VFSM(config-if)# exit
```

### ACL Example 4

Use this configuration to deny all ARP packets that ingress a port.

1. Configure an Access Control List.

```
VFSM(config)# access-control list 2 ethernet ethernet-type arp
VFSM(config)# access-control list 2 action deny
```

2. Add ACL 2 to port EXT2.

```
VFSM(config)# interface port 2
VFSM(config-if)# access-control list 2
VFSM(config-if)# exit
```

### ACL Example 5

Use the configuration below to permit access to hosts with destination MAC address that matches 11:05:00:10:00:00 FF:F5:FF:FF:FF:FF and deny access to all other hosts.

1. Configure Access Control Lists.

```
VFSM(config)# access-control list 30 ethernet destination-mac-address
11:05:00:10:00:00 FF:F5:FF:FF:FF:FF
VFSM(config)# access-control list 30 action permit
VFSM(config)# access-control list 100 ethernet destination-mac-address
00:00:00:00:00:00 00:00:00:00:00:00
VFSM(config)# access-control list 100 action deny
```

2. Add ACLs to a port.

```
VFSM(config)# interface port 2
VFSM(config-if)# access-control list 30
VFSM(config-if)# access-control list 100
VFSM(config-if)# exit
```

## ACL Example 6

This configuration blocks traffic from a network that is destined for a specific egress port. All traffic that ingresses port EXT1 from the network 100.10.1.0/24 and is destined for port INT3 is denied.

1. Configure an Access Control List.

```
VFSM(config)# access-control list 4 ipv4 source-ip-address 100.10.1.0  
255.255.255.0  
VFSM(config)# access-control list 4 egress-port 3  
VFSM(config)# access-control list 4 action deny
```

2. Add ACL 4 to port EXT1.

```
VFSM(config)# interface port 1  
VFSM(config-if)# access-control list 4  
VFSM(config-if)# exit
```

---

## VLAN Maps

A VLAN map (VMap) is an ACL that can be assigned to a VLAN or VM group rather than to a switch port as with IPv4 ACLs. This is particularly useful in a virtualized environment where traffic filtering and metering policies must follow virtual machines (VMs) as they migrate between hypervisors.

**Note:** VMaps for VM groups are not supported simultaneously on the same ports as vNICs (see [“Virtual NICs” on page 217](#)).

VMaps are configured from the ACL menu, available with the following ISCLI command path:

```
VFSM(config)# access-control vmap <VMAP ID (1-128)>

  action          Set filter action
  egress-port     Set to filter for packets egressing this port
  ethernet        Ethernet header options
  ipv4            IP version 4 header options
  meter           ACL metering configuration
  mirror          Mirror options
  packet-format   Set to filter specific packet format types
  re-mark         ACL re-mark configuration
  statistics      Enable access control list statistics
  tcp-udp         TCP and UDP filtering options
```

The VFSM supports up to 128 VMaps.

Individual VMap filters are configured in the same fashion as IPv4 ACLs, except that VLANs cannot be specified as a filtering criteria (unnecessary, since the VMaps are assigned to a specific VLAN or associated with a VM group VLAN).

Once a VMap filter is created, it can be assigned or removed using the following configuration commands:

- For a regular VLAN:

```
VFSM(config)# vlan <VLAN ID>
VFSM(config-vlan)# [no] vmap <VMap ID> [intports|extports]
```

- For a VM group (see [“VM Group Types” on page 239](#)):

```
VFSM(config)# [no] virt vmgroup <ID> vmap <VMap ID> [intports|extports]
```

**Note:** Each VMap can be assigned to only one VLAN or VM group. However, each VLAN or VM group may have multiple VMaps assigned to it.

When the optional `intports` or `extports` parameter is specified, the action to add or remove the VMap applies for either the internal ports or external ports only. If omitted, the operation will be applied to all ports in the associated VLAN or VM group.

**Note:** VMaps have a lower priority than port-based ACLs. If both an ACL and a VMap match a particular packet, both filter actions will be applied as long as there is no conflict. In the event of a conflict, the port ACL will take priority, though switch statistics will count matches for both the ACL and VMap.

## VMap Example

In this example, EtherType 2 traffic from VLAN 3 internal ports is mirrored to a network monitor on port EXT4.

```
VFSM(config)# access-control vmap 21 packet-format ethernet ethernet-type2
VFSM(config)# access-control vmap 21 mirror port 4
VFSM(config)# access-control vmap 21 action permit
VFSM(config)# vlan 3
VFSM(config-vlan)# vmap 21 intports
```

---

## Management ACLs

Management ACLs (MACLs) filter inbound traffic i.e. traffic toward the CPU. MACLs are applied switch-wide. Traffic can be filtered based on the following:

- IPv4 source address
- IPv4 destination address
- IPv4 protocols
- TCP/UDP destination or source port

Lower MACL numbers have higher priority. Up to 128 MACLs can be configured.

Following is an example MACL configuration based on a destination IP address and a TCP-UDP destination port:

```
VFSM(config)# access-control mac1 1 ipv4 destination-ip-address 1.1.1.1
255.255.255.0
VFSM(config)# access-control mac1 1 tcp-udp destination-port 111 0xffff
VFSM(config)# access-control mac1 1 statistics
VFSM(config)# access-control mac1 1 action permit
VFSM(config)# access-control mac1 1 enable
```

Use the following command to view the MACL configuration:

```
VFSM(config)# show access-control mac1 1

MACL 1 profile   : Enabled
  IPv4
    - DST IP      : 1.1.1.1/255.255.255.0
  TCP/UDP
    - DST Port    : 111/0xffff
  Action         : Permit
  Statistics     : Enabled
```



# Part 3: Switch Basics

This section discusses basic switching functions:

- VLANs
- Port Trunking
- Spanning Tree Protocols (Spanning Tree Groups, Rapid Spanning Tree Protocol, and Multiple Spanning Tree Protocol)
- Quality of Service



---

## Chapter 7. VLANs

This chapter describes network design and topology considerations for using Virtual Local Area Networks (VLANs). VLANs are commonly used to split up groups of network users into manageable broadcast domains, to create logical segmentation of workgroups, and to enforce security policies among logical segments. The following topics are discussed in this chapter:

- [“VLANs and Port VLAN ID Numbers” on page 125](#)
- [“VLAN Tagging/Trunk Mode” on page 127](#)
- [“VLAN Topologies and Design Considerations” on page 132](#)
- [“Protocol-Based VLANs” on page 135](#)
- [“Private VLANs” on page 139](#)

**Note:** Basic VLANs can be configured during initial switch configuration (see “Using the Setup Utility” in the *IBM Networking OS 7.8 Command Reference*). More comprehensive VLAN configuration can be done from the Command Line Interface (see “VLAN Configuration” as well as “Port Configuration” in the *IBM Networking OS 7.8 Command Reference*).

---

## VLANs Overview

Setting up virtual LANs (VLANs) is a way to segment networks to increase network flexibility without changing the physical network topology. With network segmentation, each switch port connects to a segment that is a single broadcast domain. When a switch port is configured to be a member of a VLAN, it is added to a group of ports (workgroup) that belong to one broadcast domain.

Ports are grouped into broadcast domains by assigning them to the same VLAN. Frames received in one VLAN can only be forwarded within that VLAN, and multicast, broadcast, and unknown unicast frames are flooded only to ports in the same VLAN.

The VFSM automatically supports jumbo frames. This default cannot be manually configured or disabled.

The Virtual Fabric Switch Module (VFSM) supports jumbo frames with a Maximum Transmission Unit (MTU) of 9,216 bytes. Within each frame, 18 bytes are reserved for the Ethernet header and CRC trailer. The remaining space in the frame (up to 9,198 bytes) comprise the packet, which includes the payload of up to 9,000 bytes and any additional overhead, such as 802.1q or VLAN tags. Jumbo frame support is automatic: it is enabled by default, requires no manual configuration, and cannot be manually disabled.

**Note:** Jumbo frames are not supported for traffic sent to switch management interfaces.

---

## VLANs and Port VLAN ID Numbers

### VLAN Numbers

IBM N/OS supports up to 1024 VLANs per switch. Even though the maximum number of VLANs supported at any given time is 1024, each can be identified with any number between 1 and 4095. VLAN 1 is the default VLAN for the external ports and the internal blade ports.

VLAN 4095 is reserved for use by the management network, which includes internal management ports (MGT1 and MGT2) and (by default) internal ports. This configuration allows Serial over LAN (SoL) management—a feature available on certain server blades. Management functions can also be assigned to other VLANs (using the following command:

```
VFSM(config)# vlan <x>
VFSM(config-vlan)# management
```

Use the following command to view VLAN information:

```
VFSM(config)# show vlan
```

VLAN	Name	Status	MGT	Ports
1	Default VLAN	ena	dis	INTA2-EXT15 EXT19
2	VLAN 2	ena	dis	empty
222	VLAN 222	ena	dis	empty
1002	VLAN 1002	ena	dis	empty
1515	VLAN 1515	ena	dis	empty
4094	VLAN 4094	ena	dis	INTA1
4095	Mgmt VLAN	ena	ena	EXTM MGT1

Primary	Secondary	Type	Ports	vPorts
-----	-----	-----	-----	-----

**Note:** The sample screens that appear in this document might differ slightly from the screens displayed by your system. Screen content varies based on the type of BladeCenter unit that you are using and the firmware versions and options that are installed.

## PVID/Native VLAN Numbers

Each port in the switch has a configurable default VLAN number, known as its *PVID/Native VLAN*. By default, the PVID/Native VLAN for all non-management ports is set to 1, which correlates to the default VLAN ID. The PVID/Native VLAN for each port can be configured to any VLAN number between 1 and 4094.

Use the following CLI commands to view PVIDs/Native VLANs:

- Port information:

```
VFSM# show interface information
(or)
VFSM# show interface trunk
```

Alias	Port	Tag	Fast	RMON	Lrn	Fld	PVID	NAME	VLAN(s)
		Trk					NVLAN		
INT1	1	y	n	d	e	e	1	INT1	1 4095
INT2	2	y	n	d	e	e	1	INT2	1 4095
INT3	3	y	n	d	e	e	1	INT3	1 4095
INT4	4	y	n	d	e	e	1	INT4	1 4095
INT5	5	y	n	d	e	e	1	INT5	1 4095
INT6	6	y	n	d	e	e	1	INT6	1 4095
INT7	7	y	n	d	e	e	1	INT7	1 4095
INT8	8	y	n	d	e	e	1	INT8	1 4095
INT9	9	y	n	d	e	e	1	INT9	1 4095
INT10	10	y	n	d	e	e	1	INT10	1 4095
INT11	11	y	n	d	e	e	1	INT11	1 4095
INT12	12	y	n	d	e	e	1	INT12	1 4095
ISL1	13	n	n	d	e	e	1	ISL1	1
ISL2	14	n	n	d	e	e	1	ISL2	1
MGT1	15	y	n	d	e	e	4095*	MGT1	4095
MGT2	16	y	n	d	e	e	4095*	MGT2	4095
EXT1	17	n	n	d	e	e	1	EXT1	1
EXT2	18	n	n	d	e	e	1	EXT2	1
EXT3	19	n	n	d	e	e	1	EXT3	1
...									

\* = PVID/Native-VLAN is tagged.  
# = PVID is ingress tagged.  
Trk = Trunk mode  
NVLAN = Native-VLAN

**Note:** The sample output that appears in this document might differ slightly from that displayed by your system. Output varies based on the type of BladeCenter unit that you are using and the firmware versions and options that are installed.

- Port Configuration:

```
VFSM(config)# interface port <port number>
VFSM(config-if)# switchport access vlan <VLAN ID>
```

Each port on the switch can belong to one or more VLANs, and each VLAN can have any number of switch ports in its membership. Any port that belongs to multiple VLANs, however, must have VLAN *tagging/trunk mode* enabled (see [“VLAN Tagging/Trunk Mode” on page 127](#)).

---

## VLAN Tagging/Trunk Mode

N/OS software supports 802.1Q VLAN *tagging*, providing standards-based VLAN support for Ethernet systems.

Tagging places the VLAN identifier in the frame header of a packet, allowing each port to belong to multiple VLANs. When you add a port to multiple VLANs, you also must enable tagging/trunk mode on that port.

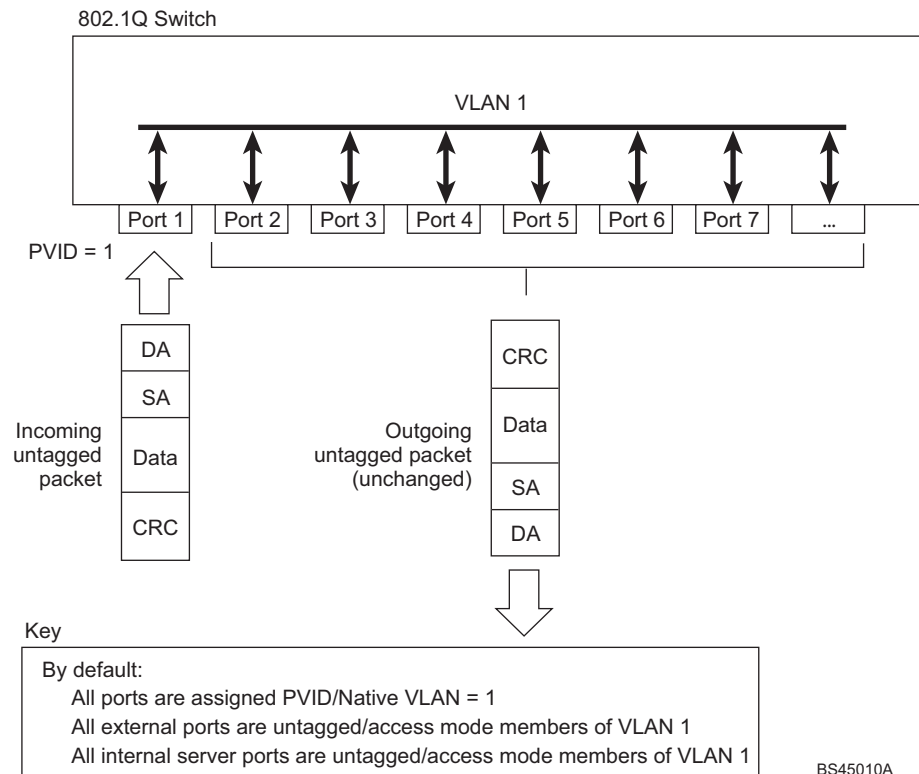
Since tagging fundamentally changes the format of frames transmitted on a tagged/trunk mode port, you must carefully plan network designs to prevent tagged frames from being transmitted to devices that do not support 802.1Q VLAN tags, or devices where tagging/trunk mode is not enabled.

Important terms used with the 802.1Q tagging feature are:

- VLAN identifier (VID)—the 12-bit portion of the VLAN tag in the frame header that identifies an explicit VLAN.
- Port VLAN identifier (PVID)/Native VLAN—a classification mechanism that associates a port with a specific VLAN. For example, a port with a PVID/Native VLAN of 3 (PVID/Native VLAN =3) assigns all untagged frames received on this port to VLAN 3. Any untagged frames received by the switch are classified with the PVID/Native VLAN of the receiving port.
- Tagged frame—a frame that carries VLAN tagging information in the header. This VLAN tagging information is a 32-bit field (VLAN tag) in the frame header that identifies the frame as belonging to a specific VLAN. Untagged frames are marked (tagged) with this classification as they leave the switch through a port that is configured as a tagged/trunk mode port.
- Untagged frame—a frame that does not carry any VLAN tagging information in the frame header.
- Untagged member—a port that has been configured as an untagged/access mode member of a specific VLAN. When an untagged frame exits the switch through an untagged/access mode member port, the frame header remains unchanged. When a tagged frame exits the switch through an untagged/access mode member port, the tag is stripped and the tagged frame is changed to an untagged frame.
- Tagged member—a port that has been configured as a tagged/trunk mode member of a specific VLAN. When an untagged frame exits the switch through a tagged/trunk mode member port, the frame header is modified to include the 32-bit tag associated with the PVID/Native VLAN. When a tagged frame exits the switch through a tagged/trunk mode member port, the frame header remains unchanged (original VID remains).

**Note:** If a 802.1Q tagged frame is received by a port that has VLAN-tagging/trunk mode disabled, then the frame is dropped at the ingress port.

Figure 3. Default VLAN settings



**Note:** The port numbers specified in these illustrations may not directly correspond to the physical port configuration of your switch model.

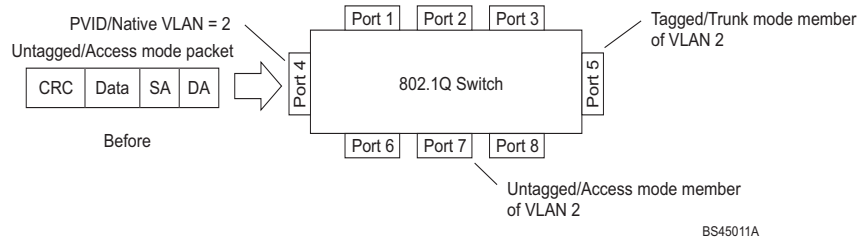
When a VLAN is configured, ports are added as members of the VLAN, and the ports are defined as either *tagged/trunk mode* or *untagged/access mode* (see [Figure 4](#) through [Figure 7](#)).

The default configuration settings for VFSMs have all ports set as untagged/access mode members of VLAN 1 with all ports configured as PVID/Native VLAN = 1. In the default configuration example shown in [Figure 3](#), all incoming packets are assigned to VLAN 1 by the default port VLAN identifier (PVID/Native VLAN =1).

Figure 4 through Figure 7 illustrate generic examples of VLAN tagging/trunk mode. In Figure 4, untagged incoming packets are assigned directly to VLAN 2 (PVID/Native VLAN = 2). Port 5 is configured as a *tagged/trunk mode* member of VLAN 2, and port 7 is configured as an *untagged/access mode* member of VLAN 2.

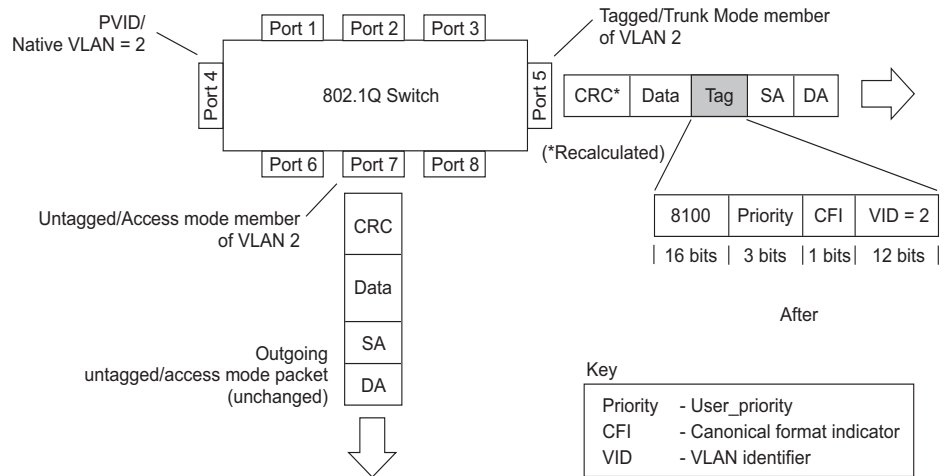
**Note:** The port assignments in the following figures are general examples and are not meant to match any specific VFSM.

Figure 4. Port-based VLAN assignment



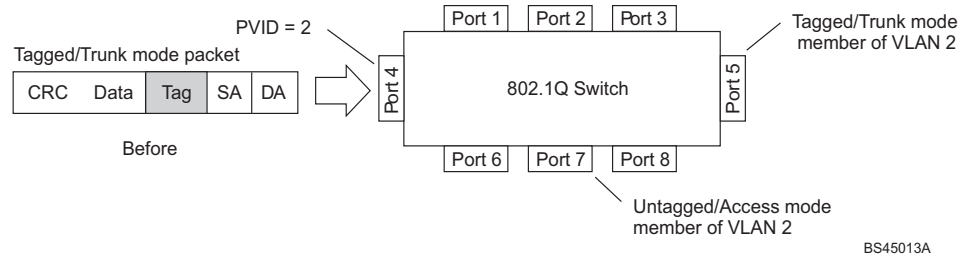
As shown in Figure 5, the untagged packet is marked (tagged) as it leaves the switch through port 5, which is configured as a tagged/trunk mode member of VLAN 2. The untagged packet remains unchanged as it leaves the switch through port 7, which is configured as an untagged/access mode member of VLAN 2.

Figure 5. 802.1Q tagging (after port-based VLAN assignment)



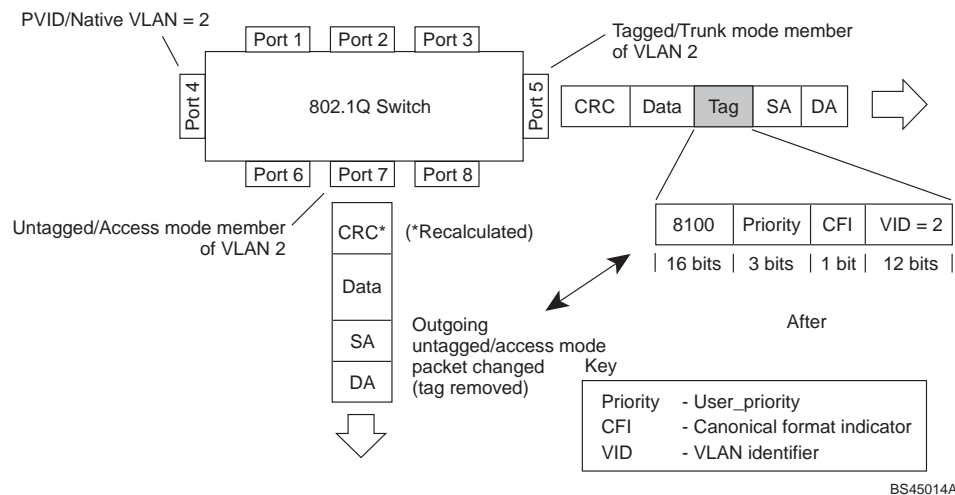
In [Figure 6](#), tagged incoming packets are assigned directly to VLAN 2 because of the tag assignment in the packet. Port 5 is configured as a *tagged/trunk* mode member of VLAN 2, and port 7 is configured as an *untagged/access* mode member of VLAN 2.

Figure 6. 802.1Q tag assignment



As shown in [Figure 7](#), the tagged packet remains unchanged as it leaves the switch through port 5, which is configured as a tagged/trunk mode member of VLAN 2. However, the tagged packet is stripped (untagged) as it leaves the switch through port 7, which is configured as an untagged/access mode member of VLAN 2.

Figure 7. 802.1Q tagging (after 802.1Q tag assignment)



**Note:** Set the configuration to factory default  
(VFSM(config)# boot configuration-block factory) to reset all non-management ports to VLAN 1.

## Ingress VLAN Tagging/Trunk Mode

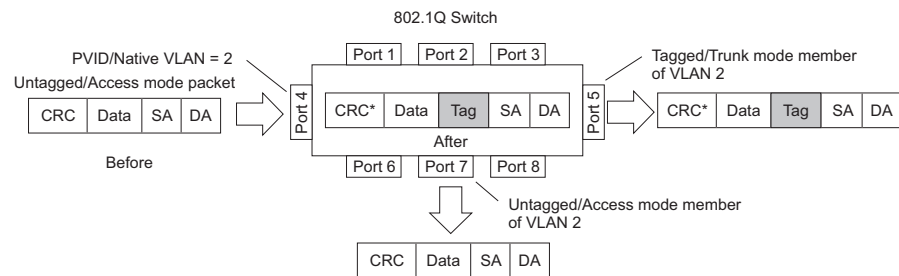
Tagging/trunk mode can be enabled on an ingress port. When a packet is received on an ingress port, and if ingress tagging/trunk mode is enabled on the port, a VLAN tag with the port PVID/Native VLAN is inserted into the packet as the outer VLAN tag. Depending on the egress port setting (tagged/trunk mode or untagged/access mode), the outer tag of the packet is retained or removed when it leaves the egress port.

Ingress VLAN tagging/trunk mode is used to tunnel packets through a public domain without altering the original 802.1Q status.

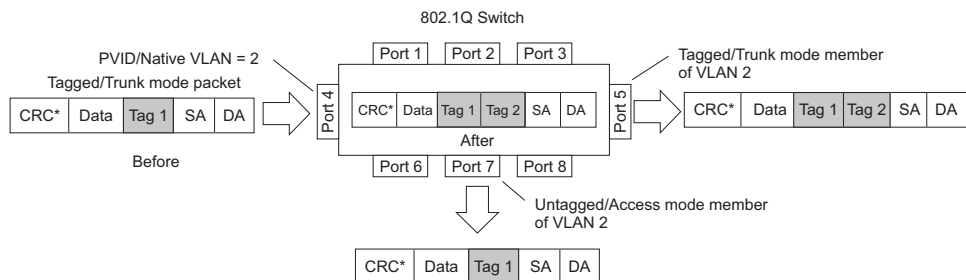
When ingress tagging/trunk mode is enabled on a port, all packets, whether untagged or tagged, will be tagged again. As shown in [Figure 8](#), when tagging/trunk mode is enabled on the egress port, the outer tag of the packet is retained when it leaves the egress port. If tagging/trunk mode is disabled on the egress port, the outer tag of the packet is removed when it leaves the egress port.

Figure 8. 802.1Q tagging (after ingress tagging assignment)

Untagged/Access mode packet received on ingress port



Tagged/Trunk mode packet received on ingress port



By default, ingress tagging is disabled. To enable ingress tagging on a port, use the following command:

```
VFSM(config)# interface port <number>
VFSM(config-if)# tagpvid-ingress
```

## Limitations

Ingress tagging cannot be configured with the following features/configurations:

- vNIC ports
- VMready ports
- Management ports

---

## VLAN Topologies and Design Considerations

- By default, the N/OS software is configured so that tagging/trunk mode is disabled on all external ports, and enabled on all internal ports.
- By default, the N/OS software is configured so that all internal ports are members of VLAN 1. Internal ports are also members of VLAN 4095 (the default management VLAN) to allow Serial over LAN (SoL) management, a feature of certain server blades.
- By default, the N/OS software is configured so that the management ports (MGT1 and MGT2) are members of the default management VLAN 4095.
- Multiple management VLANs can be configured on the switch, in addition to the default VLAN 4095, using the following commands:

```
VFSM(config)# vlan <x>
VFSM(config-vlan)# management
```

- When using Spanning Tree, STG 2-128 may contain only one VLAN unless Multiple Spanning-Tree Protocol (MSTP) mode is used. With MSTP mode, STG 1 to 32 can include multiple VLANs.

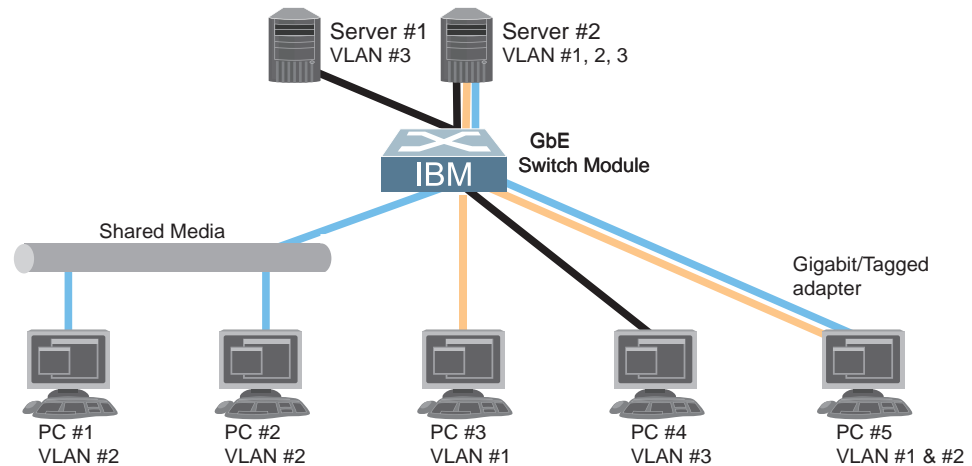
### VLAN Configuration Rules

VLANs operate according to specific configuration rules. When creating VLANs, consider the following rules that determine how the configured VLAN reacts in any network topology:

- All ports involved in trunking and port mirroring must have the same VLAN configuration. If a port is on a trunk with a mirroring port, the VLAN configuration cannot be changed. For more information trunk groups, see [“Configuring a Static Port Trunk” on page 148](#).
- All ports that are involved in port mirroring must have memberships in the same VLANs. If a port is configured for port mirroring, the port's VLAN membership cannot be changed. For more information on configuring port mirroring, see [“Port Mirroring” on page 519](#).
- Management VLANs must contain the management ports (MGT1 and MGT2), and can include one or more internal ports (INT<sub>x</sub>). External ports (EXT<sub>x</sub>) cannot be members of any management VLAN.

## Example: Multiple VLANs with Tagging/Trunk Mode Adapters

Figure 9. Multiple VLANs with VLAN-Tagged/Trunk Mode Gigabit Adapters



The features of this VLAN are described below:

Component	Description
GbE Switch Module	This switch is configured for three VLANs that represent three different IP subnets. Two servers and five clients are attached to the switch.
Server #1	This server is a member of VLAN 3 and has presence in only one IP subnet. The associated internal switch port is only a member of VLAN 3, so tagging/trunk mode is disabled.
Server #2	This high-use server needs to be accessed from all VLANs and IP subnets. The server has a VLAN-tagging/trunk mode adapter installed with VLAN tagging/trunk mode turned on. The adapter is attached to one of the internal switch ports, that is a member of VLANs 1, 2, and 3, and has tagging/trunk mode enabled. Because of the VLAN tagging capabilities of both the adapter and the switch, the server is able to communicate on all three IP subnets in this network. Broadcast separation between all three VLANs and subnets, however, is maintained.
PCs #1 and #2	These PCs are attached to a shared media hub that is then connected to the switch. They belong to VLAN 2 and are logically in the same IP subnet as Server 2 and PC 5. The associated external switch port has tagging/trunk mode disabled.
PC #3	A member of VLAN 1, this PC can only communicate with Server 2 and PC 5. The associated external switch port has tagging/trunk mode disabled.

Component	Description (continued)
PC #4	A member of VLAN 3, this PC can only communicate with Server 1 and Server 2. The associated external switch port has tagging/trunk mode disabled.
PC #5	A member of both VLAN 1 and VLAN 2, this PC has a VLAN-tagging Gigabit Ethernet adapter installed. It can communicate with Server 2 and PC 3 via VLAN 1, and to Server 2, PC 1 and PC 2 via VLAN 2. The associated external switch port is a member of VLAN 1 and VLAN 2, and has tagging/trunk mode enabled.

**Note:** VLAN tagging/trunk mode is required only on ports that are connected to other VFSMs or on ports that connect to tag-capable end-stations, such as servers with VLAN-tagging/trunk mode adapters.

---

## Protocol-Based VLANs

Protocol-based VLANs (PVLANS) allow you to segment network traffic according to the network protocols in use. Traffic for supported network protocols can be confined to a particular port-based VLAN. You can give different priority levels to traffic generated by different network protocols.

With PVLAN, the switch classifies incoming packets by Ethernet protocol of the packets, not by the configuration of the ingress port. When an untagged/access mode or priority-tagged frame arrives at an ingress port, the protocol information carried in the frame is used to determine a VLAN to which the frame belongs. If a frame's protocol is not recognized as a pre-defined PVLAN type, the ingress port's PVID/Native VLAN is assigned to the frame. When a tagged/trunk mode frame arrives, the VLAN ID in the frame's tag is used.

Each VLAN can contain up to eight different PVLANS. You can configure separate PVLANS on different VLANs, with each PVLAN segmenting traffic for the same protocol type. For example, you can configure PVLAN 1 on VLAN 2 to segment IPv4 traffic, and PVLAN 8 on VLAN 100 to segment IPv4 traffic.

To define a PVLAN on a VLAN, configure a PVLAN number (1-8) and specify the frame type and the Ethernet type of the PVLAN protocol. You must assign at least one port to the PVLAN before it can function. Define the PVLAN frame type and Ethernet type as follows:

- Frame type—consists of one of the following values:
  - Ether2 (Ethernet II)
  - SNAP (Subnetwork Access Protocol)
  - LLC (Logical Link Control)
- Ethernet type—consists of a 4-digit (16 bit) hex value that defines the Ethernet type. You can use common Ethernet protocol values, or define your own values. Following are examples of common Ethernet protocol values:
  - IPv4 = 0800
  - IPv6 = 86dd
  - ARP = 0806

## Port-Based vs. Protocol-Based VLANs

Each VLAN supports both port-based and protocol-based association, as follows:

- The default VLAN configuration is port-based. All data ports are members of VLAN 1, with no PVLAN association.
- When you add ports to a PVLAN, the ports become members of both the port-based VLAN and the PVLAN. For example, if you add port EXT1 to PVLAN 1 on VLAN 2, the port also becomes a member of VLAN 2.
- When you delete a PVLAN, its member ports remain members of the port-based VLAN. For example, if you delete PVLAN 1 from VLAN 2, port EXT1 remains a member of VLAN 2.
- When you delete a port from a VLAN, the port is deleted from all corresponding PVLANS.

## PVLAN Priority Levels

You can assign each PVLAN a priority value of 0-7, used for Quality of Service (QoS). PVLAN priority takes precedence over a port's configured priority level. If no priority level is configured for the PVLAN (priority = 0), each port's priority is used (if configured).

All member ports of a PVLAN have the same PVLAN priority level.

## PVLAN Tagging/Trunk Mode

When PVLAN tagging/trunk mode is enabled, the switch tags frames that match the PVLAN protocol. For more information about tagging/trunk mode, see ["VLAN Tagging/Trunk Mode" on page 127](#).

Untagged/access mode ports must have PVLAN tagging disabled. Tagged/trunk mode ports can have PVLAN tagging either enabled or disabled.

PVLAN tagging has higher precedence than port-based tagging. If a port is tag enabled, and the port is a member of a PVLAN, the PVLAN tags egress frames that match the PVLAN protocol.

Use the tag-pvlan command (`vlan <x> protocol-vlan <x> tag-pvlan <x>`) to define the complete list of tag-enabled ports in the PVLAN. Note that all ports not included in the PVLAN tag list will have PVLAN tagging disabled.

## PVLAN Configuration Guidelines

Consider the following guidelines when you configure protocol-based VLANs:

- Each port can support up to 16 VLAN protocols.
- The VFSM can support up to 16 protocols simultaneously.
- Each PVLAN must have at least one port assigned before it can be activated.
- The same port within a port-based VLAN can belong to multiple PVLANS.
- An untagged/access mode port can be a member of multiple PVLANS.
- A port cannot be a member of different VLANs with the same protocol association.

## Configuring PVLAN

Follow this procedure to configure a Protocol-based VLAN (PVLAN).

1. Configure VLAN tagging/trunk mode for ports.

```
VFSM(config)# interface port 1,2
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# exit
```

2. Create a VLAN and define the protocol type(s) supported by the VLAN.

```
VFSM(config)# vlan 2
VFSM(config-vlan)# protocol-vlan 1 frame-type ether2 0800
```

3. Configure the priority value for the protocol.

```
VFSM(config-vlan)# protocol-vlan 1 priority 2
```

4. Add member ports for this PVLAN.

```
VFSM(config-vlan)# protocol-vlan 1 member 1,2
```

**Note:** If VLAN tagging is turned on and the port being added to the VLAN has a different default VLAN (PVID/Native VLAN), you will be asked to confirm changing the PVID to the current VLAN.

5. Enable the PVLAN.

```
VFSM(config-vlan)# protocol-vlan 1 enable
VFSM(config-vlan)# exit
```

## 6. Verify PVLAN operation.

VFSM(config)# show vlan						
VLAN	Name		Status MGT		Ports	
-----	-----		-----		-----	
1	Default	VLAN	ena	dis	INTA2-EXT15	EXT19
2	VLAN	2	ena	dis	INTA1	INTA2
222	VLAN	222	ena	dis	empty	
1002	VLAN	1002	ena	dis	empty	
1515	VLAN	1515	ena	dis	empty	
4094	VLAN	4094	ena	dis	INTA1	
4095	Mgmt	VLAN	ena	ena	EXTM	MGT1
Primary	Secondary	Type	Ports		vPorts	
-----	-----	-----	-----		-----	
-----						
PVLAN	Protocol	FrameType	EtherType	Priority	Status	Ports
-----	-----	-----	-----	-----	-----	-----
2	1	Ether2	0800	2	enabled	INTA1 INTA2
PVLAN	PVLAN-Tagged Ports					
-----	-----					
none	none					

---

## Private VLANs

Private VLANs provide Layer 2 isolation between the ports within the same broadcast domain. Private VLANs can control traffic within a VLAN domain, and provide port-based security for host servers.

IBM N/OS supports Private VLAN configuration as described in RFC 5517.

Use Private VLANs to partition a VLAN domain into sub-domains. Each sub-domain is comprised of one primary VLAN and one secondary VLAN, as follows:

- Primary VLAN—carries unidirectional traffic downstream from promiscuous ports. Each Private VLAN has only one primary VLAN. All ports in the Private VLAN are members of the primary VLAN.
- Secondary VLAN—Secondary VLANs are internal to a private VLAN domain, and are defined as follows:
  - Isolated VLAN—carries unidirectional traffic upstream from the host servers toward ports in the primary VLAN and the gateway. Each Private VLAN can contain only one Isolated VLAN.
  - Community VLAN—carries upstream traffic from ports in the community VLAN to other ports in the same community, and to ports in the primary VLAN and the gateway. Each Private VLAN can contain multiple community VLANs.

After you define the primary VLAN and one or more secondary VLANs, you map the secondary VLAN(s) to the primary VLAN.

## Private VLAN Ports

Private VLAN ports are defined as follows:

- **Promiscuous**—A promiscuous port is an external port that belongs to the primary VLAN. The promiscuous port can communicate with all the interfaces, including ports in the secondary VLANs (Isolated VLAN and Community VLANs). Each promiscuous port can belong to only one Private VLAN.
- **Isolated**—An isolated port is a host port that belongs to an isolated VLAN. Each isolated port has complete layer 2 separation from other ports within the same private VLAN (including other isolated ports), except for the promiscuous ports.
  - Traffic sent to an isolated port is blocked by the Private VLAN, except the traffic from promiscuous ports.
  - Traffic received from an isolated port is forwarded only to promiscuous ports.
- **Community**—A community port is a host port that belongs to a community VLAN. Community ports can communicate with other ports in the same community VLAN, and with promiscuous ports. These interfaces are isolated at layer 2 from all other interfaces in other communities and from isolated ports within the Private VLAN.

Only external ports are promiscuous ports. Only internal ports may be isolated or community ports.

## Configuration Guidelines

The following guidelines apply when configuring Private VLANs:

- Management VLANs and the default VLAN cannot be Private VLANs.
- Private VLAN domains can contain only one primary VLAN and one isolated VLAN.
- Private VLAN domains cannot overlap.
- Ports cannot belong to multiple VLANs of the same private VLAN domain.

## Configuration Example

Follow this procedure to configure a Private VLAN.

1. Select a VLAN and define the Private VLAN type as primary.

```
VFSM(config)# vlan 700
VFSM(config-vlan)# private-vlan primary
VFSM(config-vlan)# exit
```

2. Configure a promiscuous port for VLAN 700.

```
VFSM(config)# interface port 1
VFSM(config-if)# switchport mode private-vlan
VFSM(config-if)# switchport private-vlan mapping 700
VFSM(config-if)# exit
```

3. Configure two secondary VLANs: isolated VLAN and community VLAN.

```
VFSM(config)# vlan 701
VFSM(config-vlan)# private-vlan isolated
VFSM(config-vlan)# exit
VFSM(config)# vlan 702
VFSM(config-vlan)# private-vlan community
VFSM(config-vlan)# exit
```

4. Map secondary VLANs to primary VLAN.

```
VFSM(config)# vlan 700-702
VFSM(config-vlan)# stg 1
VFSM(config-vlan)# exit
VFSM(config)# vlan 700
VFSM(config-vlan)# private-vlan association 701,702
VFSM(config-vlan)# exit
```

5. Configure host ports for secondary VLANs.

```
VFSM(config)# interface port 2
VFSM(config-if)# switchport mode private-vlan
VFSM(config-if)# switchport private-vlan host-association 700 701
VFSM(config-if)# exit

VFSM(config)# interface port 3
VFSM(config-if)# switchport mode private-vlan
VFSM(config-if)# switchport private-vlan host-association 700 702
VFSM(config-if)# exit
```

6. Verify the configuration.

```
VFSM(config)# show vlan private-vlan
```

Primary	Secondary	Type	Ports
700	701	isolated	1 2
700	702	community	1 3



---

## Chapter 8. Ports and Trunking

Trunk groups can provide super-bandwidth, multi-link connections between the Virtual Fabric Switch Module (VFSM) and other trunk-capable devices. A trunk group is a group of ports that act together, combining their bandwidth to create a single, larger virtual link. This chapter provides configuration background and examples for trunking multiple ports together:

- [“Trunking Overview” on page 144](#)
- [“Static Trunks” on page 145](#)
- [“Link Aggregation Control Protocol” on page 150](#)
- [“Configurable Trunk Hash Algorithm” on page 154](#)

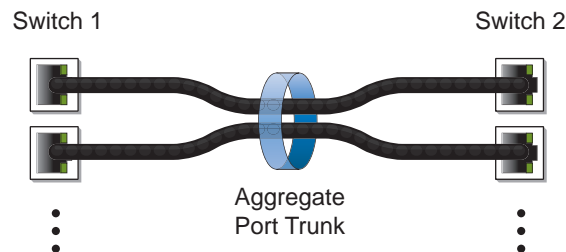
---

## Trunking Overview

When using port trunk groups between two switches, as shown in [Figure 10](#), you can create a virtual link between them, operating with combined throughput levels that depends on how many physical ports are included.

Two trunk types are available: static trunk groups (portchannel), and dynamic trunk groups. Up to 18 trunks of each type are supported in stand-alone (non-stacking) mode, and 64 trunks of each type are supported in stacking mode, depending of the number and type of available ports. Each trunk can include up to 8 member ports.

Figure 10. Port Trunk Group



Trunk groups are also useful for connecting a VFSM to third-party devices that support link aggregation, such as Cisco routers and switches with EtherChannel technology (*not* ISL/802.1Q trunking technology) and Sun's Quad Fast Ethernet Adapter. Trunk Group technology is compatible with these devices when they are configured manually.

Trunk traffic is statistically distributed among the ports in a trunk group, based on a variety of configurable options.

Also, since each trunk group is comprised of multiple physical links, the trunk group is inherently fault tolerant. As long as one connection between the switches is available, the trunk remains active and statistical load balancing is maintained whenever a port in a trunk group is lost or returned to service.

---

## Static Trunks

### Static Trunk Requirements

When you create and enable a static trunk, the trunk members (switch ports) take on certain settings necessary for correct operation of the trunking feature.

Before you configure your trunk, you must consider these settings, along with specific configuration rules, as follows:

1. Read the configuration rules provided in the section, [“Static Trunk Group Configuration Rules” on page 147.](#)
2. Determine which switch ports are to become *trunk members* (the specific ports making up the trunk).
3. Ensure that the chosen switch ports are set to `enabled`, using the `/cfg/port` command. Trunk member ports must have the same VLAN configuration.
4. Consider how the existing Spanning Tree will react to the new trunk configuration. See [“Spanning Tree Protocols” on page 157](#) for configuration guidelines.
5. Consider how existing VLANs will be affected by the addition of a trunk.

### Inter-Switch Link

When the VFSM resides in a BladeCenter HT chassis (BCHT), internal port 13 (ISL1) and internal port 14 (ISL2) are statically-configured as members of Trunk Group 18. These ports can provide an Inter-Switch Link (ISL) between two VFSMs in the chassis. The ISL provides fixed links between switch modules.

The ISL trunk configuration is as follows:

```
Bay 7 VFSM ISL1 connects to Bay 9 VFSM ISL1
Bay 7 VFSM ISL2 connects to Bay 9 VFSM ISL2
Bay 8 VFSM ISL1 connects to Bay 10 VFSM ISL1
Bay 8 VFSM ISL2 connects to Bay 10 VFSM ISL2
```

By default, the ISL ports are `disabled`. When these ports are enabled, they are automatically included in the ISL Trunk Group, which is enabled by default, and cannot be disabled or deleted. You can add up to 6 external ports to each ISL trunk (for a total of up to 8 ports), but no other internal ports can be added.

When the ISL option is detected, the VFSM includes the ISL ports in its configuration and status menus. For example:

```
VFSM(config)# show interface information
```

Alias	Port	Tag	Fast	Lrn	Fld	PVID	NAME	VLAN(s)
		Trk				NVLAN		
INT1	1	y	n	e	e	1	INT1	1 4095
INT2	2	y	n	e	e	1	INT2	1 4095
...								
INT11	11	y	n	e	e	1	INT11	1 4095
INT12	12	y	n	e	e	1	INT12	1 4095
ISL1	13	y	n	e	e	1	ISL1	1
ISL2	14	y	n	e	e	1	ISL2	1
MGT1	15	y	n	e	e	4095*	MGT	4095
MGT2	16	y	n	e	e	4095*	MGT	4095
EXT1	17	n	n	e	e	1	EXT1	1
...								

\* = PVID/Native-VLAN is tagged.  
# = PVID is ingress tagged.  
Trk = Trunk mode  
NVLAN = Native-VLAN

## Static Trunk Group Configuration Rules

The trunking feature operates according to specific configuration rules. When creating trunks, consider the following rules that determine how a trunk group reacts in any network topology:

- All trunks must originate from one network entity (a single device, a VLAG, or multiple devices acting in a stack) and lead to one destination entity. For example, you cannot combine links from two different servers into one trunk group.
- Ports from different member switches in the same stack (see [“Stacking” on page 195](#)) may be aggregated together in one trunk.
- Any physical switch port can belong to only one trunk group.
- Depending on port availability, the switch supports up to 8 ports in each trunk group.
- Internal (INT:x) and external ports (EXT:x) cannot become members of the same trunk group.
- Trunking from third-party devices must comply with Cisco® EtherChannel® technology.

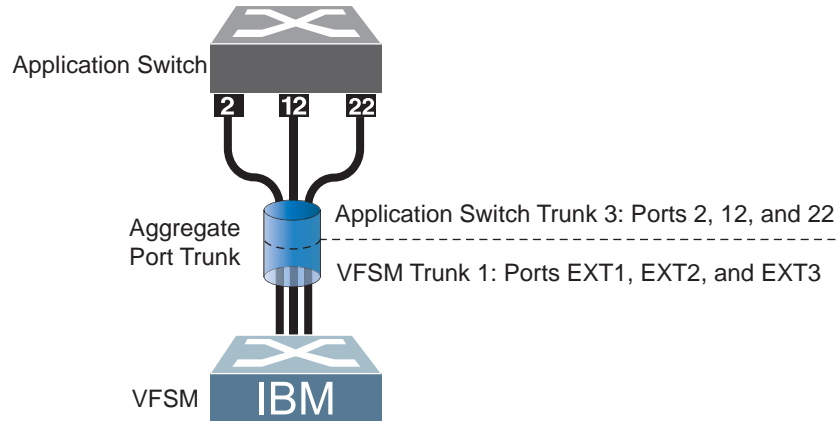
**Note:** Port Aggregation Protocol (PAgP) is not supported.

- All trunk member ports must be assigned to the same VLAN configuration before the trunk can be enabled.
- If you change the VLAN settings of any trunk member, you cannot apply the change until you change the VLAN settings of all trunk members.
- When an active port is configured in a trunk, the port becomes a *trunk member* when you enable the trunk. The Spanning Tree parameters for the port then change to reflect the new trunk settings.
- All trunk members must be in the same Spanning Tree Group (STG) and can belong to only one Spanning Tree Group (STG). However if all ports are *tagged*/trunk mode, then all trunk ports can belong to multiple STGs.
- If you change the Spanning Tree participation of any trunk member to *enabled* or *disabled*, the Spanning Tree participation of all members of that trunk changes similarly.
- When a trunk is enabled, the trunk Spanning Tree participation setting takes precedence over that of any trunk member.
- 802.1X authentication is not supported on ISL ports or on any port that is part of a trunk.
- You cannot configure a trunk member as a monitor port in a port-mirroring configuration.
- Trunks cannot be monitored by a monitor port; however, trunk members can be monitored.
- All ports in static trunks must have the same link configuration (speed, duplex, flow control).

## Configuring a Static Port Trunk

In the example below, three ports are trunked between two switches.

Figure 11. Port Trunk Group Configuration Example



Prior to configuring each switch in the above example, you must connect to the appropriate switch's Command Line Interface (CLI) as the administrator.

**Note:** For details about accessing and using any of the menu commands described in this example, see the IBM Networking OS *Command Reference*.

1. Connect the switch ports that will be members in the trunk group.
2. Configure the trunk using these steps on the VFSM:
  - a. Define a trunk group.

```
VFSM(config)# portchannel 1 port ext1,ext2,ext3      (Add port s to trunk group 1)
VFSM(config)# portchannel 1 enable                  (Enable trunk group 1)
```

- b. V the configuration.

```
VFSM(config)# show portchannel information
```

Examine the resulting information. If any settings are incorrect, make appropriate changes.

3. Repeat the process on the other switch.

```
VFSM(config)# portchannel 3 port 2,12,22
VFSM(config)# portchannel 3 enable
```

Trunk group 1 (on the VFSM) is now connected to trunk group 3 on the Application Switch.

**Note:** In this example, a VFSM and an application switch are used. If a third-party device supporting link aggregation is used (such as Cisco routers and switches with EtherChannel technology or Sun's Quad Fast Ethernet Adapter), trunk groups on the third-party device should be configured manually. Connection problems could arise when using automatic trunk group negotiation on the third-party device.

4. Examine the trunking information on each switch.

VFSM# show portchannel information	(View trunking information)
------------------------------------	-----------------------------

Information about each port in each configured trunk group is displayed. Make sure that trunk groups consist of the expected ports and that each port is in the expected state.

---

# Link Aggregation Control Protocol

## LACP Overview

Link Aggregation Control Protocol (LACP) is an IEEE 802.3ad standard for grouping several physical ports into one logical port (known as a dynamic trunk group or Link Aggregation group) with any device that supports the standard. Please refer to IEEE 802.3ad-2002 for a full description of the standard.

IEEE 802.3ad allows standard Ethernet links to form a single Layer 2 link using the Link Aggregation Control Protocol (LACP). Link aggregation is a method of grouping physical link segments of the same media type and speed in full duplex, and treating them as if they were part of a single, logical link segment. If a link in a LACP trunk group fails, traffic is reassigned dynamically to the remaining link or links of the dynamic trunk group.

The VFSM supports up to 18 LACP trunks, each with up to 8 ports.

**Note:** LACP implementation in IBM N/OS does not support the Churn machine, an option used to detect if the port is operable within a bounded time period between the actor and the partner. Only the Marker Responder is implemented, and there is no marker protocol generator.

A port's Link Aggregation Identifier (LAG ID) determines how the port can be aggregated. The Link Aggregation ID (LAG ID) is constructed mainly from the *system ID* and the port's *admin key*, as follows:

- **System ID:** an integer value based on the switch's MAC address and the system priority assigned in the CLI.
- **Admin key:** a port's Admin key is an integer value (1 - 65535) that you can configure in the CLI. Each VFSM port that participates in the same LACP trunk group must have the same *admin key* value. The Admin key is *local significant*, which means the partner switch does not need to use the same Admin key value.

For example, consider two switches, an Actor (the VFSM) and a Partner (another switch), as shown in [Table 15](#).

*Table 15. Actor vs. Partner LACP configuration*

Actor Switch	Partner Switch 1
Port EXT1 (admin key = 100)	Port 1 (admin key = 50)
Port EXT2 (admin key = 100)	Port 2 (admin key = 50)
Port EXT3 (admin key = 100)	Port 3 (admin key = 70)

In the configuration shown in [Table 15](#), Actor switch ports EXT1 and EXT2 aggregate to form an LACP trunk group with Partner switch ports 1 and 2. Only ports with the same LAG ID are aggregated in the trunk group. Actor switch port EXT3 is not aggregated in the trunk group because it has a different LAG ID. Switch ports configured with the same admin key on the Actor switch but have a different LAG ID (due to Partner switch admin key configuration or due to partner switch MAC

address being different) can be aggregated in another trunk group. i.e. Actor switch port EXT3 can be aggregated in another trunk group with ports that have the same LAG ID as EXT3.

To avoid the Actor switch ports (with the same admin key) from aggregating in another trunk group, you can configure a trunk ID. Ports with the same admin key (although with different LAG IDs) compete to get aggregated in a trunk group. The LAG ID for the trunk group is decided based on the first port that is aggregated in the group. Ports with this LAG ID get aggregated and the other ports are placed in *suspended mode*. As per the configuration shown in [Table 15](#), if port EXT1 gets aggregated first, then the LAG ID of EXT1 would be the LAG ID of the trunk. Port EXT3 would be placed in *suspended mode*. When in *suspended mode*, a port transmits only LACP data units (LACPDUs) and discards all other traffic.

A port may also be placed in *suspended mode* for the following reasons:

- When LACP is configured on the port but it stops receiving LACPDUs from the partner switch.
- When the port has a different LAG ID because of the partner switch MAC being different. For example: when a switch is connected to two partners.

If a trunk member port is down, the VFSM replaces it with a *suspended port*.

Trunk ID can be configured using the following command:

```
VFSM(config)# portchannel <19-36> lacp key <adminkey of the LAG> suspend-individual
```

LACP provides for the controlled addition and removal of physical links for the link aggregation.

Each port in the VFSM can have one of the following LACP modes.

- *off* (default)  
The user can configure this port in to a regular static trunk group.
- *active*  
The port is capable of forming an LACP trunk. This port sends LACPDUs to partner system ports.
- *passive*  
The port is capable of forming an LACP trunk. This port only responds to the LACPDU packets sent from an LACP *active* port.

Each active LACP port transmits LACPDUs, while each passive LACP port listens for LACPDUs. During LACP negotiation, the admin key is exchanged. The LACP trunk group is enabled as long as the information matches at both ends of the link. If the admin key value changes for a port at either end of the link, that port's association with the LACP trunk group is lost.

If an LACP group member port is connected to a port that is in LACP *off* mode, the LACP port will not be able to converge and the link goes down.

When the system is initialized, all ports by default are in LACP *off* mode and are assigned unique *admin keys*. To make a group of ports aggregatable, you assign them all the same *admin key*. You must set the port's LACP mode to *active* to activate LACP negotiation. You can set other port's LACP mode to *passive*, to reduce the amount of LACPDU traffic at the initial trunk-forming stage.

Use the following command to check whether the ports are trunked.

```
VFSM # show lacp information
```

Static trunks are listed as trunks 1 through 18. Dynamic trunks are listed as 19 through 36.

## Static LACP

By default, ports configured with the same LACP admin key can establish multiple link aggregations with remote endpoints that have mixed LAG IDs. For instance, the Actor switch (the VFSM) may be connected to two Partner switches with the following configurations:

*Table 16. Actor vs. Partners with mixed LACP configurations*

Actor Switch	Partner Switch 1	Partner Switch 2
Port 7 (admin key = 100)	Port 1 (admin key = 50)	
Port 8 (admin key = 100)	Port 2 (admin key = 50)	
Port 9 (admin key = 100)	Port 3 (admin key = 60)	
Port 10 (admin key = 100)	Port 4 (admin key = 60)	
Port 11 (admin key = 100)		Port 1 (admin key = 10)
Port 12 (admin key = 100)		Port 2 (admin key = 10)

This setup will establish 3 dynamic trunk groups:

- Actor Switch ports 7 and 8 with Partner Switch 1 ports 1 and 2
- Actor Switch ports 9 and 10 with Partner Switch 1 ports 3 and 4
- Actor Switch ports 11 and 12 with Partner Switch 2 ports 1 and 2

Depending on network topology and configuration, secondary link aggregations on ports that have the same LAG ID may create switching loops. To prevent this, it is possible to restrict participation for these ports to a single link aggregation, using the following command:

```
VFSM(config)#portchannel <19-36> lacp key <I-65535>  
suspend-individual
```

In this mode, ports sharing the specified LACP admin key can form a single trunk, with the designated `trunk ID`. The active trunk is picked based on the ports which occupy first the trunk ID. Member ports that cannot join this trunk are prohibited from forming secondary link aggregations. Instead, they are set in a suspend state where they discard all non-LACP traffic.

## LACP Minimum Links Option

For dynamic trunks that require a guaranteed amount of bandwidth in order to be considered useful, you can specify the minimum number of links for the trunk. If the specified minimum number of ports are not available, the trunk link will not be established. If an active LACP trunk loses one or more component links, the trunk will be placed in the down state if the number of links falls below the specified minimum. By default, the minimum number of links is 1, meaning that LACP trunks will remain operational as long as at least one link is available.

The LACP minimum links setting can be configured as follows:

- Via interface configuration mode:

```
VFSM(config)# interface port <port number or range>
VFSM(config-if)# port-channel min-links <minimum links>
VFSM(config-if)# exit
```

- Or via portchannel configuration mode:

```
VFSM(config)# interface portchannel lacp <LACP key>
VFSM(config-PortChannel)# port-channel min-links <minimum links>
VFSM(config-if)# exit
```

## Configuring LACP

Use the following procedure to configure LACP for port EXT1 and port EXT2 to participate in link aggregation.

1. Define the admin key on port EXT1. Only ports with the same admin key can form a LACP trunk group

```
VFSM(config)# interface port ext1
VFSM(config-if)# lacp key 100
```

2. Set the LACP mode on port EXT1.

```
VFSM(config-if)# lacp mode active
VFSM(config-if)# exit
```

3. Define the admin key on port EXT2. Only ports with the same admin key can form a LACP trunk group

```
VFSM(config)# interface port ext2
VFSM(config-if)# lacp key 100
```

4. Set the LACP mode on port EXT2.

```
VFSM(config-if)# lacp mode active
VFSM(config-if)# exit
```

---

## Configurable Trunk Hash Algorithm

Traffic in a trunk group is statistically distributed among member ports using a *hash* process where various address and attribute bits from each transmitted frame are recombined to specify the particular trunk port the frame will use. The VFSM uses the RTAG7 model for trunk hashing.

The switch can be configured to use a variety of hashing options. To achieve the most even traffic distribution, select options that exhibit a wide range of values for your particular network. Avoid hashing on information that is not usually present in the expected traffic, or which does not vary.

The VFSM supports the following hashing options, which can be used in any combination:

- For Layer 2 traffic, one of the following combinations may be applied:
  - Source MAC address (*smac*)

```
VFSM(config)# portchannel thash l2thash l2-source-mac-address
```

- Destination MAC address (*dmac*)

```
VFSM(config)# portchannel thash l2thash l2-destination-mac-address
```

- Both source and destination MAC address (enabled by default)

```
VFSM(config)# portchannel thash l2thash l2-source-destination-mac-address
```

**Note:** At least one Layer 2 option must always be enabled; The *smac* and *dmac* options may not both be disabled at the same time.

- For Layer 3 IPv4/IPv6 traffic, one of the following are permitted:
  - Source IP address (*sip*)

```
VFSM(config)# portchannel thash l3thash l3-source-ip-address
```

- Destination IP address (*dip*)

```
VFSM(config)# portchannel thash l3thash l3-destination-ip-address
```

- Both source and destination IP address (enabled by default)

If Layer 2 hashing is preferred for Layer 3 traffic, disable the Layer 3 *sip* and *dip* hashing options and enable the *useL2* option:

```
VFSM(config)# portchannel thash l3thash l3-source-destination-ip
```

Layer 3 traffic will then use Layer 2 options for hashing.

- Ingress port number (enabled by default)

```
VFSM(config)# [no] portchannel thash ingress
```

- Layer 4 port information (enabled by default)

```
VFSM(config)# [no] portchannel thash L4port
```

When enabled, Layer 4 port information (TCP, UDP, etc.) is added to the hash if available. The `L4port` option is ignored when Layer 4 information is not included in the packet (such as for Layer 2 packets), or when the `useL2` option is enabled.

**Note:** For MPLS packets, Layer 4 port information is excluded from the hash calculation. Instead, other IP fields are used, along with the first two MPLS labels.



---

## Chapter 9. Spanning Tree Protocols

When multiple paths exist between two points on a network, Spanning Tree Protocol (STP), or one of its enhanced variants, can prevent broadcast loops and ensure that the Virtual Fabric Switch Module (VFSM) uses only the most efficient network path.

This chapter covers the following topics:

- [“Spanning Tree Protocol Modes” on page 158](#)
- [“Global STP Control” on page 159](#)
- [“PVRST Mode” on page 160](#)
- [“Rapid Spanning Tree Protocol” on page 173](#)
- [“Multiple Spanning Tree Protocol” on page 174](#)
- [“Port Type and Link Type” on page 178](#)

---

## Spanning Tree Protocol Modes

IBM Networking OS 7.8 supports the following STP modes:

- Rapid Spanning Tree Protocol (RSTP)

IEEE 802.1D (2004) RSTP allows devices to detect and eliminate logical loops in a bridged or switched network. When multiple paths exist, STP configures the network so that only the most efficient path is used. If that path fails, STP automatically configures the best alternative active path on the network in order to sustain network operations. RSTP is an enhanced version of IEEE 802.1D (1998) STP, providing more rapid convergence of the Spanning Tree network path states on STG 1.

See [“Rapid Spanning Tree Protocol” on page 173](#) for details.

- Per-VLAN Rapid Spanning Tree (PVRST)

PVRST mode is based on RSTP to provide rapid Spanning Tree convergence, but supports instances of Spanning Tree, allowing one STG per VLAN. PVRST mode is compatible with Cisco R-PVST/R-PVST+ mode.

PVRST is the default Spanning Tree mode on the VFSM. See [“PVRST Mode” on page 160](#) for details.

- Multiple Spanning Tree Protocol (MSTP)

IEEE 802.1Q (2003) MSTP provides both rapid convergence and load balancing in a VLAN environment. MSTP allows multiple STGs, with multiple VLANs in each.

MSTP is supported in stand-alone and stacking mode.

See [“Multiple Spanning Tree Protocol” on page 174](#) for details.

---

## Global STP Control

By default, the Spanning Tree feature is globally enabled on the switch, and is set for PVRST mode. Spanning Tree (and thus any currently configured STP mode) can be globally disabled or re-enabled using the following commands:

VFSM(config)# spanning-tree mode disable	<i>(Globally disable Spanning Tree)</i>
--	---

Spanning Tree can be re-enabled by specifying the STP mode:

VFSM(config)# spanning-tree mode {pvrst rstp mst}
---

---

## PVRST Mode

**Note:** Per-VLAN Rapid Spanning Tree (PVRST) is enabled by default on the VFSM.

Using STP, network devices detect and eliminate logical loops in a bridged or switched network. When multiple paths exist, Spanning Tree configures the network so that a switch uses only the most efficient path. If that path fails, Spanning Tree automatically sets up another active path on the network to sustain network operations.

IBM N/OS PVRST mode is based on IEEE 802.1w RSTP. Like RSTP, PVRST mode provides rapid Spanning Tree convergence. However, PVRST mode is enhanced for multiple instances of Spanning Tree. In PVRST mode, each VLAN may be automatically or manually assigned to one of 127 available STGs, with each STG acting as an independent, simultaneous instance of STP. PVRST uses IEEE 802.1Q tagging to differentiate STP BPDUs and is compatible with Cisco R-PVST/R-PVST+ modes.

The relationship between ports, trunk groups, VLANs, and Spanning Trees is shown in [Table 17](#).

*Table 17. Ports, Trunk Groups, and VLANs*

Switch Element	Belongs To
Port	Trunk group, or one or more VLANs
Trunk group	One or more VLANs
VLAN (non-default)	<ul style="list-style-type: none"><li>• PVRST: One VLAN per STG</li><li>• RSTP: All VLANs are in STG 1</li><li>• MSTP: Multiple VLANs per STG</li></ul>

## Port States

The port state controls the forwarding and learning processes of Spanning Tree. In PVRST, the port state has been consolidated to the following: `discarding`, `learning`, and `forwarding`.

Due to the sequence involved in these STP states, considerable delays may occur while paths are being resolved. To mitigate delays, ports defined as *edge/portfast* ports ([“Port Type and Link Type” on page 178](#)) may bypass the `discarding` and `learning` states, and enter directly into the `forwarding` state.

## Bridge Protocol Data Units

### Bridge Protocol Data Units Overview

To create a Spanning Tree, the switch generates a configuration Bridge Protocol Data Unit (BPDU), which it then forwards out of its ports. All switches in the Layer 2 network participating in the Spanning Tree gather information about other switches in the network through an exchange of BPDUs.

A bridge sends BPDU packets at a configurable regular interval (2 seconds by default). The BPDU is used to establish a path, much like a hello packet in IP routing. BPDUs contain information about the transmitting bridge and its ports, including bridge MAC addresses, bridge priority, port priority, and path cost. If the ports are tagged/trunk mode, each port sends out a special BPDU containing the tagged information.

The generic action of a switch on receiving a BPDU is to compare the received BPDU to its own BPDU that it will transmit. If the received BPDU is better than its own BPDU, it will replace its BPDU with the received BPDU. Then, the switch adds its own bridge ID number and increments the path cost of the BPDU. The switch uses this information to block any necessary ports.

**Note:** If STP is globally disabled, BPDUs from external devices will transit the switch transparently. If STP is globally enabled, for ports where STP is turned off, inbound BPDUs will instead be discarded.

### Determining the Path for Forwarding BPDUs

When determining which port to use for forwarding and which port to block, the VFSM uses information in the BPDU, including each bridge ID. A technique based on the "lowest root cost" is then computed to determine the most efficient path for forwarding.

#### Bridge Priority

The bridge priority parameter controls which bridge on the network is the STG root bridge. To make one switch become the root bridge, configure the bridge priority lower than all other switches and bridges on your network. The lower the value, the higher the bridge priority. Use the following command to configure the bridge priority:

```
VFSM(config)# spanning-tree stp <x> bridge priority <0-65535>
```

#### Port Priority

The port priority helps determine which bridge port becomes the root port or the designated port. The case for the root port is when two switches are connected using a minimum of two links with the same path-cost. The case for the designated port is in a network topology that has multiple bridge ports with the same path-cost connected to a single segment, the port with the lowest port priority becomes the designated port for the segment. Use the following commands to configure the port priority:

```
VFSM(config)# interface port <x>
VFSM(config-if)# spanning-tree stp <STG> priority <port priority>
```

where *priority value* is a number from 0 to 255, in increments of 16 (such as 0, 16, 32, and so on). If the specified priority value is not evenly divisible by 16, the value will be automatically rounded down to the nearest valid increment whenever manually changed in the configuration, or whenever a configuration file from a release prior to N/OS 6.5 is loaded.

## Root Guard

The root guard feature provides a way to enforce the root bridge placement in the network. It keeps a new device from becoming root and thereby forcing STP re-convergence. If a root-guard enabled port detects a root device, that port will be placed in a blocked state.

You can configure the root guard at the port level using the following commands:

```
VFSM(config)# interface port <port number>
VFSM(config-if)# spanning-tree guard root
```

The default state is “none”, i.e. disabled.

## Loop Guard

In general, STP resolves redundant network topologies into loop-free topologies. The loop guard feature performs additional checking to detect loops that might not be found using Spanning Tree. STP loop guard ensures that a non-designated port does not become a designated port.

To globally enable loop guard, enter the following command:

```
VFSM(config)# spanning-tree loopguard
```

**Note:** The global loop guard command will be effective on a port only if the port-level loop guard command is set to default as shown below:

```
VFSM(config-if)# spanning-tree guard loop none
```

To enable loop guard at the port level, enter the following commands:

```
VFSM(config)# interface port <port number>
VFSM(config-if)# spanning-tree guard loop
```

The default state is “none”, i.e. disabled.

## Port Path Cost

The port path cost assigns lower values to high-bandwidth ports, such as 10 Gigabit Ethernet, to encourage their use. The cost of a port also depends on whether the port operates at full-duplex (lower cost) or half-duplex (higher cost). For example, if a 100-Mbps (Fast Ethernet) link has a “cost” of 10 in half-duplex mode, it will have a cost of 5 in full-duplex mode. The objective is to use the fastest links so that the route with the lowest cost is chosen. A value of 0 (the default) indicates that the default cost will be computed for an auto-negotiated link or trunk speed.

Use the following command to modify the port path cost:

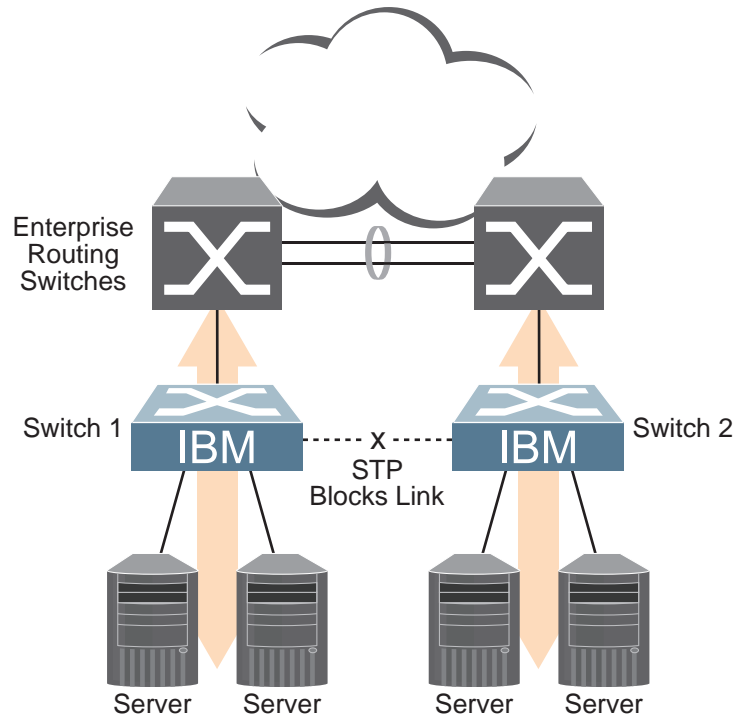
```
VFSM(config)# interface port <port number>
VFSM(config-if)# spanning-tree stp <STG> path-cost <path cost value>
VFSM(config-if)# exit
```

The port path cost can be a value from 1 to 200000000. Specify 0 for automatic path cost.

## Simple STP Configuration

[Figure 12](#) depicts a simple topology using a switch-to-switch link between two VFSM 1 and 2 (via either external ports or internal Inter-Switch Links).

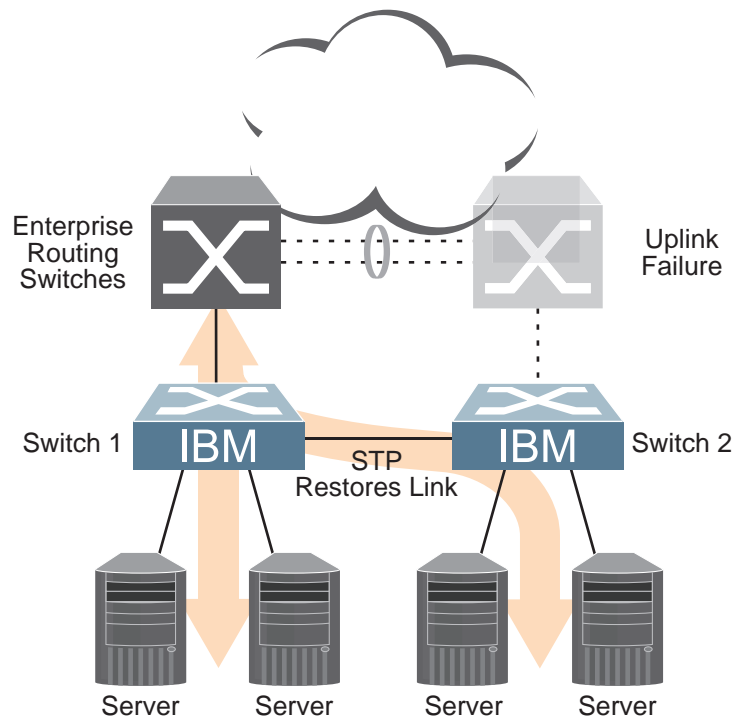
Figure 12. Spanning Tree Blocking a Switch-to-Switch Link



To prevent a network loop among the switches, STP must block one of the links between them. In this case, it is desired that STP block the link between the IBM switches, and not one of the VFSM uplinks or the Enterprise switch trunk.

During operation, if one VFSM experiences an uplink failure, STP will activate the IBM switch-to-switch link so that server traffic on the affected VFSM may pass through to the active uplink on the other VFSM, as shown in [Figure 13](#).

Figure 13. Spanning Tree Restoring the Switch-to-Switch Link



In this example, port EXT10 on each VFSM is used for the switch-to-switch link. To ensure that the VFSM switch-to-switch link is blocked during normal operation, the port path cost is set to a higher value than other paths in the network. To configure the port path cost on the switch-to-switch links in this example, use the following commands on each VFSM.

```
VFSM(config)# interface port 10
VFSM(config-if)# spanning-tree stp 1 path-cost 60000
VFSM(config-if)# exit
```

## Per-VLAN Spanning Tree Groups

PVRST mode supports a maximum of 127 STGs, with each STG acting as an independent, simultaneous instance of STP.

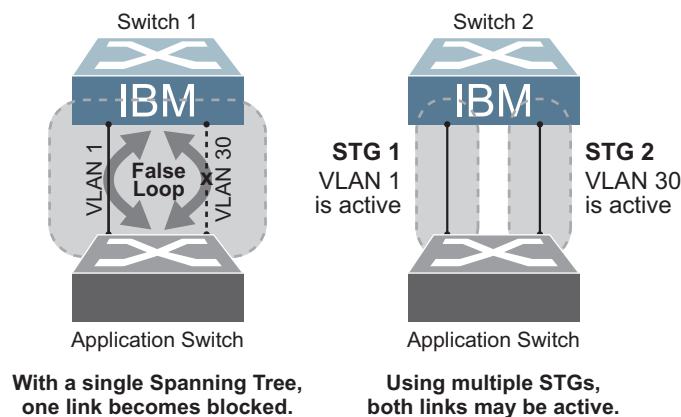
Multiple STGs provide multiple data paths which can be used for load-balancing and redundancy. To enable load balancing between two VFSMs using multiple STGs, configure each path with a different VLAN and then assign each VLAN to a separate STG. Since each STG is independent, they each send their own IEEE 802.1Q tagged Bridge Protocol Data Units (BPDUs).

Each STG behaves as a bridge group and forms a loop-free topology. The default STG 1 may contain multiple VLANs (typically until they can be assigned to another STG). STGs 2-127 may contain only one VLAN each.

### Using Multiple STGs to Eliminate False Loops

Figure 14 shows a simple example of why multiple STGs are needed. In the figure, two ports on a VFSM are connected to two ports on an application switch. Each of the links is configured for a different VLAN, preventing a network loop. However, in the first network, since a single instance of Spanning Tree is running on all the ports of the VFSM, a physical loop is assumed to exist, and one of the VLANs is blocked, impacting connectivity even though no actual loop exists.

Figure 14. Using Multiple Instances of Spanning Tree Group



In the second network, the problem of improper link blocking is resolved when the VLANs are placed into different Spanning Tree Groups (STGs). Since each STG has its own independent instance of Spanning Tree, each STG is responsible only for the loops within its own VLAN. This eliminates the false loop, and allows both VLANs to forward packets between the switches at the same time.

### VLAN and STG Assignment

In PVRST mode, up to 127 STGs are supported. Ports cannot be added directly to an STG. Instead, ports must be added as members of a VLAN, and the VLAN must then be assigned to the STG.

STG 1 is the default STG. Although VLANs can be added to or deleted from default STG 1, the STG itself cannot be deleted from the system. By default, STG 1 is enabled and includes VLAN 1, which by default includes all switch ports (except for management VLANs and management ports).

STG 128 is reserved for switch management. By default, all ports in STG 128 are disabled, but includes management VLAN 4095 and the management ports.

By default, all other STGs (STG 2 through 127) are enabled, though they initially include no member VLANs. VLANs must be assigned to STGs. By default, this is done automatically using VLAN Automatic STG Assignment (VASA), though it can also be done manually (see [“Manually Assigning STGs” on page 167](#)).

When VASA is enabled (as by default), each time a new VLAN is configured, the switch will automatically assign that new VLAN to its own STG. Conversely, when a VLAN is deleted, if its STG is not associated with any other VLAN, the STG is returned to the available pool.

The specific STG number to which the VLAN is assigned is based on the VLAN number itself. For low VLAN numbers (1 through 127), the switch will attempt to assign the VLAN to its matching STG number. For higher numbered VLANs, the STG assignment is based on a simple modulus calculation; the attempted STG number will “wrap around,” starting back at the top of STG list each time the end of the list is reached. However, if the attempted STG is already in use, the switch will select the next available STG. If an empty STG is not available when creating a new VLAN, the VLAN is automatically assigned to default STG 1.

If ports are tagged/trunk mode, each tagged/trunk mode port sends out a special BPDU containing the tagged information. Also, when a tagged/trunk mode port belongs to more than one STG, the egress BPDUs are tagged to distinguish the BPDUs of one STG from those of another STG.

VASA is enabled by default, but can be disabled or re-enabled using the following commands:

```
VFSM(config)# [no] spanning-tree stg-auto
```

**Note:** VASA is not recommended for use with the Emulex Virtual Fabric Adapter for IBM BladeCenter, as in FCoE environments. Disable VASA when connecting the switch to an Emulex Virtual Fabric Adapter.

If VASA is disabled, when you create a new VLAN, that VLAN automatically belongs to default STG 1. To place the VLAN in a different STG, assign it manually.

VASA applies only to PVRST mode and is ignored in RSTP and MSTP modes.

## Manually Assigning STGs

The administrator may manually assign VLANs to specific STGs, whether or not VASA is enabled.

1. If no VLANs exist (other than default VLAN 1), see [“Guidelines for Creating VLANs” on page 168](#) for information about creating VLANs and assigning ports to them.

2. Assign the VLAN to an STG using one of the following methods:
  - From the global configuration mode:

```
VFSM(config)# spanning-tree stp <STG number> vlan <VLAN>
```

- Or from within the VLAN configuration mode:

```
VFSM(config)# vlan <VLAN number>  
VFSM(config-vlan)# stg <STG number>  
VFSM(config-vlan)# exit
```

When a VLAN is assigned to a new STG, the VLAN is automatically removed from its prior STG.

**Note:** For proper operation with switches that use Cisco PVST+, it is recommended that you create a separate STG for each VLAN.

## Guidelines for Creating VLANs

- When you create a new VLAN, if VASA is enabled (the default), that VLAN is automatically assigned its own STG. If VASA is disabled, the VLAN automatically belongs to STG 1, the default STG. To place the VLAN in a different STG, see [“Manually Assigning STGs” on page 167](#). The VLAN is automatically removed from its old STG before being placed into the new STG.
- Each VLAN must be contained *within* a single STG; a VLAN cannot span multiple STGs. By confining VLANs within a single STG, you avoid problems with Spanning Tree blocking ports and causing a loss of connectivity within the VLAN. When a VLAN spans multiple switches, it is recommended that the VLAN remain within the same STG (be assigned the same STG ID) across all the switches.
- If ports are tagged/trunk mode, all trunked ports can belong to multiple STGs.
- A port cannot be directly added to an STG. The port must first be added to a VLAN, and that VLAN added to the desired STG.

## Rules for VLAN Tagged/Trunk Mode Ports

- Tagged/trunk mode ports can belong to more than one STG, but untagged/access mode ports can belong to only one STG.
- When a tagged/trunk mode port belongs to more than one STG, the egress BPDUs are tagged to distinguish the BPDUs of one STG from those of another STG.

## Adding and Removing Ports from STGs

- When you add a port to a VLAN that belongs to an STG, the port is also added to that STG. However, if the port you are adding is an untagged/access mode port and is already a member of another STG, that port will be removed from its current STG and added to the new STG. An untagged/access mode port cannot belong to more than one STG.

For example: Assume that VLAN 1 belongs to STG 1, and that port 1 is untagged/access mode and does not belong to any STG. When you add port 1 to VLAN 1, port 1 will automatically become part of STG 1.

However, if port 5 is untagged/access mode and is a member of VLAN 3 in STG 2, then adding port 5 to VLAN 1 in STG 1 will not automatically add the port to STG 1. Instead, the switch will prompt you to decide whether to change the PVID/Native VLAN from 3 to 1:

```
"Port 5 is an Untagged/Access-mode port and its current PVID/Native-VLAN is 3.  
Confirm changing PVID/Native-VLAN from 3 to 1 [y/n]:" y
```

- When you remove a port from VLAN that belongs to an STG, that port will also be removed from the STG. However, if that port belongs to another VLAN in the same STG, the port remains in the STG.

As an example, assume that port 2 belongs to only VLAN 2, and that VLAN 2 belongs to STG 2. When you remove port 2 from VLAN 2, the port is moved to default VLAN 1 and is removed from STG 2.

However, if port 2 belongs to both VLAN 1 and VLAN 2, and both VLANs belong to STG 2, removing port 2 from VLAN 2 does not remove port 2 from STG 2, because the port is still a member of VLAN 1, which is still a member of STG 2.

- An STG cannot be deleted, only disabled. If you disable the STG while it still contains VLAN members, Spanning Tree will be off on all ports belonging to that VLAN.

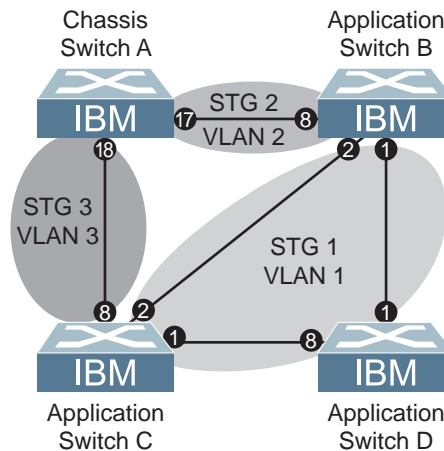
The relationship between port, trunk groups, VLANs, and Spanning Trees is shown in [Table 17 on page 160](#).

## Switch-Centric Configuration

PVRST is switch-centric: STGs are enforced only on the switch where they are configured. The STG ID is not transmitted in the Spanning Tree BPDU. Each Spanning Tree decision is based entirely on the configuration of the particular switch.

For example, in [Figure 15](#), though VLAN 2 is shared by the Switch A and Switch B, each switch is responsible for the proper configuration of its own ports, VLANs, and STGs. Switch A identifies its own port 17 as part of VLAN 2 on STG 2, and the Switch B identifies its own port 8 as part of VLAN 2 on STG 2.

Figure 15. Implementing Multiple Spanning Tree Groups



The VLAN participation for each Spanning Tree Group in [Figure 15 on page 170](#) is as follows:

- **VLAN 1 Participation**

Assuming Switch B to be the root bridge, Switch B transmits the BPDU for VLAN 1 on ports 1 and 2. Switch C receives the BPDU on port 2, and Switch D receives the BPDU on port 1. Because there is a network loop between the switches in VLAN 1, either Switch D will block port 8 or Switch C will block port 1, depending on the information provided in the BPDU.

- **VLAN 2 Participation**

Switch B, the root bridge, generates a BPDU for STG 2 from port 8. Switch A receives this BPDU on port 17, which is assigned to VLAN 2, STG 2. Because switch B has no additional ports participating in STG 2, this BPDU is not forwarded to any additional ports and Switch B remains the designated root.

- **VLAN 3 Participation**

For VLAN 3, Switch A or Switch C may be the root bridge. If Switch A is the root bridge for VLAN 3, STG 3, then Switch A transmits the BPDU from port 18. Switch C receives this BPDU on port 8 and is identified as participating in VLAN 3, STG 2. Since Switch C has no additional ports participating in STG 3, this BPDU is not forwarded to any additional ports and Switch A remains the designated root.

## Configuring Multiple STGs

This configuration shows how to configure the three instances of STGs on the switches A, B, C, and D illustrated in [Figure 15 on page 170](#).

Because VASA is enabled by default, each new VLAN is automatically assigned its own STG. However, for this configuration example, some VLANs are explicitly reassigned to other STGs.

1. Set the Spanning Tree mode on each switch to PVRST.

```
VFSM(config)# spanning-tree mode pvrst
```

**Note:** PVRST is the default mode on the VFSM. This step is not required unless the STP mode has been previously changed, and is shown here merely as an example of manual configuration.

2. Configure the following on Switch A:
3. Enable VLAN 2 and VLAN 3.

```
VFSM(config)# vlan 2
VFSM(config-vlan)# exit
VFSM(config)# vlan 3
VFSM(config-vlan)# exit

If VASA is disabled, enter the following commands:
VFSM(config)# spanning-tree stp 2 vlan 2
VFSM(config)# spanning-tree stp 3 vlan 3
```

Add port 17 to VLAN 2, port 18 to VLAN 3.

```
VFSM(config)# interface port 17
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk allowed vlan add 2
VFSM(config-if)# exit

VFSM(config)# interface port 18
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk allowed vlan add 3
VFSM(config-if)# exit
```

VLAN 2 and VLAN 3 are removed from STG 1.

**Note:** In PVRST mode, each instance of STG is enabled by default.

4. Configure the following on Switch B:

Add port 8 to VLAN 2. Ports 1 and 2 are by default in VLAN 1 assigned to STG 1.

```
VFSM(config)# vlan 2
VFSM(config-vlan)# stg 2
VFSM(config-vlan)# exit
VFSM(config)# interface port 8
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk allowed vlan add 2
VFSM(config-if)# exit

If VASA is disabled, enter the following command:
VFSM(config)# spanning-tree stp 2 vlan 2
```

VLAN 2 is automatically removed from STG 1. By default VLAN 1 remains in STG 1.

5. Configure the following on application switch C:

Add port 8 to VLAN 3. Ports 1 and 2 are by default in VLAN 1 assigned to STG 1.

```
VFSM(config)# vlan 3
VFSM(config-vlan)# stg 3
VFSM(config-vlan)# exit
VFSM(config)# interface port 8
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk allowed vlan add 3
VFSM(config-if)# exit

If VASA is disabled, enter the following command:
VFSM(config)# spanning-tree stp 3 vlan 3
```

VLAN 3 is automatically removed from STG 1. By default VLAN 1 remains in STG 1.

Switch D does not require any special configuration for multiple Spanning Trees. Switch D uses default STG 1 only.

---

## Rapid Spanning Tree Protocol

RSTP provides rapid convergence of the Spanning Tree and provides the fast re-configuration critical for networks carrying delay-sensitive traffic such as voice and video. RSTP significantly reduces the time to reconfigure the active topology of the network when changes occur to the physical topology or its configuration parameters. RSTP reduces the bridged-LAN topology to a single Spanning Tree.

RSTP was originally defined in IEEE 802.1w (2001) and was later incorporated into IEEE 802.1D (2004), superseding the original STP standard.

RSTP parameters apply only to Spanning Tree Group (STG) 1. The PVRST mode STGs 2-128 are not used when the switch is placed in RSTP mode.

RSTP is compatible with devices that run IEEE 802.1D (1998) Spanning Tree Protocol. If the switch detects IEEE 802.1D (1998) BPDUs, it responds with IEEE 802.1D (1998)-compatible data units. RSTP is not compatible with Per-VLAN Rapid Spanning Tree (PVRST) protocol.

**Note:** In RSTP mode, Spanning Tree for the management ports is turned off by default.

### Port States

RSTP port state controls are the same as for PVRST: discarding, learning, and forwarding.

Due to the sequence involved in these STP states, considerable delays may occur while paths are being resolved. To mitigate delays, ports defined as *edge* ports ([“Port Type and Link Type” on page 178](#)) may bypass the discarding and learning states, and enter directly into the forwarding state.

### RSTP Configuration Guidelines

This section provides important information about configuring RSTP. When RSTP is turned on, the following occurs:

- STP parameters apply only to STG 1.
- Only STG 1 is available. All other STGs are turned off.
- All VLANs, including management VLANs, are moved to STG 1.

### RSTP Configuration Example

This section provides steps to configure RSTP.

1. Configure port and VLAN membership on the switch.
2. Set the Spanning Tree mode to Rapid Spanning Tree.

```
VFSM(config)# spanning-tree mode rstp
```

3. Configure port parameters.

```
VFSM(config)# interface port 15
VFSM(config-if)# spanning-tree stp 1 path-cost 1000
VFSM(config-if)# spanning-tree stp 1 bridge forward-delay 20
```

---

## Multiple Spanning Tree Protocol

**Note:** MSTP is supported in stand-alone and stacking mode.

Multiple Spanning Tree Protocol (MSTP) extends Rapid Spanning Tree Protocol (RSTP), allowing multiple Spanning Tree Groups (STGs) which may each include multiple VLANs. MSTP was originally defined in IEEE 802.1s (2002) and was later included in IEEE 802.1Q (2003).

In MSTP mode, the VFSM supports up to 32 instances of Spanning Tree, corresponding to STGs 1-32, with each STG acting as an independent, simultaneous instance of STP.

MSTP allows frames assigned to different VLANs to follow separate paths, with each path based on an independent Spanning Tree instance. This approach provides multiple forwarding paths for data traffic, thereby enabling load-balancing, and reducing the number of Spanning Tree instances required to support a large number of VLANs.

Due to Spanning Tree's sequence of discarding, learning, and forwarding, lengthy delays may occur while paths are being resolved. Ports defined as *edge* ports (["Port Type and Link Type" on page 178](#)) bypass the Discarding and Learning states, and enter directly into the Forwarding state.

**Note:** In MSTP mode, Spanning Tree for the management ports is turned off by default.

## MSTP Region

A group of interconnected bridges that share the same attributes is called an MST region. Each bridge within the region must share the following attributes:

- Alphanumeric name
- Revision number
- VLAN-to STG mapping scheme

MSTP provides rapid re-configuration, scalability and control due to the support of regions, and multiple Spanning-Tree instances support within each region.

## Common Internal Spanning Tree

The Common Internal Spanning Tree (CIST) provides a common form of Spanning Tree Protocol, with one Spanning-Tree instance that can be used throughout the MSTP region. CIST allows the switch to interoperate with legacy equipment, including devices that run IEEE 802.1D (1998) STP.

CIST allows the MSTP region to act as a virtual bridge to other bridges outside of the region, and provides a single Spanning-Tree instance to interact with them.

CIST port configuration includes Hello time, Edge port enable/disable, and Link Type. These parameters do not affect Spanning Tree Groups 1–128. They apply only when the CIST is used.

## MSTP Configuration Guidelines

This section provides important information about configuring Multiple Spanning Tree Groups:

- When MSTP is turned on, the switch automatically moves management VLAN 4095 to the CIST. When MSTP is turned off, the switch moves VLAN 4095 from the CIST to Spanning Tree Group 128.
- When you enable MSTP, you must configure the Region Name. A default version number of 0 is configured automatically.
- Each bridge in the region must have the same name, version number, and VLAN mapping.

## MSTP Configuration Examples

### Example 1

This section provides steps to configure MSTP on the VFSM.

1. Configure port and VLAN membership on the switch.
2. Set the mode to Multiple Spanning Tree region parameters and set the mode to MSTP.

```
VFSM(config)# spanning-tree mst configuration    (Enter MST configuration mode)
VFSM(config-mst)# name <name>                  (Define the Region name)
VFSM(config-mst)# exit
VFSM(config)# spanning-tree mode mst           (Set mode to Multiple Spanning Trees)
```

3. Assign VLANs to MSTP instances.

```
VFSM(config)# spanning-tree mst configuration    (Enter MST configuration mode)
VFSM(config-mst)# instance <instance ID> instance <VLAN number or range>
```

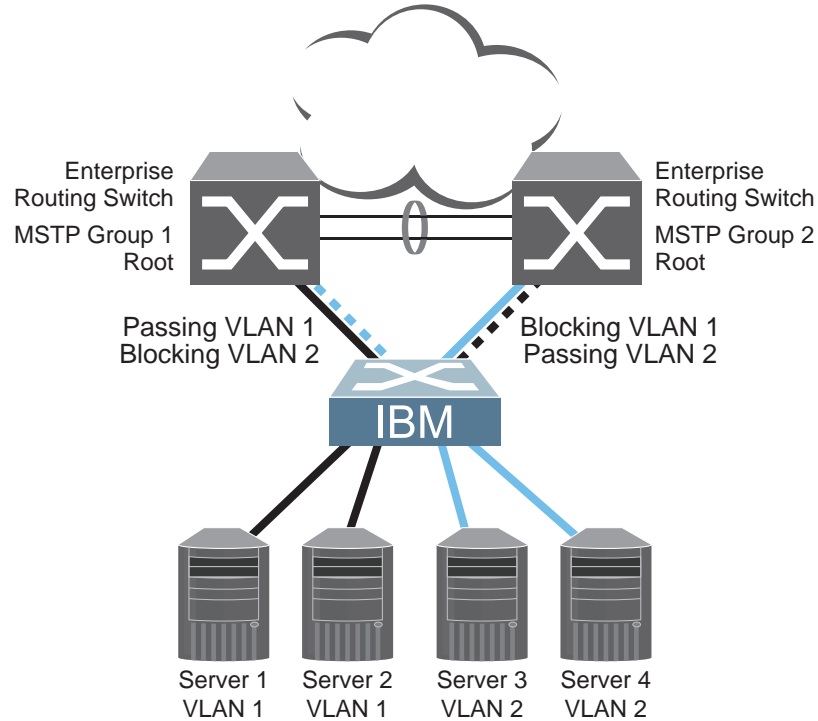
4. Set the MSTP region revision number.

```
VFSM(config)# spanning-tree mst configuration    (Enter MST configuration mode)
VFSM(config-mst)# revision <0-65535>
```

## MSTP Configuration Example 2

This configuration shows how to configure MSTP Groups on the switch, as shown in [Figure 16](#).

Figure 16. Implementing Multiple Spanning Tree Groups



This example shows how multiple Spanning Trees can provide redundancy without wasting any uplink ports. In this example, the server ports are split between two separate VLANs. Both VLANs belong to two different MSTP groups. The Spanning Tree *priority* values are configured so that each routing switch is the root for a different MSTP instance. All of the uplinks are active, with each uplink port backing up the other.

1. Configure port membership and define the STGs for VLAN 1. Enable tagging/trunk mode on uplink ports that share VLANs. Port 19 and port 20 connect to the Enterprise Routing switches.

```
VFSM(config)# interface port 19.20
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# exit
```

2. Configure MSTP: Spanning Tree mode, region name, and version.

```
VFSM(config)# spanning-tree mst configuration
VFSM(config-mst)# name MyRegion                (Define the Region name)
VFSM(config-mst)# revision 100                  (Define the Region level)
VFSM(config-mst)# exit
VFSM(config)# spanning-tree mode mst             (Set mode to Multiple Spanning Tree)
```

3. Map VLANs to MSTP instances:

```
VFSM(config)# spanning-tree mst configuration
VFSM(config-mst)# instance 1 vlan 1
VFSM(config-mst)# instance 1 vlan 2
```

4. Add server ports 1 and 2 to VLAN 1. Add uplink ports 19 and port 20 to VLAN 1

```
VFSM(config)# interface port 1,2,19,20
VFSM(config-vlan)# switchport trunk allowed vlan add 1
VFSM(config-vlan)# exit
```

5. Configure port membership and define the STGs for VLAN 2. Add server ports 3, 4 and 5 to VLAN 2. Add uplink ports 19 and 20 to VLAN 2. Assign VLAN 2 to STG 2.

```
VFSM(config)# interface port 3,4,5,19,20
VFSM(config-vlan)# switchport trunk allowed vlan add 2
VFSM(config-vlan)# exit
```

**Note:** Each STG is enabled by default.

---

## Port Type and Link Type

### Edge Port

A port that does not connect to a bridge is called an *edge port*. Since edge ports are assumed to be connected to non-STP devices (such as directly to hosts or servers), they are placed in the forwarding state as soon as the link is up. Ports INT1-INT14 should be configured as edge ports.

Edge ports send BPDUs to upstream STP devices like normal STP ports, but should not receive BPDUs. If a port with *edge* enabled does receive a BPDU, it immediately begins working as a normal (non-edge) port, and participates fully in Spanning Tree.

Use the following commands to define or clear a port as an edge port:

```
VFSM(config)# interface port <port>
VFSM(config-if)# [no] spanning-tree portfast
VFSM(config-if)# exit
```

### Link Type

The link type determines how the port behaves in regard to Rapid Spanning Tree. Use the following commands to define the link type for the port:

```
VFSM(config)# interface port <port>
VFSM(config-if)# [no] spanning-tree link-type <type>
VFSM(config-if)# exit
```

where *type* corresponds to the duplex mode of the port, as follows:

- `p2p`                      A full-duplex link to another device (point-to-point)
- `shared`                    A half-duplex link is a shared segment and can contain more than one device.
- `auto`                      The switch dynamically configures the link type.

**Note:** Any STP port in full-duplex mode can be manually configured as a shared port when connected to a non-STP-aware shared device (such as a typical Layer 2 switch) used to interconnect multiple STP-aware devices.

---

## Chapter 10. Quality of Service

Quality of Service (QoS) features allow you to allocate network resources to mission-critical applications at the expense of applications that are less sensitive to such factors as time delays or network congestion. You can configure your network to prioritize specific types of traffic, ensuring that each type receives the appropriate QoS level.

The following topics are discussed in this section:

- [“QoS Overview” on page 180](#)
- [“Using ACL Filters” on page 182](#)
- [“Using DSCP Values to Provide QoS” on page 184](#)
- [“Using 802.1p Priorities to Provide QoS” on page 189](#)
- [“Queuing and Scheduling” on page 190](#)
- [“Control Plane Protection” on page 191](#)

---

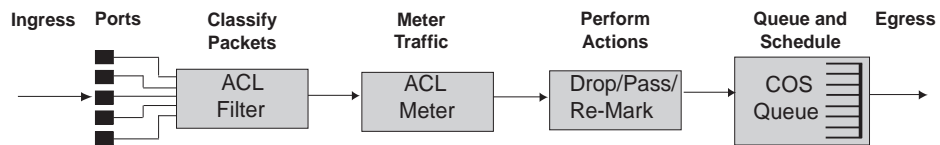
## QoS Overview

QoS helps you allocate guaranteed bandwidth to critical applications, and limit bandwidth for less critical applications. Applications such as video and voice must have a certain amount of bandwidth to work correctly; using QoS, you can provide that bandwidth when necessary. Also, you can put a high priority on applications that are sensitive to timing out or those that cannot tolerate delay, assigning that traffic to a high-priority queue.

By assigning QoS levels to traffic flows on your network, you can ensure that network resources are allocated where they are needed most. QoS features allow you to prioritize network traffic, thereby providing better service for selected applications.

[Figure 17 on page 180](#) shows the basic QoS model used by the Virtual Fabric Switch Module (VFSM).

Figure 17. QoS Model



The VFSM uses the Differentiated Services (DiffServ) architecture to provide QoS functions. DiffServ is described in IETF RFC 2474 and RFC 2475.

With DiffServ, you can establish policies for directing traffic. A policy is a traffic-controlling mechanism that monitors the characteristics of the traffic (for example, its source, destination, and protocol) and performs a controlling action on the traffic when certain characteristics are matched.

The VFSM can classify traffic by reading the DiffServ Code Point (DSCP) or IEEE 802.1p priority value, or by using filters to match specific criteria. When network traffic attributes match those specified in a traffic pattern, the policy instructs the VFSM to perform specified actions on each packet that passes through it. The packets are assigned to different Class of Service (COS) queues and scheduled for transmission.

The basic VFSM QoS model works as follows:

- Classify traffic:
  - Read DSCP
  - Read 802.1p Priority
  - Match ACL filter parameters
- Meter traffic:
  - Define bandwidth and burst parameters
  - Select actions to perform on in-profile and out-of-profile traffic
- Perform actions:
  - Drop packets
  - Pass packets
  - Mark DSCP or 802.1p Priority
  - Set COS queue (with or without re-marking)

- Queue and schedule traffic:
  - Place packets in one of the available COS queues
  - Schedule transmission based on the COS queue weight

---

## Using ACL Filters

Access Control Lists (ACLs) are filters that allow you to classify and segment traffic, so you can provide different levels of service to different traffic types. Each filter defines conditions that packets must match for inclusion in a particular service class, and also the actions that are performed for matching traffic.

The VFSM allows you to classify packets based on various parameters. For example:

- Ethernet—source MAC, destination MAC, VLAN number/mask, Ethernet type, priority
- IPv4—source IP address/mask, destination address/mask, type of service, IP protocol number
- IPv6—source IP address/prefix, destination address/prefix, next header, flow label, traffic class
- TCP/UDP—source port, destination port, TCP flag
- Packet format—Ethernet format, tagging format, IPv4, IPv6
- Egress port

For ACL details, see [“Access Control Lists” on page 103](#).

## Summary of ACL Actions

Actions determine how the traffic is treated. The VFSM QoS actions include the following:

- Pass or Drop the packet
- Re-mark the packet with a new DiffServ Code Point (DSCP)
- Re-mark the 802.1p field
- Set the COS queue

## ACL Metering and Re-Marking

You can define a profile for the aggregate traffic flowing through the VFSM by configuring a QoS meter (if desired) and assigning ACL Groups to ports. When you add ACL Groups to a port, make sure they are ordered correctly in terms of precedence.

Actions taken by an ACL are called *In-Profile* actions. You can configure additional In-Profile and Out-of-Profile actions on a port. Data traffic can be metered, and re-marked to ensure that the traffic flow provides certain levels of service in terms of bandwidth for different types of network traffic.

## Metering

QoS metering provides different levels of service to data streams through user-configurable parameters. A meter is used to measure the traffic stream against a traffic profile which you create. Thus, creating meters yields In-Profile and Out-of-Profile traffic for each ACL, as follows:

- **In-Profile**—If there is no meter configured or if the packet conforms to the meter, the packet is classified as In-Profile.
- **Out-of-Profile**—If a meter is configured and the packet does not conform to the meter (exceeds the committed rate or maximum burst rate of the meter), the packet is classified as Out-of-Profile.

**Note:** Metering is not supported for IPv6 ACLs. All traffic matching an IPv6 ACL is considered in-profile for re-marking purposes.

Using meters, you set a Committed Rate in Kbps (1000 bits per second in each Kbps). All traffic within this Committed Rate is In-Profile. Additionally, you can set a Maximum Burst Size that specifies an allowed data burst larger than the Committed Rate for a brief period. These parameters define the In-Profile traffic.

Meters keep the sorted packets within certain parameters. You can configure a meter on an ACL, and perform actions on metered traffic, such as packet re-marking.

### Re-Marking

Re-marking allows for the treatment of packets to be reset based on new network specifications or desired levels of service. You can configure the ACL to re-mark a packet as follows:

- Change the DSCP value of a packet, used to specify the service level traffic should receive.
- Change the 802.1p priority of a packet.

---

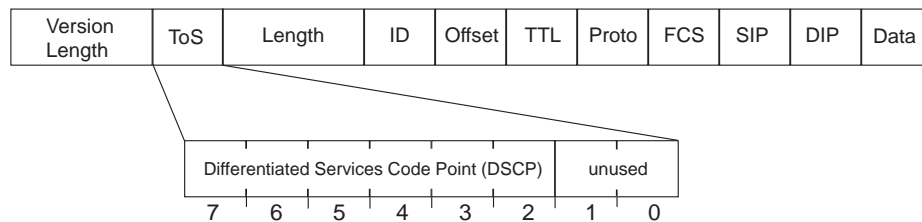
## Using DSCP Values to Provide QoS

The six most significant bits in the TOS byte of the IP header are defined as DiffServ Code Points (DSCP). Packets are marked with a certain value depending on the type of treatment the packet must receive in the network device. DSCP is a measure of the Quality of Service (QoS) level of the packet.

### Differentiated Services Concepts

To differentiate between traffic flows, packets can be classified by their DSCP value. The Differentiated Services (DS) field in the IP header is an octet, and the first six bits, called the DS Code Point (DSCP), can provide QoS functions. Each packet carries its own QoS state in the DSCP. There are 64 possible DSCP values (0-63).

Figure 18. Layer 3 IPv4 Packet



The VFSM can perform the following actions to the DSCP:

- Read the DSCP value of ingress packets
- Re-mark the DSCP value to a new value
- Map the DSCP value to an 802.1p priority

Once the DSCP value is marked, the VFSM can use it to direct traffic prioritization.

### Trusted/Untrusted Ports

By default, all ports on the VFSM are trusted. To configure untrusted ports, re-mark the DSCP value of the incoming packet to a lower DSCP value using the following commands:

```
VFSM(config)# interface port 1
VFSM(config-if)# dscp-marking
VFSM(config-if)# exit
VFSM(config)# qos dscp dscp-mapping <DSCP value (0-63)> <new value>
VFSM(config)# qos dscp re-marking
```

## Per-Hop Behavior

The DSCP value determines the Per Hop Behavior (PHB) of each packet. The PHB is the forwarding treatment given to packets at each hop. QoS policies are built by applying a set of rules to packets, based on the DSCP value, as they hop through the network.

The VFSM default settings are based on the following standard PHBs, as defined in the IEEE standards:

- Expedited Forwarding (EF)—This PHB has the highest egress priority and lowest drop precedence level. EF traffic is forwarded ahead of all other traffic. EF PHB is described in RFC 2598.
- Assured Forwarding (AF)—This PHB contains four service levels, each with a different drop precedence, as shown below. Routers use drop precedence to determine which packets to discard last when the network becomes congested. AF PHB is described in RFC 2597.

Drop Precedence	Class 1	Class 2	Class 3	Class 4
Low	AF11 (DSCP 10)	AF21 (DSCP 18)	AF31 (DSCP 26)	AF41 (DSCP 34)
Medium	AF12 (DSCP 12)	AF22 (DSCP 20)	AF32 (DSCP 28)	AF42 (DSCP 36)
High	AF13 (DSCP 14)	AF23 (DSCP 22)	AF33 (DSCP 30)	AF43 (DSCP 38)

- Class Selector (CS)—This PHB has eight priority classes, with CS7 representing the highest priority, and CS0 representing the lowest priority, as shown below. CS PHB is described in RFC 2474.

Priority	Class Selector	DSCP
Highest	CS7	56
	CS6	48
	CS5	40
	CS4	32
	CS3	24
	CS2	16
	CS1	8
Lowest	CS0	0

## QoS Levels

Table 18 shows the default service levels provided by the VFSM, listed from highest to lowest importance:

Table 18. Default QoS Service Levels

Service Level	Default PHB	802.1p Priority
Critical	CS7	7
Network Control	CS6	6
Premium	EF, CS5	5
Platinum	AF41, AF42, AF43, CS4	4
Gold	AF31, AF32, AF33, CS3	3
Silver	AF21, AF22, AF23, CS2	2
Bronze	AF11, AF12, AF13, CS1	1
Standard	DF, CS0	0

## DSCP Re-Marking and Mapping

### DSCP Re-Marking Overview

The VFSM can re-mark the DSCP value of ingress packets to a new value, and set the 802.1p priority value, based on the DSCP value. You can view the settings by using the following command:

```
VFSM(config)# show qos dscp
Current DSCP Remarking Configuration: OFF

  DSCP    New DSCP    New 802.1p Prio
  -----
    0         0         0
    1         1         0
    ...
   51        51         0
   52        52         0
   53        53         0
   54        54         0
   55        55         0
   56        56         7
   57        57         0
   58        58         0
   59        59         0
   60        60         0
   61        61         0
   62        62         0
   63        63         0
```

Use the following command to turn on DSCP re-marking globally:

```
VFSM(config)# qos dscp re-marking
```

Then you must enable DSCP re-marking on any port that you wish to perform this function (Interface Port mode).

**Note:** If an ACL meter is configured for DSCP re-marking, the meter function takes precedence over QoS re-marking.

## DSCP Re-Marking Configuration Examples

### Example 1

The following example includes the basic steps for re-marking DSCP value and mapping DSCP value to 802.1p.

1. Turn DSCP re-marking on globally, and define the DSCP-DSCP-802.1p mapping. You can use the default mapping.

```
VFSM(config)# qos dscp re-marking
VFSM(config)# qos dscp dscp-mapping <DSCP value (0-63)> <new value>
VFSM(config)# qos dscp dot1p-mapping <DSCP value (0-63)> <802.1p value>
```

2. Enable DSCP re-marking on a port.

```
VFSM(config)# interface port 1
VFSM(config-if)# qos dscp re-marking
VFSM(config-if)# exit
```

### Example 2

The following example includes steps to assign strict priority to VoIP traffic and a lower priority to all other traffic.

1. Create an ACL to re-mark DSCP value and COS queue for all VoIP packets.

```
VFSM(config)# access-control list 2 tcp-udp source-port 5060 0xffff
VFSM(config)# access-control list 2 meter committed-rate 10000000
VFSM(config)# access-control list 2 meter enable
VFSM(config)# access-control list 2 re-mark in-profile dscp 56
VFSM(config)# access-control list 2 re-mark dot1p 7
VFSM(config)# access-control list 2 action permit
```

2. Create an ACL to set a low priority to all other traffic.

```
VFSM(config)# access-control list 3 action set-priority 1
VFSM(config)# access-control list 3 action permit
```

3. Apply the ACLs to a port and enable DSCP marking.

```
VFSM(config)# interface port 5
VFSM(config-if)# access-control list 2
VFSM(config-if)# access-control list 3 ethernet source-mac-address
00:00:00:00:00:00 00:00:00:00:00:00
VFSM(config-if)# dscp-marking
VFSM(config-if)# exit
```

4. Enable DSCP re-marking globally.

```
VFSM(config)# qos dscp re-marking
```

5. Assign the DSCP re-mark value.

```
VFSM(config)# qos dscp dscp-mapping 40 9  
VFSM(config)# qos dscp dscp-mapping 46 9
```

6. Assign strict priority to VoIP COS queue.

```
VFSM(config)# qos transmit-queue weight-cos 7 0
```

7. Map priority value to COS queue for non-VoIP traffic.

```
VFSM(config)# qos transmit-queue mapping 1 1
```

8. Assign weight to the non-VoIP COS queue.

```
VFSM(config)# qos transmit-queue weight-cos 1 2
```

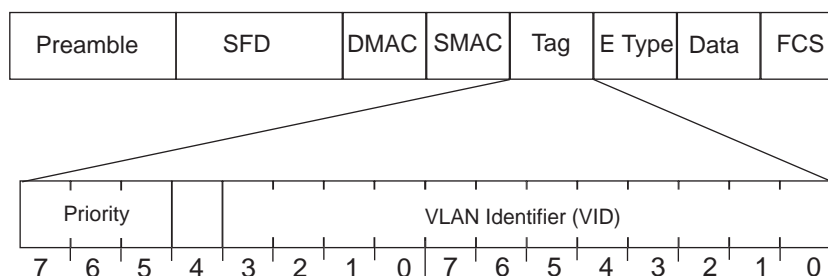
## Using 802.1p Priorities to Provide QoS

### 802.1p Overview

IBM Networking OS provides Quality of Service functions based on the priority bits in a packet's VLAN header. (The priority bits are defined by the 802.1p standard within the IEEE 802.1q VLAN header.) The 802.1p bits, if present in the packet, specify the priority that should be given to packets during forwarding. Packets with a numerically higher (non-zero) priority are given forwarding preference over packets with lower priority bit value.

The IEEE 802.1p standard uses eight levels of priority (0-7). Priority 7 is assigned to highest priority network traffic, such as OSPF or RIP routing table updates, priorities 5-6 are assigned to delay-sensitive applications such as voice and video, and lower priorities are assigned to standard applications. A value of 0 (zero) indicates a "best effort" traffic prioritization, and this is the default when traffic priority has not been configured on your network. The VFSM can filter packets based on the 802.1p values, and it can assign or overwrite the 802.1p value in the packet.

Figure 19. Layer 2 802.1q/802.1p VLAN Tagged Packet



Ingress packets receive a priority value, as follows:

- **Tagged packets**—VFSM reads the 802.1p priority in the VLAN tag.
- **Untagged packets**—VFSM tags the packet and assigns an 802.1p priority, based on the port's default priority (`/cfg/port <x>/8021ppri`).

Egress packets are placed in a COS queue based on the priority value, and scheduled for transmission based on the scheduling weight of the COS queue.

### 802.1p Configuration Example

To configure a port's default 802.1p priority value, use the following commands.

```
VFSM(config)# interface port 1
VFSM(config-if)# dot1p <802.1p value (0-7)>
VFSM(config-if)# exit
```

See [“Queuing and Scheduling” on page 190](#) for details on scheduling weights.

---

## Queuing and Scheduling

The VFSM can be configured to have either 2 or 8 output Class of Service (COS) queues per port, into which each packet is placed. Each packet's 802.1p priority determines its COS queue, except when an ACL action sets the COS queue of the packet.

**Note:** In stacking mode, because one COS queue is reserved for internal use, the number of configurable COS queues is either 1 or 7.

You can configure the following attributes for COS queues:

- Map 802.1p priority value to a COS queue
- Define the scheduling weight of each COS queue

You can map 802.1p priority value to a COS queue, as follows:

```
VFSM(config)# qos transmit-queue mapping <802.1p priority value (0-7)> <COS queue (0-7)>
```

To set the COS queue scheduling weight, use the following command.

```
VFSM(config)# qos transmit-queue weight-cos <COSq number> <COSq weight (0-15)>
```

The scheduling weight can be set from 0 to 15. Weight values from 1 to 15 set the queue to use weighted round-robin (WRR) scheduling, which distributes larger numbers of packets to queues with the highest weight values. For distribution purposes, each packet is counted the same, regardless of the packet's size.

A scheduling weight of 0 (zero) indicates strict priority. Traffic in strict priority queue has precedence over other all queues. If more than one queue is assigned a weight of 0, the strict queue with highest queue number will be served first. Once all traffic in strict queues is delivered, any remaining bandwidth will be allocated to the WRR queues, divided according to their weight values.

**Note:** Use caution when assigning strict scheduling to queues. Heavy traffic in queues assigned with a weight of 0 can starve lower priority queues.

For a scheduling method that uses a weighted deficit round-robin (WDRR) algorithm, distributing packets with an awareness of packet size, see [“Enhanced Transmission Selection” on page 281](#).

---

## Control Plane Protection

Control plane receives packets that are required for the internal protocol state machines. This type of traffic is usually received at low rate. However, in some situations such as DOS attacks, the switch may receive this traffic at a high rate. If the control plane protocols are unable to process the high rate of traffic, the switch may become unstable.

The control plane receives packets that are channeled through protocol-specific packet queues. Multiple protocols can be channeled through a common packet queue. However, one protocol cannot be channeled through multiple packet queues. These packet queues are applicable only to the packets received by the software and does not impact the regular switching or routing traffic. Packet queue with a higher number has higher priority.

You can configure the bandwidth for each packet queue. Protocols that share a packet queue will also share the bandwidth.

Given below are the commands to configure the control plane protection (CoPP) feature:

```
VFSM(config)# qos protocol-packet-control packet-queue-map <0-47> <protocol>
(Configure a queue for a protocol)
VFSM(config)# qos protocol-packet-control rate-limit-packet-queue <0-47> <1-10000>
(Set the bandwidth for the queue, in packets per second)
```



# **Part 4: Advanced Switching Features**



---

## Chapter 11. Stacking

This chapter describe how to implement the stacking feature in the IBM Virtual Fabric 10Gb Switch Module. The following concepts are covered:

- [“Stacking Overview” on page 196](#)
- [“Stack Membership” on page 198](#)
- [“Configuring a Stack” on page 202](#)
- [“Managing a Stack” on page 207](#)
- [“Upgrading Software in an Existing Stack” on page 209](#)
- [“Replacing or Removing Stacked Switches” on page 211](#)
- [“ISCLI Stacking Commands” on page 214](#)

---

## Stacking Overview

A *stack* is a group of up to eight Virtual Fabric Switch Module switches with IBM Networking OS that work together as a unified system. A stack has the following properties, regardless of the number of switches included:

- The network views the stack as a single entity.
- The stack can be accessed and managed as a whole using standard switch IP interfaces configured with IPv4 addresses.
- Once the stacking links have been established (see below), the number of ports available in a stack equals the total number of remaining ports of all the switches that are part of the stack.
- The number of available IP interfaces, VLANs, Trunks, Trunk Links, and other switch attributes are not aggregated among the switches in a stack. The totals for the stack as a whole are the same as for any single switch configured in stand-alone mode.

## Stacking Requirements

Before IBM N/OS switches can form a stack, they must meet the following requirements:

- All switches must be the same model (Virtual Fabric Switch Module).
- Each switch must be installed with N/OS, version 7.8 or later. The same release version is not required, as the Master switch will push a firmware image to each differing switch which is part of the stack.
- The recommended stacking topology is a bidirectional ring (see [Figure 20 on page 203](#)). To achieve this, two external 10Gb Ethernet ports on each switch must be reserved for stacking. By default, the first two 10Gb Ethernet ports are used.
- The cables used for connecting the switches in a stack carry low-level, inter-switch communications as well as cross-stack data traffic critical to shared switching functions. Always maintain the stability of stack links in order to avoid internal stack reconfiguration.

## Stacking Limitations

The VFSM with N/OS 7.8 can operate in one of two modes:

- Default mode, which is the regular stand-alone (or non-stacked) mode.
- Stacking mode, in which multiple physical switches aggregate functions as a single switching device.

When in stacking mode, the following stand-alone features are not supported:

- Active Multi-Path Protocol (AMP)
- BCM rate control
- Border Gateway Protocol (BGP)
- IGMP Relay and IGMPv3
- IPv6
- Loopback Interfaces
- MAC address notification
- Operation, Administration and Maintenance (OAM)
- OSPF and OSPFv3
- Port flood blocking
- Protocol-based VLANs
- RIP
- Router IDs
- Route maps
- sFlow port monitoring
- Static Forwarding Database (FDB)
- Static MAC address adding
- Static multicast
- Multicast Listener Discovery (MLD)
- Uni-Directional Link Detection (UDLD)
- Virtual Router Redundancy Protocol (VRRP)

**Note:** In stacking mode, switch menus and command for unsupported features may be unavailable, or may have no effect on switch operation.

---

## Stack Membership

A stack contains up to eight switches, interconnected by a stack trunk in a local ring topology (see [Figure 20 on page 203](#)). With this topology, only a single stack link failure is allowed.

An operational stack must contain one Master and one or more Members, as follows:

- **Master**  
One switch controls the operation of the stack and is called the Master. The Master provides a single point to manage the stack. A stack must have one and only one Master. The firmware image, configuration information, and run-time data are maintained by the Master and pushed to each switch in the stack as necessary.
- **Member**  
Member switches provide additional port capacity to the stack. Members receive configuration changes, run-time information, and software updates from the Master.
- **Backup**  
One member switch can be designated as a Backup to the Master. The Backup takes over control of the stack if the Master fails. Configuration information and run-time data are synchronized with the Master.

## The Master Switch

An operational stack can have only one active Master at any given time. In a normal stack configuration, one switch is configured as a Master and all others are configured as Members.

When adding new switches to an existing stack, the administrator should explicitly configure each new switch for its intended role as a Master (only when replacing a previous Master) or as a Member. All stack configuration procedures in this chapter depict proper role specification.

However, although uncommon, there are scenarios in which a stack may temporarily have more than one Master switch. Should this occur, one Master switch will automatically be chosen as the active Master for the entire stack. The selection process is designed to promote stable, predictable stack operation and minimize stack reboots and other disruptions.

## Splitting and Merging One Stack

If stack links or Member switches fail, any Members which cannot access either the Master or Backup are considered *isolated* and will not process network traffic (see [“No Backup” on page 200](#)). Members which have access to a Master or Backup (or both), despite other link or Member failures, will continue to operate as part of their active stack.

If multiple stack links or stack Member switches fail, thereby separating the Master and Backup into separate sub-stacks, the Backup automatically becomes an active Master for the partial stack in which it resides. Later, if the topology failures are corrected, the partial stacks will merge, and the two active Masters will come into contact.

In this scenario, if both the (original) Master and the Backup (acting as Master) are in operation when the merger occurs, the original Master will reassert its role as active Master for the entire stack. If any configuration elements were changed and applied on the Backup during the time it acted as Master (and forwarded to its connected Members), the Backup and its affected Members will reboot and will be reconfigured by the returning Master before resuming their regular roles.

However, if the original Master switch is disrupted (powered down or in the process of rebooting) when it is reconnected with the active stack, the Backup (acting as Master) will retain its acting Master status in order to avoid disruption to the functioning stack. The deferring Master will temporarily assume a role as Backup.

If both the Master and Backup are rebooted, the switches will assume their originally configured roles.

If, while the stack is still split, the Backup (acting as Master) is explicitly reconfigured to become a regular Master, then when the split stacks are finally merged, the Master with the lowest MAC address will become the new active Master for the entire stack.

## Merging Independent Stacks

If switches from different stacks are linked together in a stack topology without first reconfiguring their roles as recommended, it is possible that more than one switch in the stack might be configured as a Master.

Although all switches which are configured for stacking and joined by stacking links are recognized as potential stack participants by any operational Master switches, they are not brought into operation within the stack until explicitly assigned (or “bound”) to a specific Master switch.

Consider two independent stacks, Stack A and Stack B, which are merged into one stacking topology. The stacks will behave independently until the switches in Stack B are bound to Master A (or vice versa). In this example, once the Stack B switches are bound to Master A, Master A will automatically reconfigure them to operate as Stack A Members, regardless of their original status within Stack B.

However, for purposes of future Backup selection, reconfigured Masters retain their identity as configured Masters, even though they otherwise act as Members. In case the configured Master goes down and the Backup takes over as the new Master, these reconfigured Masters become the new Backup. When the original configured Master of the stack boots up again, it acts as a Member. This is one way to have multiple backups in a stack.

## Backup Switch Selection

An operational stack can have one optional Backup at any given time. Only the Backup specified in the active Master’s configuration is eligible to take over current stack control when the Master is rebooted or fails. The Master automatically synchronizes configuration settings with the specified Backup to facilitate the transfer of control functions.

The Backup retains its status until one of the following occurs:

- The Backup setting is deleted or changed using the following command from the active Master:

```
VFSM(config)# no stack backup

-or-

VFSM(config)# stack backup <cnum 1-8>
```

- A new Master assumes operation as active Master in the stack, and uses its own configured Backup settings.
- The active Master is rebooted with the boot configuration set to factory defaults (clearing the Backup setting).

## Master Failover

When the Master switch is present, it controls the operation of the stack and pushes configuration information to the other switches in the stack. If the active Master fails, then the designated Backup (if one is defined in the Master's configuration) becomes the new acting Master and the stack continues to operate normally.

## Secondary Backup

When a Backup takes over stack control operations, if any other configured Masters (acting as Member switches) are available within the stack, the Backup will select one as a secondary Backup. The primary Backup automatically reconfigures the secondary Backup and specifies itself (the primary Backup) as the new Backup in case the secondary fails. This prevents the chain of stack control from migrating too far from the original Master and Backup configuration intended by the administrator.

## Master Recovery

If the prior Master recovers in a functioning stack where the Backup has assumed stack control, the prior Master does not reassert itself as the stack Master. Instead, the prior Master will assume a role as a secondary Backup to avoid further stack disruption.

Upon stack reboot, the Master and Backup will resume their regular roles.

## No Backup

If a Backup is not configured on the active Master, or the specified Backup is not operating, then if the active Master fails, the stack will reboot without an active Master.

When a group of stacked switches are rebooted without an active Master present, the switches are considered to be *isolated*. All isolated switches in the stack are placed in a `WAITING` state until a Master appears. During this `WAITING` period, all the external ports and internal server ports of these Member switches are placed into operator-disabled state. Without the Master, a stack cannot respond correctly to networking events.

## Stack Member Identification

Each switch in the stack has two numeric identifiers, as follows:

- **Attached Switch Number** (`asnum`)  
An `asnum` is automatically assigned by the Master switch, based on each Member switch's physical connection in relation to the Master. The `asnum` is mainly used as an internal ID by the Master switch and is not user-configurable.
- **Configured Switch Number** (`csnum`):  
The `csnum` is the logical switch ID assigned by the stack administrator. The `csnum` is used in most stacking-related configuration commands and switch information output. It is also used as a port prefix to distinguish the relationship between the ports on different switches in the stack.

It is recommended that `asnum 1` and `csnum 1` be used for identifying the Master switch. By default, `csnum 1` is assigned to the Master. If `csnum 1` is not available, the lowest available `csnum` is assigned to the Master.

---

## Configuring a Stack

**Note:** When stacking mode is enabled on the switch, the configuration is reset to factory default and the port numbering changes.

## Configuration Overview

This section provides procedures for creating a stack of switches. The high-level procedure is as follows:

- Choose one Master switch for the entire stack.
- Set all stack switches to stacking mode.
- Configure the same stacking VLAN for all switches in the stack.
- Configure the desired stacking interlinks.
- Configure an external IP interface on the Master (if external management is desired).
- Bind Member switches to the Master.
- Assign a Backup switch.

These tasks are covered in detail in the following sections.

## Best Configuration Practices

The following are guidelines for building an effective switch stack:

- Always connect the stack switches in a complete ring topology (see [Figure 20 on page 203](#)).
- Avoid disrupting the stack connections unnecessarily while the stack is in operation.
- For enhanced redundancy when creating port trunks, include ports from different stack members in the trunks.
- Avoid altering the stack `asnum` and `csnum` definitions unnecessarily while the stack is in operation.
- When in stacking mode, the highest QoS priority queue is reserved for internal stacking requirements. Therefore, only seven priority queues will be available for regular QoS use.
- Configure only as many QoS levels as necessary. This allows the best use of packet buffers.
- Before configuring the stack:
  - Identify the VLAN to be used as the stacking VLAN.
  - Save the current configuration. The port numbering will change once stacking is enabled. Use the saved configuration to reassign ports/interfaces as per the new port numbering scheme. Once a stack is configured, port numbers are displayed throughout the BBI using the `csnum` to identify the switch, followed by the switch port number. For example:

2:17  
|  
Configured Switch number |  
|  
Port number

## Stacking VLANs

VLAN 4090 is the default VLAN reserved for internal traffic on stacking ports.

**Note:** Do not use VLAN 4090 for any purpose other than internal stacking traffic.

## Configuring Each Switch in a Stack

To configure each switch for stacking, connect to the internal management IP interface for each switch (assigned by the management system) and use the CLI to perform the following steps.

**Note:** IPv6 is not supported in stacking mode. IP interfaces must use IPv4 addressing for proper stack configuration.

1. On each switch, enable stacking:

```
VFSM(config)# boot stack enable
```

2. On each switch, set the stacking membership mode.

By default, each switch is set to Member mode. However, one switch must be set to Master mode. Use the following command on only the designated Master switch:

```
VFSM(config)# boot stack mode master
```

**Note:** If any Member switches are incorrectly set to Master mode, use the `mode member` option to set them back to Member mode.

3. On each switch, configure the stacking VLAN (or use the default setting).

Although any VLAN (except VLAN 1) may be defined for stack traffic, it is highly recommended that the default, VLAN 4090 as shown below, be reserved for stacking.

```
VFSM(config)# boot stack vlan 4090
```

4. On each switch, designate the stacking links.

To create the recommended topology, at least two 10Gb external ports on each switch should be dedicated to stacking. By default, 10Gb Ethernet ports EXT1 and EXT2 are used.

Use the following command to specify the links to be used in the stacking trunk:

```
VFSM(config)# boot stack hlgig-trunk <list of port names or aliases>
```

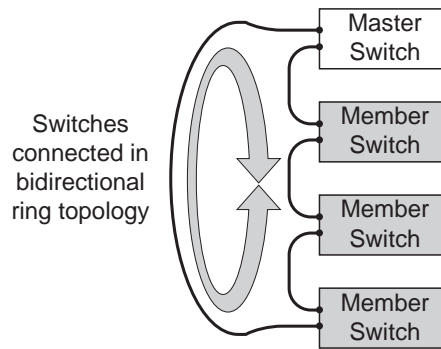
5. On each switch, perform a reboot:

```
VFSM(config)# reload
```

6. Physically connect the stack trunks.

To create the recommended topology, attach the two designated stacking links in a bidirectional ring. As shown in [Figure 20](#), connect each switch in turn to the next, starting with the Master switch. To complete the ring, connect the last Member switch back to the Master.

Figure 20. Example of Stacking Connections



**Note:** The stacking feature is designed such that the stacking links in a ring topology do not result in broadcast loops. The stacking ring is thus valid (no stacking links are blocked), even when Spanning Tree protocol is enabled.

Once the stack trunks are connected, the switches will perform low-level stacking configuration.

**Note:** Although stack link failover/failback is accomplished on a sub-second basis, to maintain the best stacking operation and avoid traffic disruption, it is recommended not to disrupt stack links after the stack is formed.

## Additional Master Configuration

Once the stack links are connected, access the internal management IP interface of the Master switch (assigned by the management system) and complete the configuration.

## Configuring an External IPv4 Address for the Stack

In addition to the internal management IP interface assigned to the Master switch by the management system, a standard switch IP interface can be used for connecting to and managing the stack externally. Configure an IP interface with the following:

- Stack IPv4 address and mask
- IPv4 default gateway address
- VLAN number used for external access to the stack (rather than the internal VLAN 4090 used for inter-stack traffic)

For an IPv4 network, use the following commands:

```
VFSM(config)# ip gateway <gateway number> address <gateway IPv4 address>
VFSM(config)# ip gateway <gateway number> enable
VFSM(config)# interface ip <IP interface number>
VFSM(config-ip-if) ip address <stack IPv4 address>
VFSM(config-ip-if) ip netmask <IPv4 subnet mask>
VFSM(config-ip-if) vlan <VLAN ID>
VFSM(config-ip-if) enable
VFSM(config-ip-if) exit
```

One completed, stack management can be performed via Telnet or BBI (if enabled) from any point in the configured VLAN, using the IPv4 address of the configured IP interface.

In the event that the Master switch fails, if a Backup switch is configured (see [“Assigning a Stack Backup Switch” on page 206](#)), the external IP interface for the stack will still be available. Otherwise, the administrator must manage the stack through the internal management IP interface assigned to the Backup switch by the management system.

## Locating an External Stack Interface

If the IPv4 address and VLAN of an external IP interface for the stack is unknown, connect to the Master switch using the IPv4 address assigned by the management system, and execute the following command:

```
VFSM(config)# show interface ip
```

## Viewing Stack Connections

To view information about the switches in a stack, execute the following command:

```
VFSM(config)# show stack switch
```

Stack name: STK  
Local switch is the master.

Local switch:

- csnum - 1
- MAC - 00:00:00:00:01:00
- UUID - 0d258000000000000000000000000000145ee06000
- Bay Number - 7
- Switch Type - 7 (1-10GbESM)
- Chassis Type - 2 (BladeCenter H)
- Switch Mode (cfg) - Master
- Priority - 225
- Stack MAC - 00:00:00:00:01:1f

Master switch:

- csnum - 1
- MAC - 00:00:00:00:01:00
- UUID - 0d258000000000000000000000000000145ee06000
- Bay Number - 7

Backup switch:

- csnum - 2
- MAC - 00:22:00:ad:43:00
- UUID - 0d258000000000000000000000000000145ee06000
- Bay Number - 8

Configured Switches:

csnum	UUID	BAY	MAC	asnum
C1	0d258000000000000000000000000000145ee06000	7	00:00:00:00:01:00	A1
C2	0d258000000000000000000000000000145ee06000	8	00:22:00:ad:43:00	A3
C3	0d258000000000000000000000000000145ee06000	9	00:11:00:af:ce:00	A2

Attached Switches in Stack:

asnum	UUID	BAY	MAC	csnum	State
A1	0d258000000000000000000000000000145ee06000	7	00:00:00:00:01:00	C1	IN_STACK
A2	0d258000000000000000000000000000145ee06000	7	00:11:00:af:ce:00	C3	IN_STACK
A3	0d258000000000000000000000000000145ee06000	7	00:22:00:ad:43:00	C2	IN_STACK

## Binding Members to the Stack

You can bind Member switches to a stack `csnum` using either their `asnum` or its chassis UUID and bay number:

```
VFSM(config)# stack switch-number <csnum> universal-unic-id <chassis UUID>
VFSM(config)# stack switch-number <csnum> bay <bay number (1-4)>

-Or-

VFSM(config)# stack switch-number <csnum> bind <asnum (1-16)>
```

To remove a Member switch, execute the following command:

```
VFSM(config)# no stack switch-number <csnum>
```

## Assigning a Stack Backup Switch

To define a Member switch as a Backup (optional) which will assume the Master role if the Master switch should fail, execute the following command:

```
VFSM(config)# stack backup <csnum>
```

---

## Managing a Stack

### Managing through the Stack Master

The stack is managed primarily through the Master switch. The Master switch then pushes configuration changes and run-time information to the Member switches.

Use Telnet or the Browser-Based Interface (BBI) to access the Master, as follows:

- Use the management IP address assigned to the Master by the management system.
- On any switch in the stack, connect to any port that is not part of an active trunk, and use the IP address of any IP interface to access the stack.

**Note:** Protected mode is automatically turned on in stacking mode. Stack can be managed using an external port or using the management IP address assigned to the Master by the management system. See [“Protected Mode” on page 80](#).

### Connecting to Member Switches

Though the stack is managed primarily through the Master switch, limited management functions are available on any Member switch in the stack.

Using Telnet to access the management IP address of any Member switch, the administrator can view a wide variety of switch information and can change some of the Member stacking properties.

Other forms of access (such as BBI or SNMP) are not available through Member switch IP management interfaces.

### Rebooting Stacked Switches using the ISCLI

The administrator can reboot individual switches in the stack, or the entire stack using the following commands:

VFSM(config)# reload	<i>(Reboot all switches in the stack)</i>
VFSM(config)# reload master	<i>(Reboot only the stack Master)</i>
VFSM(config)# reload switch <csnum list>	<i>(Reboot only the listed switches)</i>

## Rebooting Stacked Switches using the BBI

The **Configure > System > Config/Image Control** window allows the administrator to perform a reboot of individual switches in the stack, or the entire stack. The following table describes the stacking Reboot buttons.

Table 19. Stacking Boot Management buttons

Field	Description
Reboot Stack	Performs a software reboot/reset of all switches in the stack. The software image specified in the Image To Boot drop-down list becomes the active image.
Reboot Master	Performs a software reboot/reset of the Master switch. The software image specified in the Image To Boot drop-down list becomes the active image.
Reboot Switches	Performs a reboot/reset on selected switches in the stack. Select one or more switches in the drop-down list, and click Reboot Switches. The software image specified in the Image To Boot drop-down list becomes the active image.

The **Update Image/Cfg** section of the window applies to the Master. When a new software image or configuration file is loaded, the file first loads onto the Master, and the Master pushes the file to all other switches in the stack, placing it in the same software or configuration bank as that on the Master. For example, if the new image is loaded into image 1 on the Master switch, the Master will push the same firmware to image 1 on each Member switch.

---

## Upgrading Software in an Existing Stack

Upgrade all stacked switches at the same time. The Master controls the upgrade process. Use the following procedure to perform a software upgrade for a stacked system.

1. Load new software on the Master.

The Master pushes the new software image to all Members in the stack, as follows:

- If the new software is loaded into image 1, the Master pushes the software into image 1 on all Members.
- If loaded into image 2, the Master pushes the software into image 2 on all Members.

The software push can take several minutes to complete.

2. Verify that the software push is complete. Use either the BBI or the ISCLI:

- From the BBI, go to Dashboard > Stacking > Push Status and view the Image Push Status Information, or
- From the ISCLI, use following command to verify the software push:

```
VFSM(config)# show stack push-status

Image 1 transfer status info:
  Switch 00:16:60:f9:33:00:
    last receive successful
  Switch 00:17:ef:c3:fb:00:
    not received - file not sent or transfer in progress

Image 2 transfer status info:
  Switch 00:16:60:f9:33:00:
    last receive successful
  Switch 00:17:ef:c3:fb:00:
    last receive successful

Boot image transfer status info:
  Switch 00:16:60:f9:33:00:
    last receive successful
  Switch 00:17:ef:c3:fb:00:
    last receive successful

Config file transfer status info:
  Switch 00:16:60:f9:33:00:
    last receive successful
  Switch 00:17:ef:c3:fb:00:
    last receive successful
```

3. Reboot all switches in the stack. Use either the CLI or the BBI.

- From the BBI, select Configure > System > Config/Image Control. Click Reboot Stack.
- From the CLI, use the following command:

```
VFSM(config)# reload
```

4. Once the switches in the stack have rebooted, verify that all of them are using the same version of firmware. Use either the CLI or the BBI.
  - From the BBI, open Dashboard > Stacking > Stack Switches and view the Switch Firmware Versions Information from the Attached Switches in Stack.
  - From the CLI, use the following command:

```
VFSM(config)# show stack version
Switch Firmware Versions:
-----
asnum  csnun      MAC           S/W   Version  Serial #
-----
A1     C1         00:00:00:00:01:00 image1 0.0.0.0 CH49000000
A2     C2         00:11:00:af:ce:00 image1 0.0.0.0 CH49000001
A3           00:22:00:ad:43:00 image1 0.0.0.0 CH49000002
```

---

## Replacing or Removing Stacked Switches

Stack switches may be replaced or removed while the stack is in operation. However, the following conditions must be met in order to avoid unnecessary disruption:

- If removing an active Master switch, make sure that a valid Backup exists in the stack.
- It is best to replace only one switch at a time.
- If replacing or removing multiple switches in a ring topology, when one switch has been properly disconnected (see the procedures that follow), any adjacent switch can also be removed.
- Removing any two, non-adjacent switches in a ring topology will divide the ring and disrupt the stack.

Use the following procedures to replace a stack switch.

### Removing a Switch from the Stack

1. Make sure the stack is configured in a ring topology.

**Note:** When an open-ended daisy-chain topology is in effect (either by design or as the result of any failure of one of the stacking links in a ring topology), removing a stack switch from the interior of the chain can divide the chain and cause serious disruption to the stack operation.

2. If removing a Master switch, make sure that a Backup switch exists in the stack, then turn off the Master switch.

This will force the Backup switch to assume Master operations for the stack.

3. Remove the stack link cables from the old switch only.
4. Disconnect all network cables from the old switch only.
5. Remove the old switch from the chassis.

### Installing the New Switch or Healing the Topology

If using a ring topology, but not installing a new switch for the one removed, close the ring by connecting the open stack links together, essentially bypassing the removed switch.

Otherwise, if replacing the removed switch with a new unit, use the following procedure:

1. Make sure the new switch meets the stacking requirements on [page 196](#).
2. Place the new switch in its determined place according to the *Virtual Fabric Switch Module Installation Guide*.
3. Connect to the CLI of the new switch (not the stack interface)
4. Enable stacking:

```
VFSM(config)# boot stack enable
```

5. Set the stacking mode.

By default, each switch is set to Member mode. However, if the incoming switch has been used in another stacking configuration, it may be necessary to ensure the proper mode is set.

- If replacing a Member or Backup switch:

```
VFSM(config)# boot stack mode member
```

- If replacing a Master switch:

```
VFSM(config)# boot stack mode master
```

6. Configure the stacking VLAN on the new switch, or use the default setting.

Although any VLAN may be defined for stack traffic, it is highly recommended that the default, VLAN 4090, be reserved for stacking, as shown in the following command.

```
VFSM(config)# boot stack vlan 4090
```

7. Designate the stacking links.

It is recommended that you designate the same number of external 10Gb ports for stacking as the switch being replaced. At least one 10Gp port is required. Use the following command to specify the links to be used in the stacking trunk:

```
VFSM(config)# boot stack hlgig-trunk <list of external port s>
```

8. Attach the required stack link cables to the designated stack links on the new switch.

9. Attach the desired network cables to the new switch.

10. Reboot the new switch:

```
VFSM(config)# reload
```

When the new switch boots, it will join the existing stack. Wait for this process to complete.

## Binding the New Switch to the Stack

1. Log in to the stack interface.

**Note:** If replacing the Master switch, be sure to log in to the stack interface (hosted temporarily on the Backup switch) rather than logging in directly to the newly installed Master.

2. From the stack interface, assign the `csnum` for the new switch.

You can bind Member switches to a stack `csnum` using either the new switch's `asnum` or its chassis UUID and bay number:

```
VFSM(config)# stack switch-number <csnum> universal-unic-id <uuid> bay <Slot ID>
```

-or-

```
VFSM(config)# stack switch-number <csnum> bind <asnum>
```

3. Apply and save your configuration changes.

**Note:** If replacing the Master switch, the Master will not assume control from the Backup unless the Backup is rebooted or fails.

---

## ISCLI Stacking Commands

Stacking-related ISCLI commands are listed below. For details on specific commands, see the *Virtual Fabric Switch Module ISCLI Reference*.

- [no] boot stack enable
- boot stack hlgig-trunk <port list>
- boot stack mode master|member
- boot stack push-image boot-image|image1|image2 <asnum>
- boot stack vlan <VLAN> <asnum>|master|backup|all
- default boot stack <asnum>|master|backup|all
- [no] logging log stacking
- no stack backup
- no stack name
- no stack switch-number <csnum>
- show boot stack <asnum>|master|backup|all
- show stack attached-switches
- show stack backup
- show stack dynamic
- show stack link
- show stack name
- show stack path-map [<csnum>]
- show stack push-status
- show stack switch
- show stack switch-number [<csnum>]
- show stack version
- stack backup <csnum>
- stack name <word>
- stack switch-number <csnum> bind <asnum>
- stack switch-number <csnum> universal-unic-id <chassis UUID>  
[bay <bay number>]
- stack switch-number <csnum> bay <bay number>  
[universal-unic-id <chassis UUID>]

---

## Chapter 12. Virtualization

Virtualization allows resources to be allocated in a fluid manner based on the logical needs of the data center, rather than on the strict, physical nature of components. The following virtualization features are included in IBM Networking OS 7.8 on the Virtual Fabric Switch Module (VFSM):

- Virtual Local Area Networks (VLANs)  
VLANs are commonly used to split groups of networks into manageable broadcast domains, create logical segmentation of workgroups, and to enforce security policies among logical network segments.  
For details on this feature, see [“VLANs” on page 123](#).
- Port trunking  
A port trunk pools multiple physical switch ports into a single, high-bandwidth logical link to other devices. In addition to aggregating capacity, trunks provides link redundancy.  
For details on this feature, see [“Ports and Trunking” on page 143](#).
- Stacking  
Multiple switches (from the same or different chassis) can be aggregated into a single super-switch, combining port capacity while at the same time simplifying their management. IBM N/OS 7.8 supports one stack with up to eight switches.  
For details on this feature, see [“Stacking” on page 195](#).
- Virtual Network Interface Card (vNIC) support  
Some NICs, such as the Emulex Virtual Fabric Adapter for IBM BladeCenter, can virtualize NIC resources, presenting multiple virtual NICs to the server’s OS or hypervisor. Each vNIC appears as a regular, independent NIC with some portion of the physical NIC’s overall bandwidth. N/OS 7.8 supports up to four vNICs over each internal switch port.  
For details on this feature, see [“Virtual NICs” on page 217](#).
- VMready  
The switch’s VMready software makes it *virtualization aware*. Servers that run hypervisor software with multiple instances of one or more operating systems can present each as an independent *virtual machine* (VM). With VMready, the switch automatically discovers virtual machines (VMs) connected to switch.  
For details on this feature, see [“VMready” on page 237](#).

N/OS virtualization features provide a highly-flexible framework for allocating and managing switch resources.

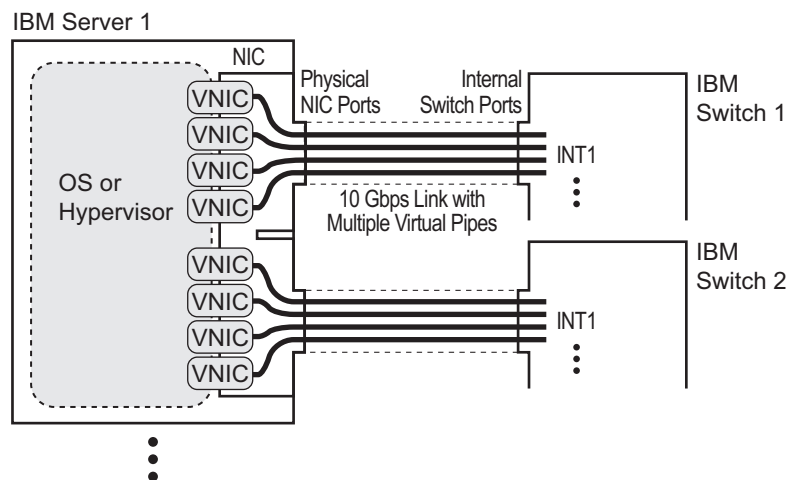


## Chapter 13. Virtual NICs

A Network Interface Controller (NIC) is a component within a blade server that allows the server to be connected to a network. The NIC provides the physical point of connection, as well as internal software for encoding and decoding network packets.

Virtualizing the NIC helps to resolve issues caused by limited NIC slot availability. By virtualizing a 10Gbps NIC, its resources can be divided into multiple logical instances known as virtual NICs (vNICs). Each vNIC appears as a regular, independent NIC to the server operating system or a hypervisor, with each vNIC using some portion of the physical NIC's overall bandwidth.

Figure 21. Virtualizing the NIC for Multiple Virtual Pipes on Each Link



A VFSM with IBM Networking OS 7.8 supports the Emulex Virtual Fabric Adapter (VFA) for IBM BladeCenter to provide the following vNIC features:

- Up to four vNICs are supported on each internal switch port.
- Each vNIC can accommodate one of the following traffic types: regular Ethernet, iSCSI, or Fibre Channel over Ethernet (FCoE).
- When using the Emulex Virtual Fabric Adapter for IBM BladeCenter, FCoE traffic is expected to occur only on vNIC pipe 2.
- vNICs with traffic of the same type can be grouped together, along with regular internal ports, external uplink ports, and trunk groups, to define vNIC groups for enforcing communication boundaries.
- In the case of a failure on the external uplink ports associated with a vNIC group, the switch can signal affected vNICs for failover while permitting other vNICs to continue operation.
- Each vNIC can be allocated a symmetric percentage of the 10Gbps bandwidth on the link (from NIC to switch, and from switch to NIC).
- The VFSM can be used as the single point of vNIC configuration.

The following restriction applies to vNICs:

- vNICs are not supported simultaneously with VM groups (see [“VMready” on page 237](#)) on the same switch ports.

By default, vNICs are disabled. The administrator can enable vNICs and configure vNIC features on the switch using the standard management options such as the IBM N/OS CLI, the ISCLI, and the Browser-based Interface (BBI).

To enable the vNIC feature on the switch, use the following command on the vNIC Configuration Menu:

```
VFSM(config)# vnic enable
```

**Note:** The Emulex Virtual Fabric Adapter for IBM BladeCenter can also operate in Physical NIC (PNIC) mode, in which case vNIC features are non-applicable.

---

## vNIC IDs

### vNIC IDs on the Switch

N/OS 7.8 supports up to four vNICs attached to each internal switch port. Each vNIC is provided its own independent virtual pipe on the port.

On stand-alone (non-stacked) switches, each vNIC is identified by port and vNIC number:

*<port number or alias> . <vNIC pipe number (1-4)>*

For example:

INT1.1, INT1.2, INT1.3, and INT1.4 represent the vNICs on port INT1.

INT2.1, INT2.2, INT2.3, and INT2.4 represent the vNICs on port INT2, etc.

These vNIC IDs are used when adding vNICs to vNIC groups, and are shown in some configuration and information displays.

Whenever switches are stacked, the switch `csnum` ID is also required:

*<switch csnum> : <port number or alias> . <vNIC pipe number (1-4)>*

For example:

INT4 : 2 . 3 refers to switch 4, port 2, vNIC 3.

**Note:** The configuration examples in this chapter depict stand-alone (non-stacked) port and vNIC designations.

## vNIC Interface Names on the Server

When running in virtualization mode, the Emulex Virtual Fabric Adapter presents eight vNICs to the OS or hypervisor (four for each of the two physical NIC ports). Each vNIC is identified in the OS or hypervisor with a different PCIe function number (0-7). PCIe function numbers correlate to vNIC IDs on the switch as follows:

Table 20. vNIC ID Correlation

PCIe Function ID	NIC Port	Switch Slot	vNIC Pipe	vNIC ID
0	0	Bay 7	1	INT <sub>x</sub> .1
2	0	Bay 7	2	INT <sub>x</sub> .2
4	0	Bay 7	3	INT <sub>x</sub> .3
6	0	Bay 7	4	INT <sub>x</sub> .4
1	1	Bay 9	1	INT <sub>x</sub> .1
3	1	Bay 9	2	INT <sub>x</sub> .2
5	1	Bay 9	3	INT <sub>x</sub> .3
7	1	Bay 9	4	INT <sub>x</sub> .4

In this, the *x* in the vNIC ID represents the internal switch port to which the NIC port is connected. Each physical NIC port is connected to a different switch bay in the blade chassis.

---

## vNIC Bandwidth Metering

N/OS 7.8 supports bandwidth metering for vNIC traffic. By default, each of the four vNICs on any given port is allowed an equal share (25%) of NIC capacity when enabled. However, you may configure the percentage of available switch port bandwidth permitted to each vNIC.

vNIC bandwidth can be configured as a value from 1 to 100, with each unit representing 1% (or 100Mbps) of the 10Gbps link. By default, each vNICs enabled on a port is assigned 25 units (equal to 25% of the link, or 2.5Gbps). When traffic from the switch to the vNIC reaches its assigned bandwidth limit, the switch will drop packets egressing to the affected vNIC.

**Note:** Bandwidth metering drops excess packets when configured limits are reached. Consider using the ETS feature in applications where packet loss is not desirable (see [“Enhanced Transmission Selection” on page 281](#)).

To change the bandwidth allocation, use the following commands:

```
VFSM(config)# vnic port <port alias or number> index <vNIC number (1-4)>
VFSM(vnic-config)# bandwidth <allocated percentage>
```

**Note:** vNICs that are disabled are automatically allocated a bandwidth value of 0.

A combined maximum of 100 units can be allocated among vNIC pipes enabled for any specific port (bandwidth values for disabled pipes are not counted). If more than 100 units are assigned to enabled pipes, an error will be reported when attempting to apply the configuration.

The bandwidth metering configuration is automatically synchronized between the switch and vNICs for regular Ethernet and iSCSI traffic. Once configured on the switch, there is no need to manually configure vNIC bandwidth metering limits on the NIC.

**Note:** FCoE vNIC does not use egress metering. ETS and PFC must be enabled to ensure lossless transmission for FCoE traffic. ETS does traffic shaping. You can configure a minimum bandwidth for each traffic class. For example, 40% for FCoE priority 3, 60% for the Ethernet traffic. FCoE traffic gets 40% minimum guaranteed bandwidth. If the Ethernet traffic only consumes 30% bandwidth, then FCoE traffic can use 70%. If there is no other Ethernet traffic, then FCoE traffic can use 100%. The FCoE vNIC can use up to 100% of the bandwidth, with a minimum guaranteed bandwidth of 40%.

---

## vNIC Uplink Modes

The switch supports two modes for configuring the vNIC uplinks: dedicated mode and shared mode. The default is the dedicated mode. To enable the shared mode, enter the following command:

```
VFSM(config)# vnic uplink-share
```

In the dedicated mode, only one vNIC group is assigned to an uplink port. This port can be a regular port or a trunk port. The NIC places an outer tag on the vNIC group packets. This outer tag contains the vNIC group VLAN. The uplink NIC strips off the outer tag before sending out the packet. For details, see [“vNIC Groups in Dedicated Mode” on page 226](#).

In the shared mode, multiple vNIC groups can be assigned to an uplink port. This port can be a regular port or a trunk port. The vNIC groups share the uplink. You may assign a few vNIC groups to share an uplink and the other vNIC groups to have a single uplink each. In either case, the switch still operates in shared mode. As in the dedicated mode, the NIC places an outer tag on the vNIC group packets. This outer tag contains the vNIC group VLAN. The uplink NIC does not strip off the outer tag. The vNIC group tag defines the regular VLAN for the packet. This behavior is particularly useful in cases where the downstream server does not set any tag. Effectively, each vNIC group is a VLAN, which you can assign by configuring the VLAN to the vNIC group. You must enable the tag configuration on the uplink port. For details, see [“vNIC Groups in Shared Mode” on page 227](#).

The table below compares the configurations of the two modes.

*Table 21. Comparison: Dedicated Mode vs. Shared Mode*

Configuration Area	Dedicated Mode	Shared Mode
Port	“tagpvid” must be disabled.	“tagpvid” is user configurable.
	“pvid” = vNIC group VLAN.	“pvid” is user configurable.
	“tag” is user configurable.	“tag” must be enabled.
	Port can be added only to the vNIC group VLAN.	Port can be added to multiple VLANs in addition to the vNIC group VLANs that are automatically configured.
	Inserts vNIC group VLAN in the outer tag of ingress packets.	Inserts regular VLAN in the outer tag. VLAN tags are passed to and received from the uplink switch similar to vNIC ports.
		To handle untagged packets, configure the pvid/native VLAN of the uplink port to one of the vNIC group VLANs, and disable “tag-pvid”.
VLAN	Add the port to a vNIC group VLAN and delete it from any other VLAN when the vNIC group VLAN is enabled.	Add the port to all vNIC group VLANs that are sharing the port. Do not remove it from any other VLAN.
	Delete the port from the vNIC group VLAN and add it back to the default VLAN 1 when the vNIC group is disabled/deleted or when the vNIC feature is globally disabled.	Remove the port from a vNIC group VLAN when the vNIC group is disabled/deleted. When the vNIC feature is globally disabled or the port is not added in any vNIC group, remove the port from all vNIC group VLANs and add it back to default VLAN 1 if no non-vNIC VLAN exists on the port.
	Do not add a port or trunk to multiple vNIC groups that are enabled.	Can add a port or trunk to multiple vNIC groups that are enabled.
	Do not configure additional VLANs on the uplink ports.	Can configure additional VLANs on the uplink ports.
STP	An uplink port can only be in one STG.	An uplink port can be in multiple STGs.
	When you add a port to a vNIC group, STP is automatically disabled.	When you add a port to a vNIC group, STP is automatically disabled.
	When you remove a port from a vNIC group, STP is automatically reset to factory default.	When you remove a port from a vNIC group, STP is automatically reset to factory default.
Failover	An uplink up/event can trigger the failover state change only of one vNIC group.	An uplink up/event can trigger the failover state change of multiple vNIC groups.

---

## LACP Trunks

The uplink used by vNIC groups can be a regular port, static trunk, or a dynamic trunk. If you are using a dynamic trunk for the uplink, you must configure the same LACP admin key for the vNIC uplink ports you want to group as a trunk on the upstream switch. You cannot connect these vNIC uplink ports to multiple upstream switches unless these switches are a single logical switch from LAG perspective.

Use the command below to add the LACP trunk as the uplink for a vNIC group.

```
VFSM(config)# vnic vnicgroup <vNIC group number> key <LACP admin key>
```

---

## vNIC Groups

vNICs can be grouped together, along with internal and external switch ports and trunks, into vNIC groups. Each vNIC group is essentially a separate virtual network within the switch. Elements within a vNIC group have a common logical function and can communicate with each other, while elements in different vNIC groups are separated.

N/OS 7.8 supports up to 32 independent vNIC groups. To enforce group boundaries, each vNIC group is assigned its own unique VLAN.

The VLAN configured for the vNIC group will be automatically assigned to member vNICs, ports, and trunks and should not be manually configured for those elements.

**Note:** Once a VLAN is assigned to a vNIC group, that VLAN is used only for vNIC purposes and is no longer available for configuration. Likewise, any VLAN configured for regular purposes cannot be configured as a vNIC group VLAN.

The vNIC group rules are as follows:

- vNIC groups may have one or more vNIC members. However, any given vNIC can be a member of only one vNIC group.
- All vNICs on a given port must belong to different vNIC groups.
- Each vNIC group may contain traffic of one type only (either regular Ethernet, iSCSI, or FCoE). Traffic of different types may not be mixed within any vNIC group.
- External ports that are part of a trunk may not be individually added to a vNIC group. Only one individual external port, one static trunk, or one dynamic trunk (consisting of multiple external ports) may be added to any given vNIC group.
- In dedicated mode, for any internal ports, external port, or port trunk group connected to regular (non-vNIC) devices:
  - These elements can be placed in only one vNIC group (they cannot be members of multiple vNIC groups).
  - Once added to a vNIC group, the PVID/Native VLAN for the element is automatically set to use the vNIC group VLAN number, and PVID/Native VLAN tagging on the element is automatically disabled.
- By default, STP is disabled on any external port added to a vNIC group. STP can be re-enabled on the port if desired.

- Because regular, inner VLAN IDs are ignored by the switch for traffic in vNIC groups, following rules and restrictions apply:
  - The inner VLAN tag may specify any VLAN ID in the full, supported range (2 to 4094) and may even duplicate outer vNIC group VLAN IDs. However, in the shared mode, inner VLAN tag and the vNIC group VLAN ID should be the same.
  - Per-VLAN IGMP snooping is not supported in vNIC groups.
  - The inner VLAN tag is not processed in any way in vNIC groups: The inner tag cannot be stripped or added on port egress, is not used to restrict multicast traffic, is not matched against ACL filters, and does not influence Layer 3 switching.
  - For vNIC ports on the switch, because the outer vNIC group VLAN is transparent to the OS/hypervisor and upstream devices, VLAN tagging/trunk mode should be configured as normally required (on or off) for the those devices, ignoring any outer tag.
- Virtual machines (VMs) and other VEs associated with vNICs are automatically detected by the switch when VMready is enabled (see [“VMready” on page 237](#)). However, vNIC groups are isolated from other switch elements. VEs in vNIC groups cannot be assigned to VM groups.

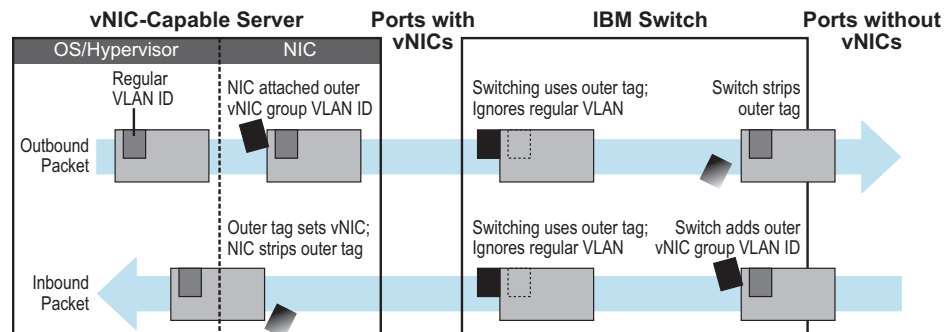
Configuration for vNIC groups is accomplished using the vNIC Group Configuration menu:

```
VFSM(config)# vnic vnicgroup <vNIC group number>
```

## vNIC Groups in Dedicated Mode

The vNIC group VLAN ID is placed on all vNIC group packets as an “outer” tag. As shown in [Figure 22](#), the outer vNIC group VLAN ID is placed on the packet in addition to any regular VLAN tag assigned by the network, server, or hypervisor. The outer vNIC group VLAN is used only between the VFSM and the NIC.

Figure 22. Outer and Inner VLAN Tags



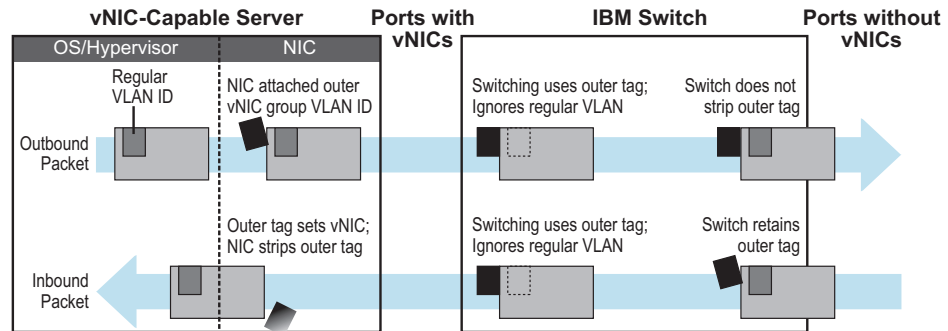
Within the VFSM, all Layer 2 switching for packets within a vNIC group is based on the outer vNIC group VLAN. The VFSM does not consider the regular, inner VLAN ID (if any) for any VLAN-specific operation.

The outer vNIC group VLAN is removed by the NIC before the packet reaches the server OS or hypervisor, or by the switch before the packet egresses any internal port or external uplink port which does not need it for vNIC processing.

## vNIC Groups in Shared Mode

The vNIC group VLAN ID is placed on all vNIC group packets as an “outer” tag. As shown in Figure 23, the outer vNIC group VLAN ID is placed on the packet in addition to any regular VLAN tag assigned by the network, server, or hypervisor.

Figure 23. Outer and Inner VLAN Tags



Within the VFSM, all Layer 2 switching for packets within a vNIC group is based on the outer vNIC group VLAN. The VFSM does not consider the regular, inner VLAN ID (if any) for any VLAN-specific operation.

The outer vNIC group VLAN is not removed by the switch before the packet egresses any internal port or external uplink port. For untagged packets sent by the server, the uplink NIC uses this outer tag to switch the packet to destined VLAN.

The shared mode is useful in cases where the multiple vNIC groups need to share an uplink port. The vNIC group tag defines the user VLAN. Following is an use case:

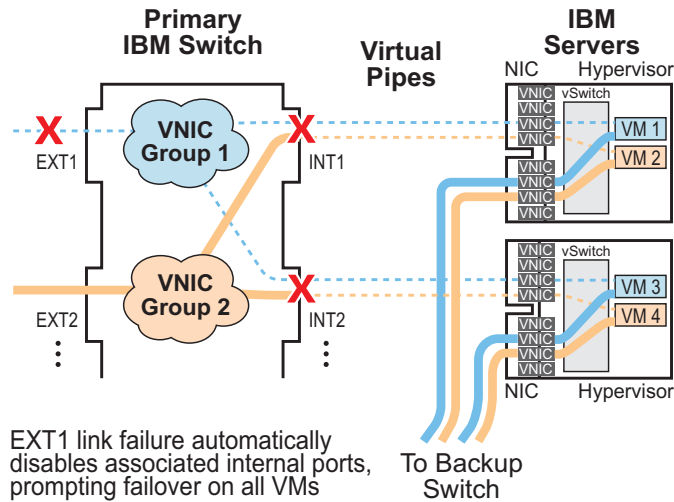
An ESX server is presented with eight vNICs (four from bay 7 and four from bay 9) used with four virtual switches of the ESX host and with no tagged port groups. A pair of odd/even vNICs is placed within each virtual switch. On the VFSM, four vNIC groups are created and the desired VLAN for each vNIC group is configured. For example, if vNIC group 1 on the VFSM has four interfaces: 1.1, 2.1, 3.1, 4.1. vNIC group 1 is configured with VLAN 10. Packets coming from any VM connecting with the virtual switch that VMNIC 2 and 3 (vNIC 1.1, 2.1, 3.1, and 4.1 on bay 7 and bay 9) will be assigned with VLAN 10. These packets go out the uplink with VLAN 10 tag. The upstream switch sends these packets to the desired destination on VLAN 10.

## vNIC Teaming Failover

For NIC failover in a non-virtualized environment, when a service group's external uplink ports fail or are disconnected, the switch disables the affected group's internal ports, causing the server to failover to the backup NIC and switch.

However, in a virtualized environment, disabling the affected internal ports would disrupt all vNIC pipes on those ports, not just those that have lost their external uplinks (see [Figure 24](#)).

Figure 24. Regular Failover in a Virtualized Environment

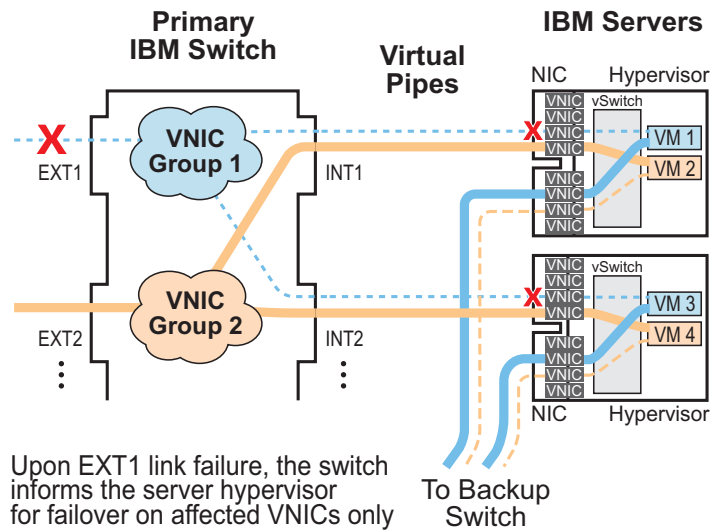


To avoid disrupting vNICs that have not lost their external uplinks, N/OS 7.8 and the Emulex Virtual Fabric Adapter for IBM BladeCenter provide vNIC-aware failover.

In the dedicated mode, when a vNIC group's external uplink ports fail, the switch cooperates with the affected NIC to prompt failover only on the appropriate vNICs. This allows the vNICs that are not affected by the failure to continue without disruption (see [Figure 25 on page 229](#)).

In the shared mode, when the shared external uplink port fails, the switch prompts failover to all vNICs, belonging to multiple vNIC groups, that are affected by the failure. The vNIC groups not connected to the failed link will not be affected.

Figure 25. vNIC Failover Solution



By default, vNIC Teaming Failover is disabled on each vNIC group, but can be enabled or disabled independently for each vNIC group using the following commands:

```
VFSM(config)# vnic vnicgroup <group number>
VFSM(vnic-group-config)# failover
```

## vNIC Configuration Example

Consider the following example configuration of vNICs for regular Ethernet traffic:

Figure 26. Multiple vNIC Groups

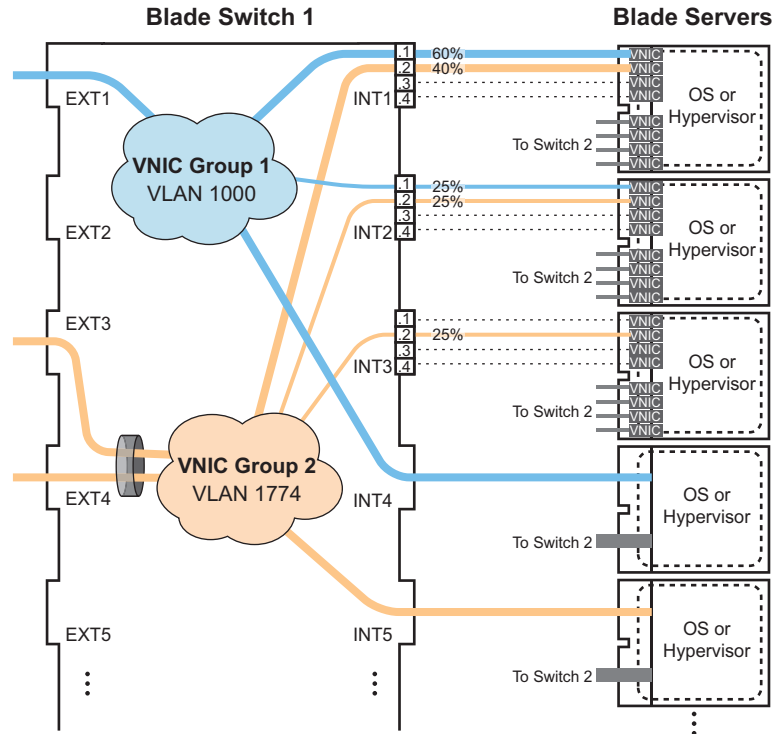


Figure 26 has the following vNIC network characteristics:

- vNIC group 1 has an outer tag for VLAN 1000. The group is comprised of vNIC pipes INT1.1 and INT2.1, internal port INT4 (a non-vNIC port), and external uplink port EXT1.
- vNIC group 2 has an outer tag for VLAN 1774. The group is comprised of vNIC pipes INT1.2, INT2.2 and INT3.2, internal port INT5, and an external uplink trunk of ports EXT3 and EXT4.
- vNIC failover is enabled for both vNIC groups.
- vNIC bandwidth on port INT1 is set to 60% for vNIC 1 and 40% for vNIC 2.
- Other enabled vNICs (INT2.1, INT2.2, and INT3.2) are permitted the default bandwidth of 25% (2.5Gbps) on their respective ports.
- All remaining vNICs are disabled (by default) and are automatically allocated 0 bandwidth.

1. Configure the external trunk to be used with vNIC group 2.

```
VFSM(config)# portchannel 1 port EXT3,EXT4 enable
```

2. Enable the vNIC feature on the switch.

```
VFSM(config)# vnic enable
```

### 3. Configure the virtual pipes for the vNICs attached to each internal port:

```
VFSM(config)# vnic port INTA1 index 1      (Select vNIC 1 on the port)
VFSM(vnic-config)# enable                  (Enable the vNIC pipe)
VFSM(vnic-config)# bandwidth 60            (Allow 60% egress bandwidth)
VFSM(vnic-config)# exit
VFSM(config)# vnic port INTA1 index 2      (Select vNIC 2 on the port)
VFSM(vnic-config)# enable                  (Enable the vNIC pipe)
VFSM(vnic-config)# bandwidth 40            (Allow 40% egress bandwidth)
VFSM(vnic-config)# exit

VFSM(config)# vnic port INTA2 index 1      (Select vNIC 1 on the port)
VFSM(vnic-config)# enable                  (Enable the vNIC pipe)
VFSM(vnic-config)# exit
VFSM(config)# vnic port INTA2 index 2      (Select vNIC 2 on the port)
VFSM(vnic-config)# enable                  (Enable the vNIC pipe)
VFSM(vnic-config)# exit
```

As a configuration shortcut, vNICs do not have to be explicitly enabled in this step. When a vNIC is added to the vNIC group (in the next step), the switch will prompt you to confirm automatically enabling the vNIC if it is not yet enabled (shown for INT3.2).

**Note:** vNICs are not supported simultaneously on the same switch ports as VMready.

### 4. Add ports, trunks, and virtual pipes to their vNIC groups.

```
VFSM(config)# vnic vnicgroup 1             (Select vNIC group)
VFSM(vnic-group-config)# vlan 1000         (Specify the VLAN)
VFSM(vnic-group-config)# member INTA1.1    (Add vNIC pipes to the group)
VFSM(vnic-group-config)# member INTA2.1
VFSM(vnic-group-config)# port INTA4        (Add non-vNIC port to the group)
VFSM(vnic-group-config)# port EXT1         (Add uplink port to the group)
VFSM(vnic-group-config)# failover          (Enable vNIC failover for the group)
VFSM(vnic-group-config)# enable            (Enable the vNIC group)
VFSM(vnic-group-config)# exit

VFSM(config)# vnic vnicgroup 2
VFSM(vnic-group-config)# vlan 1774
VFSM(vnic-group-config)# member INTA1.2
VFSM(vnic-group-config)# member INTA2.2
VFSM(vnic-group-config)# member INTA3.2
vNIC 3.2 is not enabled.
Confirm enabling vNIC3.2 [y/n]: y
VFSM(vnic-group-config)# port INTA5
VFSM(vnic-group-config)# trunk 1
VFSM(vnic-group-config)# failover
VFSM(vnic-group-config)# enable
VFSM(vnic-group-config)# exit
```

Once VLAN 1000 and 1774 are configured for vNIC groups, they will not be available for configuration in the regular VLAN menus (`/cfg/l2/vlan`).

**Note:** vNICs are not supported simultaneously on the same switch ports as VMready.

## vNICs for iSCSI on Emulex Virtual Fabric Adapter

The N/OS vNIC feature works with standard network applications like iSCSI as previously described. However, the Emulex Virtual Fabric Adapter for IBM BladeCenter expects iSCSI traffic to occur only on a single vNIC pipe. When using the Emulex Adapter 2, only vNIC pipe 2 may participate in iSCSI.

To configure the switch for this solution, iSCSI traffic should be placed in its own vNIC group, comprised of the external uplink port leading to the iSCSI target, and the related <port>.2 vNIC pipes connected to the participating servers. For example:

1. Enable the vNIC feature on the switch.

```
VFSM # vnic enable
```

2. Configure the virtual pipes for the iSCSI vNICs attached to each internal port:

```
VFSM(config)# vnic port INTA1 index 2      (Select vNIC 2 on the server port)
VFSM(vnic_config)# enable                  (Enable the vNIC pipe)
VFSM(vnic_config)# exit
VFSM(config)# vnic port INTA2 index 2      (Select vNIC 2 on the server port)
VFSM(vnic_config)# enable                  (Enable the vNIC pipe)
VFSM(vnic_config)# exit
VFSM(config)# vnic port INTA3 index 2      (Select vNIC 2 on the server port)
VFSM(vnic_config)# enable                  (Enable the vNIC pipe)
VFSM(vnic_config)# exit
```

**Note:** vNICs are not supported simultaneously on the same switch ports as VMready.

3. Add ports and virtual pipes to a vNIC group.

```
VFSM(config)# vnic vnicgroup 1             (Select vNIC group)
VFSM(vnic-group-config)# vlan 1000         (Specify the VLAN)
VFSM(vnic-group-config)# member INTA1.2    (Add iSCSI vNIC pipes to the group)
VFSM(vnic-group-config)# member INTA2.2
VFSM(vnic-group-config)# member INTA3.2
VFSM(vnic-group-config)# port EXT1         (Add the uplink port to the group)
VFSM(vnic-group-config)# enable            (Enable the vNIC group)
VFSM(vnic-group-config)# exit
```

**Note:** For vNICs on the Emulex Virtual Fabric Adapter, iSCSI and FCoE are mutually exclusive. iSCSI and FCoE cannot be used at the same time.

## vNICs for FCoE on Emulex Virtual Fabric Adapter

N/OS vNICs support FCoE (see [“FCoE and CEE” on page 261](#)). Similar to the iSCSI application, when using the Emulex Virtual Fabric Adapter for IBM BladeCenter, FCoE traffic is expected to occur only on vNIC pipe 2. In this case, the additional vNIC configuration for FCoE support is minimal.

Consider an example where the Fibre Channel network is connected to an FCoE Forwarder (FCF) bridge module in Bay 5 of the blade chassis. The switch in Bay 7 is connected to the FCF via bridge ports BR5A through BR5D, and to an ENode on port INT1.

**Note:** An additional topology employing two VFSMs and two FCFs is also supported, with the switch in Bay 7 connected to the FCF in Bay 5, and the switch in Bay 9 connected to the FCF in Bay 3. Topologies using two switches and one FCF are not currently supported.

1. The following steps are required as part of the regular FCoE configuration (see [“FIP Snooping Configuration” on page 275](#)):
  - a. Set up the bridge module connection and reboot the switch.

**Note:** It is recommended to use a 40Gbps connection to the bridge module. Also, this configuration assumes the switch is located in Bay 7 of the blade chassis. A switch in Bay 9 would connect to the bridge module in Bay 3 instead.

- b. Disable STP on the bridge ports in all STGs in which they appear.
- c. Disable the FIP Snooping automatic VLAN creation.
- d. Disable FIP Snooping on all external ports not used for FCoE. FIP snooping should be enabled only on ports connected to an FCF or ENode.
- e. Turn on CEE and FIP Snooping.
- f. Manually configure the FCoE ports and VLAN: enable VLAN tagging/trunk mode on all FCoE ports, enable PVID/Native VLAN tagging on ENode ports (only for Emulex CNA BE 2), and place FCoE ports into a supported VLAN.

When CEE is turned on and the regular FCoE configuration is complete, FCoE traffic will be automatically assigned to PFC priority 3, and be initially allocated 50% of port bandwidth via ETS.

The following steps are specific to vNIC configuration.

2. On the NIC, ensure that FCoE traffic occurs on vNIC pipe 2 only. Refer to your Emulex Virtual Fabric Adapter documentation for details.
3. On the switch, enable the vNIC feature.

```
VFSM(config) # vnic enable
```

4. (Optional) For additional security, set the desired operation mode for FCoE ports:

```
VFSM(config)# fcoe fips port INT1 fcf-mode off
                                     (Select ENode port; Set as ENode connection)
VFSM(config)# fcoe fips port EXT4 fcf-mode on
                                     (Select FCF port; Set as FCF connection)
```

5. (Optional) Bandwidth metering:

The Emulex Virtual Fabric Adapter ignores ETS bandwidth metering configuration. Instead, for FCoE traffic, all bandwidth not previously assigned to the other vNIC pipes is automatically allocated to the FCoE pipe (vNIC pipe 2). For example: by default, pipes 1, 3, and 4 are allocated a total of 75% of the port's bandwidth, which leaves FCoE the remaining 25%. If there is no other vNIC traffic on the port, the FCoE vNIC will use 100% of the port capacity. The FCoE vNIC uses ETS and PFC for a lossless transmission.

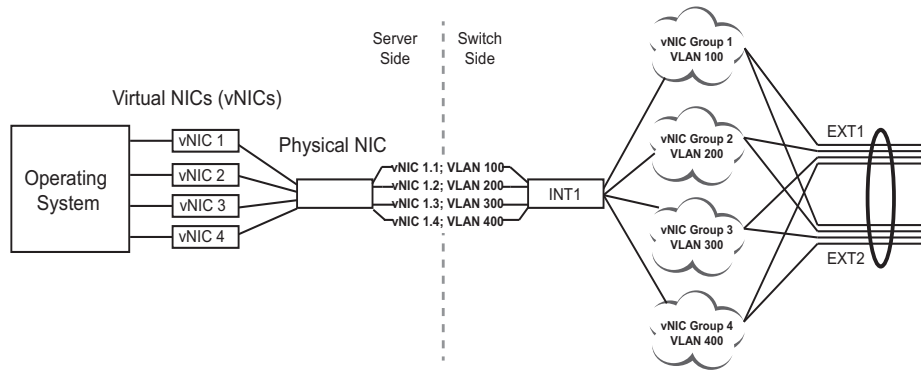
6. Apply and save the configuration.

No additional configuration for vNIC pipes or vNIC groups is required for FCoE. However, for other networks connected to the switch, appropriate vNIC pipes and vNIC groups should be configured as normal, if desired.

## vNIC Configuration Example - Shared Mode with LACP Trunk

The figure below shows an example configuration of a shared uplink mode with LACP trunk.

Figure 27. Shared Mode with LACP Trunk



The configuration overview of the network shown in [Figure 27](#) is as follows:

- The uplinks are in shared mode.
- There are four vNIC groups that belong to separate VLANs.
- Two external ports, EXT1 and EXT2, are dynamically trunked.
- The VLAN ID of the vNIC group is the actual VLAN that the vNIC belongs to.
- vNICs that belong to the same vNIC group will have the same VLAN in the upstream network.
- When packets egress the server, the NIC inserts a VLAN ID, without OS intervention, in the outer tag. The switch forwards the VLAN tag to the upstream network. The switch does not remove the outer tag.
- When switch receives ingress packets from upstream network, it forwards the packets to the vNICs based on the VLAN tag. The NIC removes the outer tag before sending the packet to the OS.
- The entire operation is transparent to the hypervisor/OS.

Following are the configuration steps for the example in [Figure 27](#).

1. Enable uplink sharing mode.

```
VFSM(config) # vnic uplink-share
```

2. Configure dynamic trunk and tagging/trunk mode on the external ports.

```
VFSM(config) # interface port ext1
VFSM(config-if) # lacp mode active
VFSM(config-if) # lacp key 50
```

Repeat the above steps for port EXT2.

3. Disable bandwidth metering on the uplink port.

```
VFSM(config) # no vnic egress-bw-meter
```

4. Configure vNIC group and assign LACP admin key.

```
VFSM(config) # no vnic vnicgroup 1
VFSM(vnic-group-config) # enable
VFSM(vnic-group-config) # vlan 100
VFSM(vnic-group-config) # failover
VFSM(vnic-group-config) # member int1.1
VFSM(vnic-group-config) # key 50
```

Repeat the above steps for vNIC groups 2, 3, and 4.

---

## Chapter 14. VMready

Virtualization is used to allocate server resources based on logical needs, rather than on strict physical structure. With appropriate hardware and software support, servers can be virtualized to host multiple instances of operating systems, known as virtual machines (VMs). Each VM has its own presence on the network and runs its own service applications.

Software known as a *hypervisor* manages the various virtual entities (VEs) that reside on the host server: VMs, virtual switches, and so on. Depending on the virtualization solution, a virtualization management server may be used to configure and manage multiple hypervisors across the network. With some solutions, VMs can even migrate between host hypervisors, moving to different physical hosts while maintaining their virtual identity and services.

The IBM Networking OS 7.8 VMready feature supports up to 2048 VEs in a virtualized data center environment. The switch automatically discovers the VEs attached to switch ports, and distinguishes between regular VMs, Service Console Interfaces, and Kernel/Management Interfaces in a VMware® environment.

VEs may be placed into VM groups on the switch to define communication boundaries: VEs in the same VM group may communicate with each other, while VEs in different groups may not. VM groups also allow for configuring group-level settings such as virtualization policies and ACLs.

The administrator can also pre-provision VEs by adding their MAC addresses (or their IPv4 address or VM name in a VMware environment) to a VM group. When a VE with a pre-provisioned MAC address becomes connected to the switch, the switch will automatically apply the appropriate group membership configuration.

The VFSM with VMready also detects the migration of VEs across different hypervisors. As VEs move, the VFSM NMotion™ feature automatically moves the appropriate network configuration as well. NMotion gives the switch the ability to maintain assigned group membership and associated policies, even when a VE moves to a different port on the switch.

VMready also works with VMware Virtual Center (vCenter) management software. Connecting with a vCenter allows the VFSM to collect information about more distant VEs, synchronize switch and VE configuration, and extend migration properties.

**Note:** VM groups and policies, VE pre-provisioning, and VE migration features are not supported simultaneously on the same ports as vNICs (see [“Virtual NICs” on page 217](#)).

---

## VE Capacity

When VMready is enabled, the switch will automatically discover VEs that reside in hypervisors directly connected on the switch ports. IBM N/OS 7.8 supports up to 2048 VEs. Once this limit is reached, the switch will reject additional VEs.

**Note:** In rare situations, the switch may reject new VEs prior to reaching the supported limit. This can occur when the internal hash corresponding to the new VE is already in use. If this occurs, change the MAC address of the VE and retry the operation. The MAC address can usually be changed from the virtualization management server console (such as the VMware Virtual Center).

---

## VM Group Types

VEs, as well as internal ports, external ports, static trunks and LACP trunks, can be placed into VM groups on the switch to define virtual communication boundaries. Elements in a given VM group are permitted to communicate with each other, while those in different groups are not. The elements within a VM group automatically share certain group-level settings.

There are two different types of VM groups:

- Local VM groups are maintained locally on the switch. Their configuration is not synchronized with hypervisors.
- Distributed VM groups are automatically synchronized with a virtualization management server (see [“Assigning a vCenter” on page 249](#)).

Each VM group type is covered in detail in the following sections.

**Note:** VM groups are not supported simultaneously on the same ports as vNICs (see [“Virtual NICs” on page 217](#)).

---

## Local VM Groups

The configuration for local VM groups is maintained on the switch (locally) and is not directly synchronized with hypervisors. Local VM groups may include only local elements: local switch ports and trunks, and only those VEs connected to one of the switch ports or pre-provisioned on the switch.

Local VM groups support limited VE migration: as VMs and other VEs move to different hypervisors connected to different ports on the switch, the configuration of their group identity and features moves with them. However, VE migration to and from more distant hypervisors (those not connected to the VFSM, may require manual configuration when using local VM groups.

### Configuring a Local VM Group

Local VM groups are configured in the VM Group menu:

```
VFSM(config)# virt vmgroup <VM group number>
```

Within the VM Group menu, use the following commands to assign group properties and membership:

cpu	(Enable sending unregistered IPMC to CPU)
flood	(Enable flooding unregistered IPMC)
key <LACP trunk key>	(Add LACP trunk to group)
optflood	(Enable optimized flooding)
port <port alias or number>	(Add port member to group)
portchannel <trunk group number>	(Add static trunk to group)
profile <profile name>	(Not used for local groups)
stg <Spanning Tree group>	(Add STG to group)
tag	(Set VLAN tagging on ports)
validate <advanced basic>	(Validate mode for the group)
vlan <VLAN number>	(Specify the group VLAN)
vm <MAC> <index> <UUID> <IPv4 address> <name>	(Add VM member to group)
vmap <VMAP number> [intports extports]	(Specify VMAP number)
vport <Virtual port>	(Add a virtual port to the group)

The following rules apply to the local VM group configuration commands:

- `cpu`: Enable sending unregistered IPMC to CPU.
- `flood`: Enable flooding unregistered IPMC.
- `key`: Add LACP trunks to the group.
- `optflood`: Enable optimized flooding to allow sending unregistered IPMC to the Mrouter ports without having any packet loss during the learning period; This option is disabled by default; When `optflood` is enabled, the `flood` and `cpu` settings are ignored.
- `port`: Add switch server ports or switch uplink ports to the group. Note that VM groups and vNICs (see [“Virtual NICs” on page 239](#)) are not supported simultaneously on the same port.
- `portchannel`: Add static port trunks to the group.
- `profile`: The profile options are not applicable to local VM groups. Only distributed VM groups may use VM profiles (see [“VM Profiles” on page 242](#)).
- `stg`: The group may be assigned to a Spanning-Tree group for broadcast loop control (see [“Spanning Tree Protocols” on page 139](#)).
- `tag`: Enable VLAN tagging for the VM group. If the VM group contains ports which also exist in other VM groups, enable tagging in both VM groups.
- `validate`: Set validate mode for the group.
- `vlan`: Each VM group must have a unique VLAN number. This is required for local VM groups. If one is not explicitly configured, the switch will automatically assign the next unconfigured VLAN when a VE or port is added to the VM group.
- `vmap`: Each VM group may optionally be assigned a VLAN-based ACL (see [“VLAN Maps” on page 253](#)).
- `vm`: Add VMs.  
VMs and other VEs are primarily specified by MAC address. They can also be specified by UUID or by the index number as shown in various VMready information output (see [“VMready Information Displays” on page 256](#)).
- `vport`: Add a virtual port.  
Add a virtual port to the group.

**Note:** Local VM groups are not supported simultaneously on the same ports as vNICs (see [“Virtual NICs” on page 217](#)).

---

## Distributed VM Groups

Distributed VM groups allow configuration profiles to be synchronized between the VFSM and associated hypervisors and VEs. This allows VE configuration to be centralized, and provides for more reliable VE migration across hypervisors.

Using distributed VM groups requires a virtualization management server. The management server acts as a central point of access to configure and maintain multiple hypervisors and their VEs (VMs, virtual switches, and so on).

The VFSM must connect to a virtualization management server before distributed VM groups can be used. The switch uses this connection to collect configuration information about associated VEs, and can also automatically push configuration profiles to the virtualization management server, which in turn configures the hypervisors and VEs. See [“Virtualization Management Servers” on page 249](#) for more information.

**Note:** Distributed VM groups are not supported simultaneously on the same ports as vNICs (see [“Virtual NICs” on page 217](#)).

## VM Profiles

VM profiles are required for configuring distributed VM groups. They are not used with local VM groups. A VM profile defines the VLAN and virtual switch bandwidth shaping characteristics for the distributed VM group. The switch distributes these settings to the virtualization management server, which in turn distributes them to the appropriate hypervisors for VE members associated with the group.

Creating VM profiles is a two part process. First, the VM profile is created as shown in the following command on the switch:

```
VFSM(config)# virt vmpofile <profile name>
```

Next, the profile must be edited and configured using the following configuration commands:

```
VFSM(config)# virt vmpofile edit <profile name> ?
eshaping <average bandwidth> <burst size> <peak>
shaping <average bandwidth> <burst size> <peak>
vlan <VLAN number>
```

For virtual switch bandwidth shaping parameters, average and peak bandwidth are specified in kilobits per second (a value of 1000 represents 1 Mbps). Burst size is specified in kilobytes (a value of 1000 represents 1 MB).

**Note:** The bandwidth shaping parameters in the VM profile are used by the hypervisor virtual switch software. To set bandwidth policies for individual VEs, see [“VM Policy Bandwidth Control” on page 254](#).

Once configured, the VM profile may be assigned to a distributed VM group as shown in the following section.

## Initializing a Distributed VM Group

**Note:** A VM profile is required before a distributed VM group may be configured. See [“VM Profiles” on page 242](#) for details.

Once a VM profile is available, a distributed VM group may be initialized using the following configuration command:

```
VFSM(config)# virt vmgroup <VM group number> profile <VM profile name>
```

Only one VM profile can be assigned to a given distributed VM group. To change the VM profile, the old one must first be removed.

```
VFSM(config)# no virt vmgroup <VM group number> profile
```

**Note:** The VM profile can be added only to an empty VM group (one that has no VLAN, VMs, or port members). Any VM group number currently configured for a local VM group (see [“Local VM Groups” on page 240](#)) cannot be converted and must be deleted before it can be used for a distributed VM group.

## Assigning Members

VMs, ports, and trunks may be added to the distributed VM group only after the VM profile is assigned.

When you add a member to a distributed VM group, a port group is created on the virtual switch (or distributed virtual switch; See [“Virtual Distributed Switch” on page 247](#)) to which the member is connected.

Group members are added, pre-provisioned, or removed from distributed VM groups in the same manner as with local VM groups ([“Local VM Groups” on page 240](#)), with the following exceptions:

- VMs: VMs and other VEs are not required to be local. Any VE known by the virtualization management server can be part of a distributed VM group.
- The VM group `vlan` option cannot be used with distributed VM groups. For distributed VM groups, the VLAN is assigned in the VM profile.

## Synchronizing the Configuration

When the configuration for a distributed VM group is applied (using the CLI `apply` command), the switch updates the assigned virtualization management server. The management server then distributes changes to the appropriate hypervisors.

For VM membership changes, hypervisors modify their internal virtual switch port groups, adding or removing internal port memberships to enforce the boundaries defined by the distributed VM groups. Virtual switch port groups created in this fashion can be identified in the virtual management server by the name of the VM profile, formatted as follows:

```
IBM_<VM profile name>
```

Using the `VFSM(config)# virt vmgroup <x> vm` command to add a server host interface to a distributed VM group does not create a new port group on the virtual switch or move the host. Instead, because the host interface already has its own virtual switch port group on the hypervisor, the VM profile settings are applied to its existing port group.

**Note:** When applying the distributed VM group configuration, the virtualization management server and associated hypervisors must take appropriate actions. If a hypervisor is unable to make requested changes, an error message will be displayed on the switch. Be sure to evaluate all error message and take the appropriate actions to be sure the expected changes are properly applied.

## Removing Member VEs

Removing a VE from a distributed VM group on the switch will have the following effects on the hypervisor:

- The VE will be moved to the `IBM_Default` port group in VLAN 0 (zero).
- Traffic shaping will be disabled for the VE.
- All other properties will be reset to default values inherited from the virtual switch.

---

## VMcheck

The VFSM primarily identifies virtual machines by their MAC addresses. An untrusted server or a VM could identify itself by a trusted MAC address leading to MAC spoofing attacks. Sometimes, MAC addresses get transferred to another VM, or they get duplicated.

The VMcheck solution addresses these security concerns by validating the MAC addresses assigned to VMs. The switch periodically sends hello messages on server ports. These messages include the switch identifier and port number. The hypervisor listens to these messages on physical NICs and stores the information, which can be retrieved using the VMware Infrastructure Application Programming Interface (VI API). This information is used to validate VM MAC addresses. Two modes of validation are available: Basic and Advanced.

Use the following command to select the validation mode or to disable validation:

```
VFSM(config)# [no] virt vmgroup <VM group number> validate {basic|advanced}
```

### Basic Validation

This mode provides port-based validation by identifying the port used by a hypervisor. It is suitable for environments in which MAC reassignment or duplication cannot occur.

The switch, using the hello message information, identifies a hypervisor port. If the hypervisor port is found in the hello message information, it is deemed to be a trusted port. Basic validation should be enabled when:

- A VM is added to a VM group, and the MAC address of the VM interface is in the Layer 2 table of the switch.
- A VM interface that belongs to a VM group experiences a “source miss” i.e. is not able to learn new MAC address.
- A trusted port goes down. Port validation must be performed to ensure that the port does not get connected to an untrusted source when it comes back up.

Use the following command to set the action to be performed if the switch is unable to validate the VM MAC address:

```
VFSM(config)# virt vmcheck action basic {log|link}
```

```
log - generates a log  
link - disables the port
```

### Advanced Validation

This mode provides VM-based validation by mapping a switch port to a VM MAC address. It is suitable for environments in which spoofing, MAC reassignment, or MAC duplication is possible.

When the switch receives frames from a VM, it first validates the VM interface based on the VM MAC address, VM Universally Unique Identifier (UUID), Switch port, and Switch ID available in the hello message information. Only if all the four parameters are matched, the VM MAC address is considered valid.

In advanced validation mode, if the VM MAC address validation fails, an ACL can be created to drop the traffic received from the VM MAC address on the switch port. Use the following command to specify the number of ACLs to be used for dropping traffic:

```
VFSM(config)# virt vmcheck acls max <1-256>
```

Use the following command to set the action to be performed if the switch is unable to validate the VM MAC address:

```
VFSM(config)# virt vmcheck action advanced {log|link|acl}
```

Following are the other VMcheck commands:

**Table 22. VMcheck Commands**

Command	Description
VFSM(config)# virt vmware hello {enable  hport <port number> haddr htimer}	Hello messages setting: enable/add port/advertise this IP address in the hello messages instead of the default management IP address/set the timer to send the hello messages
VFSM(config)# no virt vmware hello {enable hport <port number>}	Disable hello messages/remove port
VFSM(config)# [no] virt vmcheck trust <port number or range>	Mark a port as trusted; Use the no form of the command to mark port as untrusted
VFSM# no virt vmcheck acl [mac-address [<port number>] port]	Delete ACL(s): all ACLs/an ACL by MAC address ((optional) and port number) /all ACLs installed on a port

---

## Virtual Distributed Switch

A virtual Distributed Switch (vDS ) allows the hypervisor's NIC to be attached to the vDS instead of its own virtual switch. The vDS connects to the vCenter and spans across multiple hypervisors in a datacenter. The administrator can manage virtual machine networking for the entire data center from a single interface. The vDS enables centralized provisioning and administration of virtual machine networking in the data center using the VMware vCenter server.

When a member is added to a distributed VM group, a distributed port group is created on the vDS. The member is then added to the distributed port group.

Distributed port groups on a vDS are available to all hypervisors that are connected to the vDS. Members of a single distributed port group can communicate with each other.

**Note:** vDS works with ESX 4.0 or higher versions.

To add a vDS, use the command:

```
VFSM# virt vmware dvswitch add <datacenter name> <dvSwitch name> [<dvSwitch-version>]
```

## Prerequisites

Before adding a vDS on the VFSM, ensure the following:

- VMware vCenter is fully installed and configured and includes a “bladevm” administration account and a valid SSL certificate.
- A virtual distributed switch instance has been created on the vCenter. The vDS version must be higher or the same as the hypervisor version on the hosts.
- At least two hypervisors are configured.

## Guidelines

Before migrating VMs to a vDS, consider the following:

- At any one time, a VM NIC can be associated with only one virtual switch: to the hypervisor's virtual switch, or to the vDS.
- Management connection to the server must be ensured during the migration. The connection is via the Service Console or the Kernel/Management Interface.
- The vDS configuration and migration can be viewed in vCenter at the following locations:
  - vDS: Home > Inventory > Networking
  - vDS Hosts: Home > Inventory > Networking > vDS > Hosts

**Note:** These changes will not be displayed in the running configuration on the VFSM.

## Migrating to vDS

You can migrate VMs to the vDS using vCenter. The migration may also be accomplished using the operational commands on the VFSM available in the following CLI menus:

### For VMware vDS operations:

```
VFSM# virt vmware dvswitch <operational command>
add          Add a dvSwitch to a DataCenter
addhost      Add a host to a dvSwitch
adduplnk     Add a physical NIC to dvSwitch uplink ports
del          Remove a dvSwitch from a DataCenter
remhost      Remove a host from a dvSwitch
remuplnk     Remove a physical NIC from dvSwitch uplink ports
```

### For VMware distributed port group operations:

```
VFSM# virt vmware dpg <operational command>
add          Add a port group to a dvSwitch
del          Delete a port group from a dvSwitch
update       Update a port group on a dvSwitch
vmac         Change a VM NIC's port group
```

---

## Virtualization Management Servers

The VFSM can connect with a virtualization management server to collect configuration information about associated VEs. The switch can also automatically push VM group configuration profiles to the virtualization management server, which in turn configures the hypervisors and VEs, providing enhanced VE mobility.

One virtual management server must be assigned on the switch before distributed VM groups may be used. N/OS 7.8 currently supports only the VMware Virtual Center (vCenter).

**Note:** Although VM groups and policies are not supported simultaneously on the same ports as vNICs ([“Virtual NICs” on page 217](#)), vCenter synchronization can provide additional information about VEs on vNIC and non-vNIC ports.

### Assigning a vCenter

Assigning a vCenter to the switch requires the following:

- The vCenter must have a valid IPv4 address which is accessible to the switch (IPv6 addressing is not supported for the vCenter).
- A user account must be configured on the vCenter to provide access for the switch. The account must have (at a minimum) the following vCenter user privileges:
  - Network
  - Host Network > Configuration
  - Virtual Machine > Modify Device Settings

Once vCenter requirements are met, the following configuration command can be used on the VFSM to associate the vCenter with the switch:

```
VFSM(config)# virt vmware vcspec <vCenter IPv4 address> <username> [noauth]
```

This command specifies the IPv4 address and account username that the switch will use for vCenter access. Once entered, the administrator will be prompted to enter the password for the specified vCenter account.

The `noauth` option causes the switch to ignore SSL certificate authentication. This is required when no authoritative SSL certificate is installed on the vCenter.

**Note:** By default, the vCenter includes only a self-signed SSL certificate. If using the default certificate, the `noauth` option is required.

Once the vCenter configuration has been applied on the switch, the VFSM will connect to the vCenter to collect VE information.

### vCenter Scans

Once the vCenter is assigned, the switch will periodically scan the vCenter to collect basic information about all the VEs in the datacenter, and more detailed information about the local VEs that the switch has discovered attached to its own ports.

The switch completes a vCenter scan approximately every two minutes. Any major changes made through the vCenter may take up to two minutes to be reflected on the switch. However, you can force an immediate scan of the vCenter by using one of the following commands:

VFSM# virt vmware scan	(Scan the vCenter)
-OR-	
VFSM# show virt vm -v -r	(Scan vCenter and display result)

## Deleting the vCenter

To detach the vCenter from the switch, use the following configuration command:

VFSM(config)# no virt vmware vcspec
-------------------------------------

**Note:** Without a valid vCenter assigned on the switch, any VE configuration changes must be manually synchronized.

Deleting the assigned vCenter prevents synchronizing the configuration between the VFSM and VEs. VEs already operating in distributed VM groups will continue to function as configured, but any changes made to any VM profile or distributed VM group on the switch will affect only switch operation; changes on the switch will not be reflected in the vCenter or on the VEs. Likewise, any changes made to VE configuration on the vCenter will no longer be reflected on the switch.

## Exporting Profiles

VM profiles for discovered VEs in distributed VM groups are automatically synchronized with the virtual management server and the appropriate hypervisors. However, VM profiles can also be manually exported to specific hosts before individual VEs are defined on them.

By exporting VM profiles to a specific host, IBM port groups will be available to the host's internal virtual switches so that new VMs may be configured to use them.

VM migration requires that the target hypervisor includes all the virtual switch port groups to which the VM connects on the source hypervisor. The VM profile export feature can be used to distribute the associated port groups to all the potential hosts for a given VM.

A VM profile can be exported to a host using the following command:

```
VFSM# virt vmware export <VM profile name> <host list> <virtual switch name>
```

The host list can include one or more target hosts, specified by host name, IPv4 address, or UUID, with each list item separated by a space. If the virtual switch name is omitted, the administrator will be prompted to select one from a list or to enter a new virtual switch name.

Once executed, the requisite port group will be created on the specified virtual switch. If the specified virtual switch does not exist on the target host, it will be created with default properties, but with no uplink connection to a physical NIC (the administrator must assign uplinks using VMware management tools).

## VMware Operational Commands

The VFSM may be used as a central point of configuration for VMware virtual switches and port groups using the VMware operational menu, available with the following CLI command:

```
VFSM# virt vmware ?
dpg          Distributed port group operations
dvswitch     VMWare dvSwitch operations
export       Create or update a vm profile on one host
pg           Add a port group to a host
scan         Perform a VM Agent scan operation now
updpg        Update a port group on a host
vmacpg       Change a vnic's port group
vsw          Add a vswitch to a host
```

---

## Pre-Provisioning VEs

VEs may be manually added to VM groups in advance of being detected on the switch ports. By pre-provisioning the MAC address of VEs that are not yet active, the switch will be able to later recognize the VE when it becomes active on a switch port, and immediately assign the proper VM group properties without further configuration.

Undiscovered VEs are added to or removed from VM groups using the following configuration commands:

```
VFSM(config)# [no] virt vmgroup <VM group number> vm <VE MAC address>
```

For the pre-provisioning of undiscovered VEs, a MAC address is required. Other identifying properties, such as IPv4 address or VM name permitted for known VEs, cannot be used for pre-provisioning.

**Note:** Because VM groups are isolated from vNIC groups (see [“vNIC Groups” on page 225](#)), pre-provisioned VEs that appear on vNIC ports will not be added to the specified VM group upon discovery.

---

## VLAN Maps

A VLAN map (VMAP) is a type of Access Control List (ACL) that is applied to a VLAN or VM group rather than to a switch port as with regular ACLs (see [“Access Control Lists” on page 103](#)). In a virtualized environment, VMAPs allow you to create traffic filtering and metering policies that are associated with a VM group VLAN, allowing filters to follow VMs as they migrate between hypervisors.

**Note:** VLAN maps for VM groups are not supported simultaneously on the same ports as vNICs (see [“Virtual NICs” on page 217](#)).

VMAPs are configured using the following ISCLI configuration command path:

```
VFSM(config)# access-control vmap <VMAP ID> ?
  action          Set filter action
  egress-port     Set to filter for packets egressing this port
  ethernet        Ethernet header options
  ipv4            IP version 4 header options
  meter           ACL metering configuration
  packet-format   Set to filter specific packet format types
  re-mark         ACL re-mark configuration
  statistics      Enable access control list statistics
  tcp-udp         TCP and UDP filtering options
```

N/OS 7.8 supports up to 128 VMAPs. Individual VMAP filters are configured in the same fashion as regular ACLs, except that VLANs cannot be specified as a filtering criteria (unnecessary, since VMAPs are assigned to a specific VLAN or associated with a VM group VLAN).

Once a VMAP filter is created, it can be assigned or removed using the following commands:

- For regular VLANs, use config-vlan mode:

```
VFSM(config)# vlan <VLAN ID>
VFSM(config-vlan)# [no] vmap <VMAP ID> [intports| extports]
```

- For a VM group, use the global configuration mode:

```
VFSM(config)# [no] virt vmgroup <ID> vmap <VMAP ID> [intports|extports]
```

**Note:** Each VMAP can be assigned to only one VLAN or VM group. However, each VLAN or VM group may have multiple VMAPs assigned to it.

The optional `intports` or `extports` parameter can be specified to apply the action (to add or remove the VMAP) for either the internal ports or external ports only. If omitted, the operation will be applied to all ports in the associated VLAN or VM group.

**Note:** VMAPs have a lower priority than port-based ACLs. If both an ACL and a VMAP match a particular packet, both filter actions will be applied as long as there is no conflict. In the event of a conflict, the port ACL will take priority, though switch statistics will count matches for both the ACL and VMAP.

---

## VM Policy Bandwidth Control

In a virtualized environment where VEs can migrate between hypervisors and thus move among different ports on the switch, traffic bandwidth policies must be attached to VEs, rather than to a specific switch port.

VM Policy Bandwidth Control allows the administrator to specify the amount of data the switch will permit to flow to or from a particular VE, without defining a complicated matrix of ACLs or VMAPs for all port combinations where a VE may appear.

## VM Policy Bandwidth Control Commands

VM Policy Bandwidth Control can be configured using the following configuration commands:

```
VFSM(config)# virt vmpolicy vmbwidth <VM MAC>|<index>|<UUID>| <IPv4 address>|<name> ?
txrate <committed rate> <burst> [<ACL number>] (Set the VM to switch rate)
rxrate <committed rate> <burst> [<ACL number>] (Set the switch to VM rate)
bwctrl (Enable bandwidth control)
```

Bandwidth allocation can be defined either for transmit (TX) traffic or receive (RX) traffic. Because bandwidth allocation is specified from the perspective of the VE, the switch command for TX Rate Control (`txrate`) sets the data rate to be sent from the VM to the switch, and the RX Rate Control (`rxrate`) sets the data rate to be received by the VM from the switch.

The *committed rate* is specified in multiples of 64 kbps, from 64 to 10,000,000. The maximum *burst* rate is specified as 32, 64, 128, 256, 1024, 2048, or 4096 kb. If both the committed rate and burst are set to 0, bandwidth control in that direction (TX or RX) will be disabled.

When `txrate` is specified, the switch automatically selects an available ACL for internal use with bandwidth control. Optionally, if automatic ACL selection is not desired, a specific ACL may be selected. If there are no unassigned ACLs available, `txrate` cannot be configured.

## Bandwidth Policies vs. Bandwidth Shaping

VM Profile Bandwidth Shaping differs from VM Policy Bandwidth Control.

VM Profile Bandwidth Shaping (see [“VM Profiles” on page 242](#)) is configured per VM group and is enforced on the server by a virtual switch in the hypervisor. Shaping is unidirectional and limits traffic transmitted from the virtual switch to the VFSM. Shaping is performed prior to transmit VM Policy Bandwidth Control. If the egress traffic for a virtual switch port group exceeds shaping parameters, the traffic is dropped by the virtual switch in the hypervisor. Shaping uses server CPU resources, but prevents extra traffic from consuming bandwidth between the server and the VFSM. Shaping is not supported simultaneously on the same ports as vNICs.

VM Policy Bandwidth Control is configured per VE, and can be set independently for transmit and receive traffic. Bandwidth policies are enforced by the VFSM. VE traffic that exceeds configured levels is dropped by the switch upon ingress (for `txrate`) or before egress (for `rxrate`). Setting `txrate` uses ACL resources on the switch.

Bandwidth shaping and bandwidth policies can be used separately or in concert.

---

## VMready Information Displays

The VFSM can be used to display a variety of VMready information.

**Note:** Some displays depict information collected from scans of a VMware vCenter and may not be available without a valid vCenter. If a vCenter is assigned (see [“Assigning a vCenter” on page 249](#)), scan information might not be available for up to two minutes after the switch boots or when VMready is first enabled. Also, any major changes made through the vCenter may take up to two minutes to be reflected on the switch unless you force an immediate vCenter scan (see [“vCenter Scans” on page 249](#)).

### Local VE Information

A concise list of local VEs and pre-provisioned VEs is available with the following command:

```
VFSM# show virt vm
```

IP Address	VMAC Address	Index	Port	VM Group (Profile)
*172.16.46.50	00:50:56:4e:62:00	4	3	
*172.16.46.10	00:50:56:4f:f2:00	2	4	
+172.16.46.51	00:50:56:72:ec:00	1	3	
+172.16.46.11	00:50:56:7c:1c:00	3	4	
172.16.46.25	00:50:56:9c:00:00	5	4	
172.16.46.15	00:50:56:9c:21:00	0	4	
172.16.46.35	00:50:56:9c:29:00	6	3	
172.16.46.45	00:50:56:9c:47:00	7	3	

Number of entries: 8  
\* indicates VMware ESX Service Console Interface  
+ indicates VMware ESX/ESXi VMKernel or Management Interface

**Note:** The Index numbers shown in the VE information displays can be used to specify a particular VE in configuration commands.

If a vCenter is available, more verbose information can be obtained using the following command option:

```
VFSM# show virt vm -v
```

Index	MAC Address, IP Address	Name (VM or Host), @Host (VMs only)	Port, VLAN	Group	Vswitch, Port Group
0	00:50:56:9c:21:2f 172.16.46.15	atom @172.16.46.10	INT4 500		vSwitch0 Eng_A
+1	00:50:56:72:ec:86 172.16.46.51	172.16.46.50	INT3 0		vSwitch0 VMkernel
*2	00:50:56:4f:f2:85 172.16.46.10	172.16.46.10	INT4 0		vSwitch0 Mgmt
+3	00:50:56:7c:1c:ca 172.16.46.11	172.16.46.10	INT4 0		vSwitch0 VMkernel
*4	00:50:56:4e:62:f5 172.16.46.50	172.16.46.50	INT3 0		vSwitch0 Mgmt
5	00:50:56:9c:00:c8 172.16.46.25	quark @172.16.46.10	INT4 0		vSwitch0 Corp
6	00:50:56:9c:29:29 172.16.46.35	particle @172.16.46.50	INT3 0		vSwitch0 VM Network
7	00:50:56:9c:47:fd 172.16.46.45	nucleus @172.16.46.50	INT3 0		vSwitch0 Finance
--					

12 of 12 entries printed  
 \* indicates VMware ESX Service Console Interface  
 + indicates VMware ESX/ESXi VMkernel or Management Interface

To view additional detail regarding any specific VE, see [“vCenter VE Details” on page 259](#)).

## vCenter Hypervisor Hosts

If a vCenter is available, the following command displays the name and UUID of all VMware hosts, providing an essential overview of the data center:

```
VFSM# show virt vmware hosts
```

UUID	Name(s), IP Address
00a42681-d0e5-5910-a0bf-bd23bd3f7800	172.16.41.30
002e063c-153c-dd11-8b32-a78dd1909a00	172.16.46.10
00f1fe30-143c-dd11-84f2-a8ba2cd7ae00	172.16.44.50
0018938e-143c-dd11-9f7a-d8defa4b8300	172.16.46.20
...	

Using the following command, the administrator can view more detailed vCenter host information, including a list of virtual switches and their port groups, as well as details for all associated VEs:

```
VFSM# show virt vmware showhost {<UUID>|<IPv4 address>|<host name>}
Vswitches available on the host:
    vSwitch0
Port Groups and their Vswitches on the host:
    IBM_Default          vSwitch0
    VM Network           vSwitch0
    Service Console      vSwitch0
    VMkernel              vSwitch0
-----
MAC Address      00:50:56:9c:21:2f
Port             INT4
Type             Virtual Machine
VM vCenter Name  halibut
VM OS hostname   localhost.localdomain
VM IP Address    172.16.46.15
VM UUID          001c41f3-ccd8-94bb-1b94-6b94b03b9200
Current VM Host  172.16.46.10
Vswitch          vSwitch0
Port Group       IBM_Default
VLAN ID          0
...
```

## vCenter VEs

If a vCenter is available, the following command displays a list of all known VEs:

```
VFSM# show virt vmware vms
UUID                                     Name(s), IP Address
-----
001cdf1d-863a-fa5e-58c0-d197ed3e3300   30vm1
001c1fba-5483-863f-de04-4953b5caa700   VM90
001c0441-c9ed-184c-7030-d6a6bc9b4d00   VM91
001cc06e-393b-a36b-2da9-c71098d9a700   vm_new
001c6384-f764-983c-83e3-e94fc78f2c00   sturgeon
001c7434-6bf9-52bd-c48c-a410da0c2300   VM70
001cad78-8a3c-9cbe-35f6-59ca5f392500   VM60
001cf762-a577-f42a-c6ea-090216c11800   30VM6
001c41f3-ccd8-94bb-1b94-6b94b03b9200   halibut, localhost.localdomain,
                                         172.16.46.15
001cf17b-5581-ea80-c22c-3236b89ee900   30vm5
001c4312-a145-bf44-7edd-49b7a2fc3800   vm3
001caf40-a40a-de6f-7b44-9c496f123b00   30VM7
```

## vCenter VE Details

If a vCenter is available, the following command displays detailed information about a specific VE:

```
VFSM# show virt vmware showvm {<VM UUID>|<VM IPv4 address>|<VM name>}
```

```
-----  
MAC Address      00:50:56:9c:21:2f  
Port             INT4  
Type             Virtual Machine  
VM vCenter Name  halibut  
VM OS hostname   localhost.localdomain  
VM IP Address    172.16.46.15  
VM UUID          001c41f3-ccd8-94bb-1b94-6b94b03b9200  
Current VM Host  172.16.46.10  
Vswitch          vSwitch0  
Port Group       IBM_Default  
VLAN ID          0
```

---

## VMready Configuration Example

This example has the following characteristics:

- A VMware vCenter is fully installed and configured prior to VMready configuration and includes a “bladevm” administration account and a valid SSL certificate.
- The distributed VM group model is used.
- The VM profile named “Finance” is configured for VLAN 30, and specifies NIC-to-switch bandwidth shaping for 1Mbps average bandwidth, 2MB bursts, and 3Mbps maximum bandwidth.
- The VM group includes four discovered VMs on internal switch ports INT1 and INT2, and one static trunk (previously configured) that includes external ports EXT3 and EXT4.

1. Enable the VMready feature.

```
VFSM(config)# virt enable
```

2. Specify the VMware vCenter IPv4 address.

```
VFSM(config)# virt vmware vmware vcspec 172.16.100.1 bladevm
```

When prompted, enter the user password that the switch must use for access to the vCenter.

3. Create the VM profile.

```
VFSM(config)# virt vmprofile Finance
VFSM(config)# virt vmprofile edit Finance vlan 30
VFSM(config)# virt vmprofile edit Finance shaping 1000 2000 3000
```

4. Define the VM group.

```
VFSM(config)# virt vmgroup 1 profile Finance
VFSM(config)# virt vmgroup 1 vm arctic
VFSM(config)# virt vmgroup 1 vm monster
VFSM(config)# virt vmgroup 1 vm sierra
VFSM(config)# virt vmgroup 1 vm 00:50:56:4f:f2:00
VFSM(config)# virt vmgroup 1 portchannel 1
```

When VMs are added, the internal server ports on which they appear are automatically added to the VM group. In this example, there is no need to manually add ports INT1 and INT2.

**Note:** VM groups and vNICs (see [“Virtual NICs” on page 217](#)) are not supported simultaneously on the same switch ports.

5. If necessary, enable VLAN tagging/trunk mode for the VM group:

```
VFSM(config)# virt vmgroup 1 tag
```

**Note:** If the VM group contains ports which also exist in other VM groups, tagging/trunk mode should be enabled in both VM groups. In this example configuration, no ports exist in more than VM group.

---

## Chapter 15. FCoE and CEE

This chapter provides conceptual background and configuration examples for using Converged Enhanced Ethernet (CEE) features of the Virtual Fabric Switch Module, with an emphasis on Fibre Channel over Ethernet (FCoE) solutions. The following topics are addressed in this chapter:

- [“Fibre Channel over Ethernet” on page 262](#)

Fibre Channel over Ethernet (FCoE) allows Fibre Channel traffic to be transported over Ethernet links. This provides an evolutionary approach toward network consolidation, allowing Fibre Channel equipment and tools to be retained, while leveraging cheap, ubiquitous Ethernet networks for growth.

- [“FCoE Initialization Protocol Snooping” on page 271](#)

Using FCoE Initialization Protocol (FIP) snooping, the VFSM examines the FIP frames exchanged between ENodes and FCFs. This information is used to dynamically determine the ACLs required to block certain types of undesired or unvalidated traffic on FCoE links.

- [“Converged Enhanced Ethernet” on page 268](#)

Converged Enhanced Ethernet (CEE) refers to a set of IEEE standards developed primarily to enable FCoE, requiring enhancing the existing Ethernet standards to make them lossless on a per-priority traffic basis, and providing a mechanism to carry converged (LAN/SAN/IPC) traffic on a single physical link. CEE features can also be utilized in traditional LAN (non-FCoE) networks to provide lossless guarantees on a per-priority basis, and to provide efficient bandwidth allocation.

- [“Priority-Based Flow Control” on page 277](#)

Priority-Based Flow Control (PFC) extends 802.3x standard flow control to allow the switch to pause traffic based on the 802.1p priority value in each packet’s VLAN tag. PFC is vital for FCoE environments, where SAN traffic must remain lossless and should be paused during congestion, while LAN traffic on the same links should be delivered with “best effort” characteristics.

- [“Enhanced Transmission Selection” on page 281](#)

Enhanced Transmission Selection (ETS) provides a method for allocating link bandwidth based on the 802.1p priority value in each packet’s VLAN tag. Using ETS, different types of traffic (such as LAN, SAN, and management) that are sensitive to different handling criteria can be configured either for specific bandwidth characteristics, low-latency, or best-effort transmission, despite sharing converged links as in an FCoE environment.

- [“Data Center Bridging Capability Exchange” on page 287](#)

Data Center Bridging Capability Exchange Protocol (DCBX) allows neighboring network devices to exchange information about their capabilities. This is used between CEE-capable devices for the purpose of discovering their peers, negotiating peer configurations, and detecting misconfigurations.

---

## Fibre Channel over Ethernet

Fibre Channel over Ethernet (FCoE) is an effort to converge two of the different physical networks in today's data centers. It allows Fibre Channel traffic (such as that commonly used in Storage Area Networks, or SANs) to be transported without loss over 10Gb Ethernet links (typically used for high-speed Local Area Networks, or LANs). This provides an evolutionary approach toward network consolidation, allowing Fibre Channel equipment and tools to be retained, while leveraging cheap, ubiquitous Ethernet networks for growth.

With server virtualization, servers capable of hosting both Fibre Channel and Ethernet applications will provide advantages in server efficiency, particularly as FCoE-enabled network adapters provide consolidated SAN and LAN traffic capabilities.

The IBM Virtual Fabric 10Gb Switch Module with IBM Networking OS 7.8 software is compliant with the INCITS T11.3, FC-BB-5 FCoE specification, supporting up to 2048 FCoE connections.

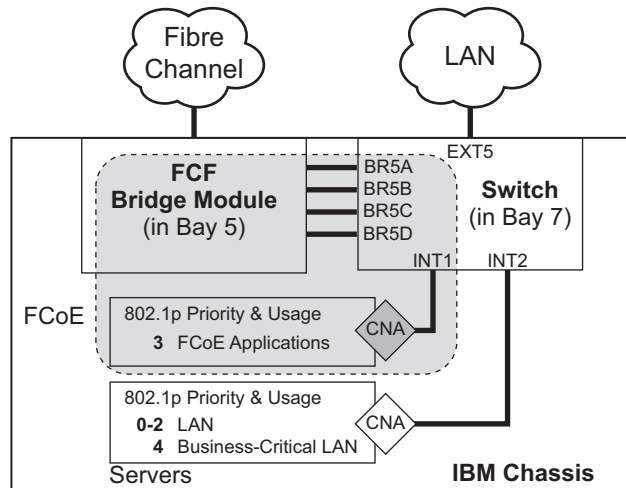
FCoE is supported in stand-alone and stacking mode.

### The FCoE Topology

In an end-to-end Fibre Channel network, switches and end devices generally establish trusted, point-to-point links. Fibre Channel switches validate end devices, enforce zoning configurations and device addressing, and prevent certain types of errors and attacks on the network.

In a converged FCoE network where Fibre Channel devices are bridged to Ethernet devices, although the direct point-to-point assurances normally provided by the Fibre Channel fabric may be lost in the transition between the different network types, the VFSM provides a solution.

Figure 28. A Mixed Fibre Channel and FCoE Network



In [Figure 28 on page 262](#), the Fibre Channel network is connected to the FCoE network through an FCoE Forwarder (FCF) bridge module in the blade chassis. The FCF acts as a Fibre Channel gateway to and from the FCoE network.

For the FCoE portion of the network, the FCF is connected to the FCoE-enabled VFSM, which is internally connected to a blade server (running Fibre Channel applications) through an FCoE-enabled Converged Network Adapter (CNA) known in Fibre Channel as Ethernet Nodes (ENodes).

**Note:** The figure also shows a non-FCoE LAN server connected to the VFSM using a CNA. This allows the LAN server to take advantage of some CEE features that are useful even outside of an FCoE environment.

In order to block undesired or unvalidated traffic on FCoE links that exists outside the regular Fibre Channel topology, Ethernet ports used in FCoE are configured with Access Control Lists (ACLs) that are narrowly tailored to permit expected FCoE traffic to and from confirmed FCFs and ENodes, and deny all other FCoE or FIP traffic. This ensures that all FCoE traffic to and from the ENode passes through the FCF.

Because manual ACL configuration is an administratively complex task, the VFSM can automatically and dynamically configure the ACLs required for use with FCoE. Using FCoE Initialization Protocol (FIP) snooping (see [“FCoE Initialization Protocol Snooping” on page 271](#)), the VFSM examines the FIP frames normally exchanged between the FCF and ENodes to determine information about connected FCoE devices. This information is used to automatically determine the appropriate ACLs required to block certain types of undesired or unvalidated FCoE traffic.

Automatic FCoE-related ACLs are independent from ACLs used for typical Ethernet purposes.

## FCoE Requirements

The following are required for implementing FCoE using the IBM Virtual Fabric 10Gb Switch Module (VFSM) with N/OS 7.8 software:

- The VFSM must be connected to the Fibre Channel network through an FCF such as a QLogic Virtual Fabric Extension Module for IBM BladeCenter, or a Cisco Nexus 5000 Series Switch.
- If the FCF is a blade chassis bridge module (such as the supported QLogic FCF), the connection between the VFSM and the bridge module requires either two or four VFSM external ports.
- For each VFSM internal port participating in FCoE, the connected blade server must use the supported FCoE CNA. The QLogic CNA is currently the first CNA supported for this purpose. Also supported is the Emulex Virtual Fabric Adapter, which includes vNIC support (with some additional topology rules).
- CEE must be turned on (see [“Turning CEE On or Off” on page 268](#)). When CEE is on, the DCBX, PFC, and ETS features are enabled and configured with default FCoE settings. These features may be reconfigured, but must remain enabled in order for FCoE to function.
- FIP snooping must be turned on (see [“FCoE Initialization Protocol Snooping” on page 271](#)). When FIP snooping is turned on, the feature is enabled on all ports by default. The administrator can disable FIP snooping on individual ports that do not require FCoE, but FIP snooping must remain enabled on all FCoE ports in order for FCoE to function.

## Port Trunking

IBM N/OS 7.8 supports port trunking for FCoE connections. The Link Aggregation (LAG) can be used for separate FCoE traffic, or for Ethernet and FCoE traffic. Ports directly connected to servers cannot be combined in a LAG group.

Uplink ports, connected to the FCF, can be grouped as static or dynamic trunks.

Internal ports cannot be grouped as trunks.

Normal trunk operations such as creating/enabling the trunk, and adding/removing member ports can be performed. When a port is added to a trunk group, FCFs previously detected on the port will be deleted. The deleted FCF may be relearned later. However, this may cause flickering in the network traffic.

Enhanced Transmission Selection (ETS), Priority-based Flow Control (PFC), and Data Center Bridging (DCBX) are configured on a per-port basis. Each port in a trunk must have the same ETS, PFC, and DCBX configuration. When a port ceases to be the trunk group member, its configuration does not change.

## Connecting to a Bridge Module

For FCoE, the VFSM may be connected to the Fibre Channel network via an FCF bridge module in the BladeCenter chassis. The QLogic Virtual Fabric Extension Module for IBM BladeCenter is the first FCF currently supported.

**Note:** LAG is not supported on the bridge module.

## Bridge Module Location

The FCF must be placed in a specific bridge module bay in the BladeCenter chassis, depending on where the VFSM is located:

- If the switch is in HSSM Bay 7, the bridge module must be in BM Bay 5.

- If the switch is in HSSM Bay 9, one or two bridge modules may be used in BM Bay 3 and/or BM Bay 5.

## Defining the Bridge Connection

The bridge connection can be configured for either 20Gbps or 40Gbps maximum throughput if one bridge module is used, or 20Gbps for each bridge module if two bridge modules are used.

To configure a bridge module connection on the VFSM, use the following ISCLI commands:

```
VFSM(config)# [no] boot bridge-module <bay_number> ?  
  bandwidth  Select bandwidth to the bridge module  
  enable     Enable connection to bridge module
```

where the bridge module number chassis bay number where the bridge module resides, and the bandwidth allocation represents 20 or 40 Gbps. For an example, see [“FIP Snooping Configuration” on page 275](#).

**Note:** After defining the bridge connection, the switch must be reset in order for the connection to be established.

Following are guidelines for configuring the bridge connection bandwidth:

- If VFSM is in Bay 7 and bridge module is in Bay 5, set bandwidth on bridge module to 40 Gbps.
- If VFSM is in Bay 9 and bridge module is in Bay 3, set bandwidth on bridge module to 40 Gbps.
- If VFSM is in Bay 7 and Bay 9, and bridge module is in Bay 5, set bandwidth on both the switches to 20 Gbps.

The VFSM-to-Bridge connection is made through the blade chassis using multiple internal KR ports (no external cabling is required). However, KR ports require the resources of 10Gb external switch ports in order to operate, as follows:

- 20Gbps connection requires 2 external ports
- 40Gbps connection requires 4 external ports

As each internal KR port is enabled, one corresponding external switch port is disabled. The external switch ports used in the bridge module connection are selected automatically, based on the bridge module location and the number of ports required. The exact ports selected for bridge connection are displayed with the `/info/link` command. Aliases for switch ports used for the bridge connection are renamed to BRxA through BRxD (depending on the number of ports required), with *x* representing the bridge module number.

Table 23 shows which external ports are disabled, depending on bridge module throughput.

Table 23. BCH chassis—Ethernet to KR port mapping

Throughput		VFSM SFP+ Ports									
BM 5	BM 3	EXT1	EXT2	EXT3	EXT4	EXT5	EXT6	EXT7	EXT8	EXT9	EXT10
0Gb	0Gb	SFP	SFP	SFP	SFP	SFP	SFP	SFP	SFP	SFP	SFP
20Gb	0Gb	KR	KR	SFP	SFP	SFP	SFP	SFP	SFP	SFP	SFP
40Gb	0Gb	KR	KR	KR	KR	SFP	SFP	SFP	SFP	SFP	SFP
0Gb	20Gb	SFP	SFP	SFP	SFP	SFP	SFP	SFP	SFP	KR	KR
0Gb	40Gb	SFP	SFP	SFP	SFP	SFP	SFP	KR	KR	KR	KR
20Gb	20Gb	KR	KR	SFP	SFP	SFP	SFP	SFP	SFP	KR	KR
SFP = External SFP+ is operational KR = Internal bridge port disables external SFP+											

---

## Converged Enhanced Ethernet

Converged Enhanced Ethernet (CEE) refers to a set of IEEE standards designed to allow different physical networks with different data handling requirements to be converged together, simplifying management, increasing efficiency and utilization, and leveraging legacy investments without sacrificing evolutionary growth.

CEE standards were developed primarily to enable Fibre Channel traffic to be carried over Ethernet networks. This required enhancing the existing Ethernet standards to make them lossless on a per-priority traffic basis, and to provide a mechanism to carry converged (LAN/SAN/IPC) traffic on a single physical link. Although CEE standards were designed with FCoE in mind, they are not limited to FCoE installations. CEE features can be utilized in traditional LAN (non-FCoE) networks to provide lossless guarantees on a per-priority basis, and to provide efficient bandwidth allocation based on application needs.

CEE is supported in stand-alone and stacking mode.

### Turning CEE On or Off

By default on the VFSM, CEE is turned off. To turn CEE on or off, use the following CLI commands:

```
VFSM(config)# [no] cee enable
```

For an example, see [“FIP Snooping Configuration” on page 275](#).



#### **CAUTION:**

**Turning CEE on and applying the configuration will automatically change some 802.1p QoS and 802.3x standard flow control settings on the VFSM. Read the following material carefully to determine whether you will need to take action to reconfigure expected settings.**

**It is recommended that you backup your configuration prior to turning CEE on. Viewing the file will allow you to manually re-create the equivalent configuration once CEE is turned on, and will also allow you to recover your prior configuration if you need to turn CEE off.**

### Effects on Link Layer Discovery Protocol

When CEE is turned on, Link Layer Discovery Protocol (LLDP) is automatically turned on and enabled for receiving and transmitting DCBX information. LLDP cannot be turned off while CEE is turned on.

## Effects on 802.1p Quality of Service

While CEE is off (the default), the VFSM allows 802.1p priority values to be used for Quality of Service (QoS) configuration (see [page 179](#)). 802.1p QoS default settings are shown in [Table 24](#), but can be changed by the administrator.

When CEE is turned on, 802.1p QoS is replaced by ETS (see [“Enhanced Transmission Selection” on page 281](#)). As a result, while CEE is turned on, the 802.1p QoS configuration commands are no longer available on the switch (the menu is restored when CEE is turned off).

In addition, when CEE is turned on, prior 802.1p QoS settings are replaced with new defaults designed for use with ETS priority groups (PGIDs) as shown in [Table 24](#):

*Table 24. CEE Effects on 802.1p Defaults*

802.1p QoS Configuration With CEE Off (default)			ETS Configuration With CEE On		
Priority	COSq	Weight	Priority	COSq	PGID
0	0	1	0	0	0
1	1	2	1	0	0
2	2	3	2	0	0
3	3	4	3	1	1
4	4	5	4	2	2
5	5	7	5	2	2
6	6	15	6	2	2
7	7	0	7	2	2

When CEE is on, the default ETS configuration also allocates a portion of link bandwidth to each PGID as shown in [Table 25](#):

*Table 25. Default ETS Bandwidth Allocation*

PGID	Typical Use	Bandwidth
0	LAN	10%
1	SAN	50%
2	Latency-sensitive LAN	40%

If the prior, non-CEE configuration used 802.1p priority values for different purposes, or does not expect bandwidth allocation as shown in [Table 25 on page 269](#), when CEE is turned on, the administrator should reconfigure ETS settings as appropriate.

## Effects on Flow Control

When CEE is off (the default), 802.3x standard flow control is enabled on all switch ports by default.

When CEE is turned on, standard flow control is disabled on all ports, and in its place, PFC (see [“Priority-Based Flow Control” on page 277](#)) is enabled on all ports for 802.1p priority value 3. This default is chosen because priority value 3 is commonly used to identify FCoE traffic in a CEE environment and must be guaranteed lossless behavior. PFC is disabled for all other priority values.

Each time CEE is turned off, the prior 802.3x standard flow control settings will be restored (including any previous changes from the defaults).

When CEE is on, PFC can be enabled only on priority value 3 and one other priority. If flow control is required on additional priorities on any given port, consider using standard flow control on that port, so that regardless of which priority traffic becomes congested, a flow control frame is generated.

---

## FCoE Initialization Protocol Snooping

FCoE Initialization Protocol (FIP) snooping is an FCoE feature. In order to enforce point-to-point links for FCoE traffic outside the regular Fibre Channel topology, Ethernet ports used in FCoE can be automatically and dynamically configured with Access Control Lists (ACLs).

Using FIP snooping, the VFSM examines the FIP frames normally exchanged between the FCF and ENodes to determine information about connected FCoE devices. This information is used to create narrowly tailored ACLs that permit expected FCoE traffic to and from confirmed Fibre Channel nodes, and deny all other undesirable FCoE or FIP traffic.

In case of trunk groups, FIP traffic from a particular FCF can be received by any member port on which the FCF was detected.

### Global FIP Snooping Settings

By default, the FIP snooping feature is turned off for the VFSM. The following commands are used to turn the feature on or off:

```
VFSM(config)# [no] fcoe fips enable
```

**Note:** FIP snooping requires CEE to be turned on (see [“Turning CEE On or Off” on page 268](#)).

When FIP snooping is on, port participation may be configured on a port-by-port basis (see below).

When FIP snooping is off, all FCoE-related ACLs generated by the feature are removed from all switch ports.

FIP snooping configuration must be the same on all member ports in a trunk group. If the configuration of a member port is changed, an error message, similar to the following, will be displayed.

```
"NOTE: The port x belongs to trunk x. The FIPS settings will be applied to all ports of the trunk"
```

The configuration changes are applied to all member ports.

For an example, see [“FIP Snooping Configuration” on page 275](#).

### FIP Snooping for Specific Ports

When FIP snooping is globally turned on (see above), ports may be individually configured for participation in FIP snooping and automatic ACL generation. By default, FIP snooping is enabled for each port. To change the setting for any specific port, use the following CLI commands:

```
VFSM(config)# [no] fcoe fips port <port number, alias, list, or range> enable
```

When FIP snooping is enabled on a port, FCoE-related ACLs will be automatically configured.

When FIP snooping is disabled on a port, all FCoE-related ACLs on the port are removed, and the switch will enforce no FCoE-related rules for traffic on the port.

**Note:** FIP Snooping and IPv6 ACLs (see [“Access Control Lists” on page 103](#)) are not support simultaneously on the same ports. To use FIP snooping, remove IPv6 ACLs from the port.

## Port FCF and ENode Detection

When FIP snooping is enabled on a port, the port is placed in FCF auto-detect mode by default. In this mode, the port assumes connection to an ENode unless FIP packets show the port is connected to an FCF.

Ports can also be specifically configured as to whether automatic FCF detection should be used, or whether the port is connected to an FCF or ENode:

```
VFSM(config)# fcoe fips port <port alias, number, list, or range> fcf-mode {auto|on|off}
```

When FCF mode is `on`, the port is assumed to be connected to a trusted FCF, and only ACLs appropriate to FCFs will be installed on the port. When `off`, the port is assumed to be connected to an ENode, and only ACLs appropriate to ENodes will be installed. When the mode is changed (either through manual configuration or as a result of automatic detection), the appropriate ACLs are automatically added, removed, or changed to reflect the new FCF or ENode connection.

## FCoE Connection Timeout

FCoE-related ACLs and VLANs are added, changed, and removed as FCoE device connection and disconnection are discovered. In addition, the administrator can enable or disable automatic removal of ACLs for FCFs and other FCoE connections that timeout (fail or are disconnected) without FIP notification.

By default, automatic removal of ACLs upon timeout is enabled. To change this function, use the following CLI command:

```
VFSM(config)# [no] fcoe fips timeout-acl
```

## FCoE ACL Rules

When FIP Snooping is enabled on a port, the switch automatically installs the appropriate ACLs to enforce the following rules for FCoE traffic:

- Ensure that FIP frames from ENodes may only be addressed to FCFs.
- Flag important FIP packets for switch processing.
- Ensure no end device uses an FCF MAC address as its source.
- Each FCoE port is assumed to be connected to an ENode and include ENode-specific ACLs installed, until the port is either detected or configured to be connected to an FCF.
- Ports that are configured to have FIP snooping disabled will not have any FIP or FCoE related ACLs installed.
- Prevent transmission of all FCoE frames from an ENode prior to its successful completion of login (FLOGI) to the FCF.
- After successful completion of FLOGI, ensure that the ENode uses only those FCoE source addresses assigned to it by FCF.
- After successful completion of FLOGI, ensure that all ENode FCoE source addresses originate from or are destined to the appropriate ENode port.
- After successful completion of each FLOGI, ensure that FCoE frames may only be addressed to the FCFs that accept them.

Initially, a basic set of FCoE-related ACLs will be installed on all ports where FIP snooping is enabled. As the switch encounters FIP frames and learns about FCFs and ENodes that are attached or disconnect, ACLs are dynamically installed or expanded to provide appropriate security.

When an FCoE connection logs out, or times out (if ACL timeout is enabled), the related ACLs will be automatically removed.

FCoE-related ACLs are independent of manually configured ACLs used for regular Ethernet purposes (see [“Access Control Lists” on page 103](#)). FCoE ACLs generally have a higher priority over standard ACLs, and do not inhibit non-FCoE and non-FIP traffic.

## FCoE VLANs

Switch FCF and ENode ports on the switch must belong to the same FCoE VLAN, and ENode ports must have VLAN tagging/trunk mode enabled. The FCoE VLAN can be either be manually configured on the switch with all member FCF and ENode ports, and must be supported by the participating FCF CNAs or automatically created if the automatic VLAN creation is enabled.

The automatic FCoE VLAN creation is turned on by default. If you need to disable it, use the following command:

```
VFSM(config)# no fcoe fips automatic-vlan
```

## Viewing FIP Snooping Information

ACLs automatically generated under FIP snooping are independent of regular, manually configure ACLs, and are not listed with regular ACLs in switch information and statistics output. Instead, FCoE ACLs are shown using the following CLI commands:

```
VFSM# show fcoe fips information (Show all FIP-related information)
VFSM# show fcoe fips port <ports> information (Show FIP info for a selected port)
```

For example:

```
VFSM# show fcoe fips port ext4 information

FIP Snooping on port 2:
This port has been detected to be an FCF port.

FIPS ACLs configured on this port:
Ethertype 0x8914, action permit.
dmac 00:00:18:01:00:XX, Ethertype 0x8914, action permit.
```

For each ACL, the required traffic criteria are listed, along with the action taken (permit or deny) for matching traffic. ACLs are listed in order of precedence and evaluated in the order shown.

The administrator can also view other FCoE information:

```
VFSM# show fcoe fips fcf (Show all detected FCFs)
Total number of FCFs detected: 1

      FCF MAC      Port      Vlan
-----
00:05:73:ce:96:68  PCH19    2002

VFSM# show fcoe fips fcoe (Show all FCoE connections)
Total number of FCoE connections: 1

      VN_PORT MAC      FCF MAC      Port      Vlan
-----
0e:fc:00:c6:01:80    00:05:73:ce:96:68  INT6      2002
```

## Operational Commands

The administrator may use the operational commands to delete FIP-related entries from the switch.

To delete a specific FCF entry and all associated ACLs from the switch, use the following command:

```
VFSM# no fcoe fips fcf <FCF MAC address> [<VLAN number>]
```

## FIP Snooping Configuration

In this example, as shown in [Figure 28 on page 262](#), FCoE devices are connected to ports BR5A through BR5D for the FCF bridge connection, and INT1 for an ENode. FIP snooping can be configured on these ports using the following CLI commands:

1. Set up the bridge module connection and reset the switch:

```
VFSM(config)# boot bridge-module 5 bandwidth 40
VFSM(config)# boot bridge-module 5 enable
VFSM(config)# reload
```

**Note:** This configuration automatically assigns four external switch ports for internal use between the VFSM and the bridge module. The external physical connectors will be disabled for the ports used by the bridge connection.

2. Disable STP on the bridge ports.

```
VFSM(config)# interface port br5a-b45d
VFSM(config-if)# no spanning-tree stp 1 enable
VFSM(config-if)# exit
```

**Note:** If the bridge ports exist in more than one Spanning Tree Group (STG), disable STP for the bridge ports in each STG in which they appear.

3. If using an Emulex Virtual Fabric Adapter, turn automatic VLAN creation off. Disable FIP snooping on all non-FCoE external ports:

```
VFSM(config)# no fcoe fips automatic-vlan
VFSM(config)# interface port ext5-ext10
VFSM(config-if)# no fcoe fips enable
VFSM(config-if)# exit
```

4. Turn CEE on.

```
VFSM(config)# cee enable
```

5. Turn global FIP snooping on:

```
VFSM(config)# fcoe fips enable
```

6. Configure the FCoE ports and VLAN.

Enable VLAN tagging/trunk mode on all FCoE ports, enable PVID/Native VLAN tagging on ENode ports, and then place all FCoE ports into a VLAN supported by the FCF (typically VLAN 1002):

```
VFSM(config)# interface port br5a-br5d
VFSM(config-if)# switchport mode trunk           (Enable VLAN tagging)
VFSM(config-if)# switchport trunk allowed vlan 1002
VFSM(config-if)# exit                             (Exit port configuration mode)
```

**Note:** If you are using Emulex CNA BE 2 - vNIC-FCoE mode, you must enable PVID/Native VLAN tagging on Enode ports.

**Note:** Placing ports into the VLAN *after* tagging/trunk mode is enabled helps to ensure that their port VLAN ID (PVID)/Native VLAN is not accidentally changed.

7. (Set by default) Enable FIP snooping on FCoE ports, and set the desired FCF mode:

```
VFSM(config)# fcoe fips port INTA1 enable        (Enable FIPS on ENode port)
VFSM(config)# fcoe fips port INTA1 fcf-mode off  (Set as ENode connection)
VFSM(config)# fcoe fips port br5a-br5d fcf-mode on (Set as FCF connection)
```

8. Save the configuration.

---

## Priority-Based Flow Control

Priority-based Flow Control (PFC) is defined in IEEE 802.1Qbb. PFC extends the IEEE 802.3x standard flow control mechanism. Under standard flow control, when a port becomes busy, the switch manages congestion by pausing all the traffic on the port, regardless of the traffic type. PFC provides more granular flow control, allowing the switch to pause specified types of traffic on the port, while other traffic on the port continues.

PFC pauses traffic based on 802.1p priority values in the VLAN tag. The administrator can assign different priority values to different types of traffic and then enable PFC for up to two specific priority values: priority value 3, and one other. The configuration can be applied on a port-by-port basis, or globally for all ports on the switch. Then, when traffic congestion occurs on a port (caused when ingress traffic exceeds internal buffer thresholds), only traffic with priority values where PFC is enabled is paused. Traffic with priority values where PFC is disabled proceeds without interruption but may be subject to loss if port ingress buffers become full.

Although PFC is useful for a variety of applications, it is required for FCoE implementation where storage (SAN) and networking (LAN) traffic are converged on the same Ethernet links. Typical LAN traffic tolerates Ethernet packet loss that can occur from congestion or other factors, but SAN traffic must be lossless and requires flow control.

For FCoE, standard flow control would pause both SAN and LAN traffic during congestion. While this approach would limit SAN traffic loss, it could degrade the performance of some LAN applications that expect to handle congestion by dropping traffic. PFC resolves these FCoE flow control issues. Different types of SAN and LAN traffic can be assigned different IEEE 802.1p priority values. PFC can then be enabled for priority values that represent SAN and LAN traffic that must be paused during congestion, and disabled for priority values that represent LAN traffic that is more loss-tolerant.

PFC requires CEE to be turned on ([“Turning CEE On or Off” on page 268](#)). When CEE is turned on, PFC is enabled on priority value 3 by default. Optionally, the administrator can also enable PFC on one other priority value, providing lossless handling for another traffic type, such as for a business-critical LAN application.

**Note:** For any given port, only one flow control method can be implemented at any given time: either PFC or standard IEEE 802.3x flow control.

## Global vs. Port-by-Port Configuration

PFC requires CEE to be turned on ([“Turning CEE On or Off” on page 268](#)). When CEE is turned on, standard flow control is disabled on all ports, and PFC is enabled on all ports for 802.1p priority value 3. This default is chosen because priority value 3 is commonly used to identify FCoE traffic in a CEE environment and must be guaranteed lossless behavior. PFC is disabled for all other priority values by default, but can be enabled for one additional priority value.

The administrator can also configure PFC on a port-by-port basis. The method used will typically depend on the following:

- Port-by-port PFC configuration is desirable in most mixed environments where some VFSM ports are connected to CEE-capable (FCoE) switches, gateways, and Converged Network Adapters (CNAs), and other VFSM ports are connected to non-CEE Layer 2/Layer 3 switches, routers and Network Interface Cards (NICs).
- Global PFC configuration is preferable in networks that implement end-to-end CEE devices. For example, if all ports are involved with FCoE and can use the same SAN and LAN priority value configuration with the same PFC settings, global configuration is easy and efficient.
- Global PFC configuration can also be used in some mixed environments where traffic with PFC-enabled priority values occurs only on ports connected to CEE devices, and not on any ports connected to non-CEE devices. In such cases, PFC can be configured globally on specific priority values even though not all ports make use them.
- PFC is not restricted to CEE and FCoE networks. In any LAN where traffic is separated into different priorities, PFC can be enabled on priority values for loss-sensitive traffic. If all ports have the same priority definitions and utilize the same PFC strategy, PFC can be globally configured.

**Note:** When using global PFC configuration in conjunction with the ETS feature (see [“Enhanced Transmission Selection” on page 281](#)), ensure that only pause-tolerant traffic (such as lossless FCoE traffic) is assigned priority values where PFC is enabled. Pausing other types of traffic can have adverse effects on LAN applications that expect uninterrupted traffic flow and tolerate dropping packets during congestion. Use PFC globally only if all priority values assigned for lossless traffic on one or more ports does not carry loss-tolerant traffic on other ports.

## PFC Configuration Example

**Note:** DCBX may be configured to permit sharing or learning PFC configuration with or from external devices. This example assumes that PFC configuration is being performed manually. See [“Data Center Bridging Capability Exchange” on page 287](#) for more information on DCBX.

This example is consistent with the network shown in [Figure 28 on page 262](#). In this example, the following topology is used.

Table 26. Port-Based PFC Configuration

Switch Port	802.1p Priority	Usage	PFC Setting
EXT5	0-2	LAN	Disabled
	3	(not used)	<b>Enabled</b>
	4	Business-critical LAN	<b>Enabled</b>
	others	(not used)	Disabled
BR5A	3	FCoE (to FCF bridge)	<b>Enabled</b>
	others	(not used)	Disabled
BR5B	3	FCoE (to FCF bridge)	<b>Enabled</b>
	others	(not used)	Disabled
INT1	3	FCoE	<b>Enabled</b>
	others	(not used)	Disabled
INT2	0-2	LAN	Disabled
	3	(not used)	<b>Enabled</b>
	4	Business-critical LAN	<b>Enabled</b>
	others	(not used)	Disabled

In this example, PFC is to facilitate lossless traffic handling for FCoE (priority value 3) and a business-critical LAN application (priority value 4).

Assuming that CEE is off (the VFSM default), the example topology shown in [Table 26](#) can be configured using the following commands:

1. Turn CEE on.

```
VFSM(config)# cee enable
```

2. Enable PFC for the FCoE traffic.

**Note:** PFC is enabled on priority 3 by default. If using the defaults, the manual configuration commands shown in this step are not necessary.

VFSM(config)# cee port INTA1 pfc priority 3 enable	(FCoE priority)
VFSM(config)# cee port INTA1 pfc priority 3 description "FCoE"	(Optional)
VFSM(config)# cee port br5b pfc priority 3 enable	(FCoE priority)
VFSM(config)# cee port br5b pfc priority 3 description "FCoE"	(Optional)

3. Enable PFC for the business-critical LAN application:

VFSM(config)# cee port INTA2 pfc priority 4 enable	(LAN priority)
VFSM(config)# cee port INTA2 pfc priority 4 description "Critical LAN"	
VFSM(config)# cee port EXT5 pfc priority 4 enable	(LAN priority)
VFSM(config)# cee port EXT5 pfc priority 4 description "Critical LAN"	

4. Save the configuration.

## Enhanced Transmission Selection

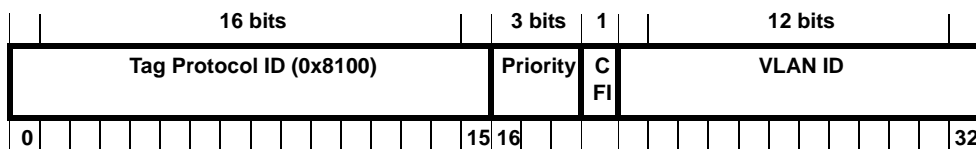
Enhanced Transmission Selection (ETS) is defined in IEEE 802.1Qaz. ETS provides a method for allocating port bandwidth based on 802.1p priority values in the VLAN tag. Using ETS, different amounts of link bandwidth can be specified for different traffic types (such as for LAN, SAN, and management).

ETS is an essential component in a CEE environment that carries different types of traffic, each of which is sensitive to different handling criteria, such as Storage Area Networks (SANs) that are sensitive to packet loss, and LAN applications that may be latency-sensitive. In a single converged link, such as when implementing FCoE, ETS allows SAN and LAN traffic to coexist without imposing contrary handling requirements upon each other.

The ETS feature requires CEE to be turned on (see [“Turning CEE On or Off” on page 268](#)).

### 802.1p Priority Values

Under the 802.1p standard, there are eight available priority values, with values numbered 0 through 7, which can be placed in the priority field of the 802.1Q VLAN tag:



Servers and other network devices may be configured to assign different priority values to packets belonging to different traffic types (such as SAN and LAN).

ETS uses the assigned 802.1p priority values to identify different traffic types. The various priority values are assigned to priority groups (PGID), and each priority group is assigned a portion of available link bandwidth.

Priorities values within in any specific ETS priority group are expected to have similar traffic handling requirements with respect to latency and loss.

802.1p priority values may be assigned by the administrator for a variety of purposes. However, when CEE is turned on, the VFMS sets the initial default values for ETS configuration as follows:

Figure 29. Default ETS Priority Groups

Typical Traffic Type	802.1p Priority	PGID	Bandwidth Allocation
LAN	0	0	10%
LAN	1		
LAN	2		
SAN	3	1	50%
Latency-Sensitive LAN	4	2	40%
Latency-Sensitive LAN	5		
Latency-Sensitive LAN	6		
Latency-Sensitive LAN	7		

In the assignment model shown in [Figure 29 on page 281](#), priorities values 0 through 2 are assigned for regular Ethernet traffic, which has “best effort” transport characteristics.

Priority 3 is typically used to identify FCoE (SAN) traffic.

Priorities 4-7 are typically used for latency sensitive traffic and other important business applications. For example, priority 4 and 5 are often used for video and voice applications such as IPTV, Video on Demand (VoD), and Voice over IP (VoIP). Priority 6 and 7 are often used for traffic characterized with a “must get there” requirement, with priority 7 used for network control which requires guaranteed delivery to support configuration and maintenance of the network infrastructure.

**Note:** The default assignment of 802.1p priority values on the VFSM changes depending on whether CEE is on or off. See [“Turning CEE On or Off” on page 268](#) for details.

## Priority Groups

For ETS use, each 801.2p priority value is assigned to a priority group which can then be allocated a specific portion of available link bandwidth. To configure a priority group, the following is required:

- CEE must be turned on ([“Turning CEE On or Off” on page 268](#)) for the ETS feature to function.
- A priority group must be assigned a priority group ID (PGID), one or more 802.1p priority values, and allocated link bandwidth greater than 0%.

## PGID

Each priority group is identified with number (0 through 6, and 15) known as the PGID.

PGID 0 through 6 may each be assigned a portion of the switch’s available bandwidth.

PGID 8 through 14 are reserved as per the 802.1Qaz ETS standard.

PGID 15 is a strict priority group. It is generally used for critical traffic, such as network management. Any traffic with priority values assigned to PGID 15 is permitted as much bandwidth as required, up to the maximum available on the switch. After serving PGID 15, any remaining link bandwidth is shared among the other groups, divided according to the configured bandwidth allocation settings.

All 802.1p priority values assigned to a particular PGID should have similar traffic handling requirements. For example, PFC-enabled traffic should not be grouped with non-PFC traffic. Also, traffic of the same general type should be assigned to the same PGID. Splitting one type of traffic into multiple 802.1p priorities, and then assigning those priorities to different PGIDs may result in unexpected network behavior.

Each 802.1p priority value may be assigned to only one PGID. However, each PGID may include multiple priority values. Up to eight PGIDs may be configured at any given time.

## Assigning Priority Values to a Priority Group

Each priority group may be configured from its corresponding ETS Priority Group, available using the following command:

```
VFSM(config)# cee global ets priority-group pgid <group number (0-7, or 15)> priority  
<priority list>
```

where *priority list* is one or more 802.1p priority values (with each separated by a space). For example, to assign priority values 0 through 2:

```
VFSM(config)# cee global ets priority-group pgid <group number (0-7, or 15)> priority 0,1,2
```

**Note:** Within any specific PGID, the PFC settings (see [“Priority-Based Flow Control” on page 277](#)) should be the same (enabled or disabled) for all priority values within the group. PFC can be enabled only on priority value 3 and one other priority. If the PFC setting is inconsistent within a PGID, a warning message is displayed when attempting to apply the configuration.

When assigning priority values to a PGID, the specified priority value will be automatically removed from its old group and assigned to the new group when the configuration is applied.

Each priority value must be assigned to a PGID. Priority values may not be deleted or unassigned. To remove a priority value from a PGID, it must be moved to another PGID.

For PGIDs 0 through 7, bandwidth allocation can also be configured through the ETS Priority Group menu. See for [“Allocating Bandwidth” on page 283](#) for details.

## Deleting a Priority Group

A priority group is automatically deleted when it contains no associated priority values, and its bandwidth allocation is set to 0%.

**Note:** The total bandwidth allocated to PGID 0 through 7 must equal exactly 100%. Reducing the bandwidth allocation of any group will require increasing the allocation to one or more of the other groups (see [“Allocating Bandwidth” on page 283](#)).

## Allocating Bandwidth

### Allocated Bandwidth for PGID 0 Through 7

The administrator may allocate a portion of the switch's available bandwidth to PGIDs 0 through 7. Available bandwidth is defined as the amount of link bandwidth that remains after priorities within PGID 15 are serviced (see [“Unlimited Bandwidth for PGID 15” on page 284](#)), and assuming that all PGIDs are fully subscribed. If any PGID does not fully consume its allocated bandwidth, the unused portion is made available to the other priority groups.

Priority group bandwidth allocation can be configured using the following command:

```
VFSM(config)# cee global ets priority-group pgid <priority group number> bandwidth  
<bandwidth allocation> pgid <priority group number> bandwidth <bandwidth allocation>
```

where *bandwidth allocation* represents the percentage of link bandwidth, specified as a number between 0 and 100, in 1% increments.

The following bandwidth allocation rules apply:

- Bandwidth allocation must be 0% for any PGID that has no assigned 802.1p priority values.
- Any PGID assigned one or more priority values must have a bandwidth allocation greater than 0%.
- Total bandwidth allocation for groups 0 through 7 must equal exactly 100%. Increasing or reducing the bandwidth allocation of any PGID also requires adjusting the allocation of other PGIDs to compensate.

If these conditions are not met, the switch will report an error when applying the configuration.

To achieve a balanced bandwidth allocation among the various priority groups, packets are scheduled according to a weighted deficit round-robin (WDRR) algorithm. WDRR is aware of packet sizes, which can vary significantly in a CEE environment, making WDRR more suitable than a regular weighted round-robin (WRR) method, which selects groups based only on packet counts.

**Note:** Actual bandwidth used by any specific PGID may vary from configured values by up to 10% of the available bandwidth in accordance with 802.1Qaz ETS standard. For example, a setting of 10% may be served anywhere from 0% to 20% of the available bandwidth at any given time.

### Unlimited Bandwidth for PGID 15

PGID 15 is permitted unlimited bandwidth and is generally intended for critical traffic (such as switch management). Traffic in this group is given highest priority and is served before the traffic in any other priority group.

If PGID 15 has low traffic levels, most of the switch's bandwidth will be available to serve priority groups 0 through 7. However, if PGID 15 consumes a larger part of the switch's total bandwidth, the amount available to the other groups is reduced.

**Note:** Consider traffic load when assigning priority values to PGID 15. Heavy traffic in this group may restrict the bandwidth available to other groups.

## Configuring ETS

Consider an example consistent with that used for port-based PFC configuration (on [page 279](#)):

Table 27. ETS Configuration

Priority	Usage	PGID	Bandwidth
0	LAN (best effort delivery)	0	10%
1	LAN (best effort delivery)		
2	LAN (best effort delivery)		
3	SAN (Fibre Channel over Ethernet, with PFC)	1	20%
4	Business Critical LAN (lossless Ethernet, with PFC)	2	30%

Table 27. ETS Configuration (continued)

Priority	Usage	PGID	Bandwidth
5	Latency-sensitive LAN	3	40%
6	Latency-sensitive LAN		
7	Network Management (strict)	15	unlimited

The example shown in [Table 27](#) is only slightly different than the default configuration shown in [Figure 29 on page 281](#). In this example, latency-sensitive LAN traffic (802.1p priority 5 through 6) are moved from priority group 2 to priority group 3. This leaves Business Critical LAN traffic (802.1p priority 4) in priority group 2 by itself. Also, a new group for network management traffic has been assigned. Finally, the bandwidth allocation for priority groups 1, 2, and 3 are revised.

**Note:** DCBX may be configured to permit sharing or learning PFC configuration with or from external devices. This example assumes that PFC configuration is being performed manually. See [“Data Center Bridging Capability Exchange” on page 287](#) for more information on DCBX.

This example can be configured using the following commands:

1. Turn CEE on.

```
VFSM(config)# cee enable
```

2. Configure each allocated priority group with a description (optional), list of 802.1p priority values, and bandwidth allocation:

```
VFSM(config)# cee global ets priority-group pgid 0 priority 0,1,2
                                     (Select a group for regular LAN, and
                                     set for 802.1p priorities 0, 1, and 2)
VFSM(config)# cee global ets priority-group pgid 0 description "Regular LAN"
                                     (Set a group description—optional)
VFSM(config)# cee global ets priority-group pgid 1 priority 3
                                     (Select a group for SAN traffic, and
                                     set for 802.1p priority 3)
VFSM(config)# cee global ets priority-group pgid 1 description "SAN"
                                     (Set a group description—optional)
VFSM(config)# cee global ets priority-group pgid 2 priority 4
                                     (Select a group for latency traffic,
                                     and set for 802.1p priority 4)
VFSM(config)# cee global ets priority-group pgid 2 description "Biz-Critical
LAN"
                                     (Set a group description—optional)
VFSM(config)# cee global ets priority-group pgid 3 description
"Latency-Sensitive LAN"
                                     (Set a group description—optional)
VFSM(config)# cee global ets priority-group pgid 3 priority 5,6 pgid 0 bandwidth
10 pgid 1 bandwidth 20 pgid 2 bandwidth 30 pgid 3 bandwidth 40
                                     (Configure link bandwidth restriction)
```

3. Configure the strict priority group with a description (optional) and a list of 802.1p priority values:

```
VFSM(config)# cee global ets priority-group pgid 15 priority 7
VFSM(config)# cee global ets priority-group pgid 15 description
"Network Management"
```

**Note:** Priority group 15 is permitted unlimited bandwidth. As such, the commands for priority group 15 do not include bandwidth allocation.

4. Save the configuration.

---

## Data Center Bridging Capability Exchange

Data Center Bridging Capability Exchange (DCBX) protocol is a vital element of CEE. DCBX allows peer CEE devices to exchange information about their advanced capabilities. Using DCBX, neighboring network devices discover their peers, negotiate peer configurations, and detect misconfigurations.

DCBX provides two main functions on the VFSM:

- Peer information exchange  
The switch uses DCBX to exchange information with connected CEE devices. For normal operation of any FCoE implementation on the VFSM, DCBX must remain enabled on all ports participating in FCoE.
- Peer configuration negotiation  
DCBX also allows CEE devices to negotiate with each other for the purpose of automatically configuring advanced CEE features such as PFC, ETS, and (for some CNAs) FIP. The administrator can determine which CEE feature settings on the switch are communicated to and matched by CEE neighbors, and also which CEE feature settings on the switch may be configured by neighbor requirements.

The DCBX feature requires CEE to be turned on (see [“Turning CEE On or Off” on page 268](#)).

## DCBX Settings

When CEE is turned on, DCBX is enabled for peer information exchange on all ports. For configuration negotiation, the following default settings are configured:

- Application Protocol: FCoE and FIP snooping is set for traffic with 802.1p priority 3
- PFC: Enabled on 802.1p priority 3
- ETS
  - Priority group 0 includes priority values 0 through 2, with bandwidth allocation of 10%
  - Priority group 1 includes priority value 3, with bandwidth allocation of 50%
  - Priority group 2 includes priority values 4 through 7, with bandwidth allocation of 40%

## Enabling and Disabling DCBX

When CEE is turned on, DCBX can be enabled and disabled on a per-port basis, using the following commands:

```
VFSM(config)# [no] cee port <port alias or number> dcbx enable
```

When DCBX is enabled on a port, Link Layer Detection Protocol (LLDP) is used to exchange DCBX parameters between CEE peers. Also, the interval for LLDP transmission time is set to one second for the first five initial LLDP transmissions, after which it is returned to the administratively configured value. The minimum delay between consecutive LLDP frames is also set to one second as a DCBX default.

## Peer Configuration Negotiation

CEE peer configuration negotiation can be set on a per-port basis for a number of CEE features. For each supported feature, the administrator can configure two independent flags:

- The `advertise` flag

When this flag is set for a particular feature, the switch settings will be transmit to the remote CEE peer. If the peer is capable of the feature, and willing to accept the VFSM settings, it will be automatically reconfigured to match the switch.

- The `willing` flag

Set this flag when required by the remote CEE peer for a particular feature as part of DCBX signaling and support. Although some devices may also expect this flag to indicate that the switch will accept overrides on feature settings, the VFSM retains its configured settings. As a result, the administrator should configure the feature settings on the switch to match those expected by the remote CEE peer.

These flags are available for the following CEE features:

- Application Protocol

DCBX exchanges information regarding FCoE and FIP snooping, including the 802.1p priority value used for FCoE traffic. The `advertise` flag is set or reset using the following command:

```
VFSM(config)# [no] cee port <port alias or number> dcbx app_proto advertise
```

The `willing` flag is set or reset using the following command:

```
VFSM(config)# [no] cee port <port alias or number> dcbx app_proto willing
```

- PFC

DCBX exchanges information regarding whether PFC is enabled or disabled on the port. The `advertise` flag is set or reset using the following command:

```
VFSM(config)# [no] cee port <port alias or number> dcbx pfc advertise
```

The `willing` flag is set or reset using the following command:

```
VFSM(config)# [no] cee port <port alias or number> dcbx pfc willing
```

- ETS

DCBX exchanges information regarding ETS priority groups, including their 802.1p priority members and bandwidth allocation percentages. The `advertise` flag is set or reset using the following command:

```
VFSM(config)# [no] cee port <port alias or number> dcbx ets advertise
```

The `willing` flag is set or reset using the following command:

```
VFSM(config)# [no] cee port <port alias or number> dcbx ets willing
```

## Configuring DCBX

Consider an example consistent [Figure 28 on page 262](#) and used with the previous FCoE examples in this chapter:

- FCoE is used on ports INT1, BR5A and BR5B.
- CEE features are also used with LANs on ports INT2 and EXT5.
- All other ports are disabled or are connected to regular (non-CEE) LAN devices.

In this example, the VFSM acts as the central point for CEE configuration. FCoE-related ports will be configured for advertising CEE capabilities, but not to accept external configuration. Other LAN ports that use CEE features will also be configured to advertise feature settings to remote peers, but not to accept external configuration. DCBX will be disabled on all non-CEE ports.

This example can be configured using the following commands:

1. Turn CEE on.

```
VFSM(config)# cee enable
```

2. Enable desired DCBX configuration negotiation on FCoE ports:

```
VFSM(config)# cee port INTA1 dcbx enable
VFSM(config)# cee port INTA1 dcbx app_proto advertise
VFSM(config)# cee port INTA1 dcbx ets advertise
VFSM(config)# cee port INTA1 dcbx pfc advertise

VFSM(config)# cee port EXT4 dcbx enable
VFSM(config)# cee port EXT4 dcbx app_proto advertise
VFSM(config)# cee port EXT4 dcbx ets advertise
VFSM(config)# cee port EXT4 dcbx pfc advertise
```

3. Enable desired DCBX advertisements on other CEE ports:

```
VFSM(config)# cee port INTA2 dcbx enable
VFSM(config)# cee port INTA2 dcbx ets advertise
VFSM(config)# cee port INTA2 dcbx pfc advertise

VFSM(config)# cee port EXT5 dcbx enable
VFSM(config)# cee port EXT5 dcbx ets advertise
VFSM(config)# cee port EXT5 dcbx pfc advertise
```

4. Disable DCBX for each non-CEE port as appropriate:

```
VFSM(config)# no cee port INTA3,INTA4-EXT11 enable
```

5. Save the configuration.



---

## Chapter 16. Edge Virtual Bridging

The 802.1Qbg/Edge Virtual Bridging (EVB) is an emerging IEEE standard for allowing networks to become virtual machine (VM)-aware. EVB bridges the gap between physical and virtual network resources. The IEEE 802.1Qbg simplifies network management by providing a standards-based protocol that defines how virtual Ethernet bridges exchange configuration information. In EVB environments, physical end stations, containing multiple virtual end stations, use a bridge to form a LAN. The virtual NIC (vNIC) configuration information of a virtual machine is available to these EVB devices. This information is generally not available to an 802.1Q bridge.

IBM Networking OS EVB features are compliant with the IEEE 802.1Qbg Authors Group Draft 0.2. For a list of documents on this feature, see: <http://www.ieee802.org/1/pages/802.1bg.html>.

The IBM Virtual Fabric 10Gb Switch Module performs the role of a 802.1Qbg bridge in an EVB environment.

IBM N/OS implementation of EVB supports the following protocols:

- Virtual Ethernet Bridging (VEB) and Virtual Ethernet Port Aggregator (VEPA): VEB and VEPA are mechanisms for switching between VMs on the same hypervisor. VEB enables switching with the server, either in the software (vSwitch), or in the hardware (using single root I/O virtualization capable NICs). VEPA requires the edge switch to support “Reflective Relay”— an operation where the switch forwards a frame back to the port on which it arrived if the destination MAC address is on the same port.
- Edge Control Protocol (ECP): ECP is a transport protocol that operates between two peers over an IEEE 802 LAN. ECP provides reliable, in-order delivery of ULP (Upper Layer Protocol) PDUs (Protocol Data Units).
- Virtual Station Interface (VSI) Discovery and Configuration Protocol (VDP): VDP allows hypervisors to advertise VSIs to the physical network. This protocol also allows centralized configuration of network policies that will persist with the VM, independent of its location.
- EVB Type-Length-Value (TLV): EVB TLV is a Link Layer Discovery protocol (LLDP)-based TLV used to discover and configure VEPA, ECP, and VDP.

---

## EVB Operations Overview

The N/OS includes a pre-standards VSI Type Database (VSIDB) implemented through the System Networking Switch Center (SNSC) or the IBM System Networking Distributed Switch 5000V. The VSIDB is the central repository for defining sets of network policies that apply to VM network ports. You can configure only one VSIDB.

**Note:** This document does not include the VSIDB configuration details. Please see the SNSC or IBM System Networking Distributed Switch 5000V guide for details on how to configure VSIDB.

The VSIDB operates in the following sequence:

1. Define VSI types in the VSIDB. The VSIDB exports the database when the VFSM sends a request.
2. Create a VM. Specify VSI type for each VM interface. See the SNSC or IBM System Networking Distributed Switch 5000V guide for details on how to specify the VSI type.

The hypervisor sends a VSI ASSOCIATE, which contains the VSI type ID, to the switch port after the VM is started. The switch updates its configuration based on the requested VSI type. The switch configures the per-VM bandwidth using the VMpolicy.

The N/OS supports the following policies for VMs:

- ACLs
- Bandwidth metering

## VSIDB Synchronization

The switch periodically checks for VSIDB changes based on the configured interval. You can configure this interval using the following command:

```
VFSM(config)# virt evb vsidb <number>
VFSM(conf-vsidb)# update-interval <time in seconds>
```

To disable periodic updates, configure the interval value as 0.

If the switch finds that the VSIDB has changed, it updates the local VSIDB cache. When the cache is successfully updated, it sends a syslog message.

After updating the local VSIDB cache, the switch disassociates any VM whose type ID or VLAN no longer exists in the updated cache.

The switch updates the local VSIDB cache when any of the following takes place:

- When, at the configured refresh interval, the switch finds that the VSIDB configuration has changed since the last poll.
- When a VM sends an ASSOCIATE message, but the VSI type does not exist in the local VSIDB cache.
- When a VM sends an ASSOCIATE message, and the VSI type exists but the VSI type's VLAN ID does not exist in the local VSIDB cache.
- When you update the VSIDB using the following command:  
VFSM# virt evb update vsidb <number>

- When the management port link status changes from down to up

## VLAN Behavior

When a VM gets associated, the corresponding VLAN is dynamically created on the switch port if the VLAN does not already exist.

VLANs that are dynamically created will be automatically removed from the switch port when there are no VMs using that VLAN on the port.

Dynamic VLAN information will not be displayed in the running configuration. However, the VLAN, port, and STP commands display the dynamic VLAN information with a “\*”.

If you configure any Layer 2/Layer 3 features on dynamically created VLANs, the VLAN information is displayed in the running configuration.

### Deleting a VLAN

If you delete a VLAN that has a VM associated with it, you will see a warning message similar to the following:

```
Warning: Vlan 10 is used by VM and can't be removed.
```

The VMs will not get disassociated.

If a VM is associated with a port, and you remove this port from a VLAN, you will see a warning message similar to the following:

```
Warning: Port 23 in Vlan 10 is used by VM and can't be removed.
```

The VMs will not get disassociated.

## Manual Reflective Relay

Reflective Relay (RR) is an operation where the switch forwards a frame back to the port on which it arrived if the destination MAC address is on the same port. When an EVB profile is configured on a port, RR is automatically enabled on the port after capability exchange with the peer, using the IEEE802.1QBG protocol. This is the usual mode of operation.

When the switch interoperates with devices that do not support IEEE 802.1QBG protocols, RR can be manually configured using the following command:

```
VFSM(config-if)# reflective-relay force
```

Manual RR and EVB profile cannot be configured on a port at the same time.

---

## EVB Configuration

This section includes the steps to configure EVB based on the following values:

- Profile number: 1
- Port number: INT1
- Retry interval: 8000 milliseconds
- VSI Database:
  - Manager IP: 172.31.37.187
  - Port: 80

**Note:** VSI Database can be accessed via HTTP or HTTPS. Also, the manager IP can be configured with and IPv4 or IPv6 address.

1. Create an EVB profile.

```
VFSM(config)# virt evb profile 1
```

*(Enter number from 1-16)*

2. Enable Reflective Relay.

```
VFSM(config-evbprof)# reflective-relay
```

3. Enable VSI discovery.

```
VFSM(config-evbprof)# vsi-discovery  
VFSM(config-evbprof)# exit
```

4. Add EVB profile to port.

```
VFSM(config)# interface port 1  
VFSM(config-if)# evb profile 1  
VFSM(config-if)# exit
```

*(Enter EVB profile ID)*

5. Configure ECP retransmission interval.

```
VFSM(config)# ecp retransmit-interval 8000
```

*(Enter retransmission interval in milliseconds (100-9000))*

6. Set VSI database information.

```
VFSM(config)# virt evb vsidb 1  
VFSM(config-vsidb)# protocol {http|https} (Select VSI database protocol; default is HTTP)  
VFSM(config-vsidb)# host 172.31.37.187 (Set VSI database Manager IP)  
VFSM(config-vsidb)# port 80 (Set VSI database Manager port)  
VFSM(config-vsidb)# filepath "vsidb" (Set VSI database document path)  
VFSM(config-vsidb)# filename "all.xml" (Set VSI database file name)  
VFSM(config-vsidb)# update-interval 30 (Set update interval in seconds)  
VFSM(config-vsidb)# exit
```

**Note:** When you connect to a SNSC VSIDB, the port/docpath configuration is as follows:

- Port: 40080
- Docpath: bhm/rest/vsitypes

When you connect to a 5000v VSIDB, the port/docpath configuration is as follows:

- Port: 80
- Docpath: vsitypes

7. Enable LLDP.

```
VFSM(config)# lldp enable
```

*(Turn on LLDP)*

---

## Configuring EVB in Stacking Mode

A *stack* is a group of up to [eight] Virtual Fabric Switch Module switches with IBM Networking OS that work together as a unified system. The switches in a stack are interconnected by a stack trunk in a local ring topology.

An operational stack must contain one Master and one or more Members, as follows:

- **Master**

One switch controls the operation of the stack and is called the Master. The Master provides a single point to manage the stack. A stack must have one and only one Master. The firmware image, configuration information, and run-time data are maintained by the Master and pushed to each switch in the stack as necessary.

- **Member**

Member switches provide additional port capacity to the stack. Members receive configuration changes, run-time information, and software updates from the Master.

- **Backup**

One member switch can be designated as a Backup to the Master. The Backup takes over control of the stack if the Master fails. Configuration information and run-time data are synchronized with the Master.

For details on implementing the stacking feature, see [“Stacking” on page 195](#).

EVB can be configured on any port in the stack. Use the Master to configure EVB on a port in the stack. The port numbers in a stack use the following format:

<switch number>:<port number>

The Master process the EVB-related information for all the switch ports in a stack. The Master performs the VSIDB synchronization (See [“VSIDB Synchronization” on page 292](#)). The Master synchronizes all EVB changes with the Backup.

If the Master fails, the Backup takes over control of the stack. The VSI associations on the Master ports are lost. All other VSI associations remain unchanged.

---

## Limitations

- If both ACL and egress bandwidth metering are enabled, traffic will first be matched with the ACL and will not be limited by bandwidth metering.
- ACLs based on a source MAC or VLAN must match the source MAC and VLAN of the VM. If not, the policy will be ignored and you will see the following warning message:

```
"vm: VSI Type ID 100 Associated mac 00:50:56:b6:c0:ff on port 6,  
ignore 1 mismatched ACL"
```

---

## Unsupported features

The following features are not supported on ports configured with EVB:

- LAG/VLAG
- VMready
- vNIC



---

## Chapter 17. Static Multicast ARP

The Microsoft Windows operating system includes the Network Load Balancing (NLB) technology that helps to balance incoming IP traffic among multi-node clusters. In multicast mode, NLB uses a shared multicast MAC address with a unicast IP address. Since the address resolution protocol (ARP) can map an IP address to only one MAC address, port, and VLAN, the packet reaches only one of the servers (the one attached to the port on which the ARP was learnt).

To avoid the ARP resolution, you must create a static ARP entry with multicast MAC address. You must also specify the list of ports through which the multicast packet must be sent out from the gateway or Layer 2/Layer 3 node.

With these configurations, a packet with a unicast IPv4 destination address and multicast MAC address can be sent out as per the multicast MAC address configuration. NLB maps the unicast IP address and multicast MAC address as follows:

Cluster multicast MAC address: 03-BF-W-X-Y-Z; where W.X.Y.Z is the cluster unicast IP address.

You must configure the static multicast ARP entry only at the Layer 2/Layer 3 or Router node, and not at the Layer 2-only node.

IBM Networking OS supports a maximum of 20 static multicast ARP entries.

---

## Configuring Static Multicast ARP

To configure multicast MAC ARP, you must perform the following steps:

- Configure the static multicast forwarding database (FDB) entry: Since there is no port list specified for static multicast ARP, and the associated MAC address is multicast, you must specify a static multicast FDB entry for the cluster MAC address to limit the multicast domain. If there is no static multicast FDB entry defined for the cluster MAC address, traffic will not be forwarded. Use the following command:

```
VFSM(config)# mac-address-table multicast <cluster MAC address> <port(s)>
```

- Configure the static multicast ARP entry: Multicast ARP static entries should be configured without specifying the list of ports to be used. Use the following command:

```
VFSM(config)# ip arp <destination unicast IP address> <destination multicast MAC address> vlan <cluster VLAN number>
```

## Configuration Example

Consider the following example:

- Cluster unicast IP address: 10.10.10.42
- Cluster multicast MAC address: 03:bf:0A:0A:0A:2A
- Cluster VLAN: 42
- List of individual or port trunks to which traffic should be forwarded: 54 and 56

Following are the steps to configure the static multicast ARP based on the given example:

1. Configure the static multicast FDB entry.

```
VFSM(config)# mac-address-table multicast 03:bf:0A:0A:0A:2A 42 54,56
```

2. Configure the static multicast ARP entry:

```
VFSM(config)# ip arp 10.10.10.42 03:bf:0A:0A:0A:2A vlan 42
```

You can verify the configuration using the following commands:

- Verify static multicast FDB entry:

```
EN 4093R(config)# show mac-address-table multicast address 03:bf:0A:0A:0A:2A
```

Multicast Address	VLAN	Port(s)
03:bf:0A:0A:0A:2A	42	54 56

- Verify static multicast ARP entry:

```
EN 4093R(config)# show ip arp

Current ARP configuration:
  rearp 5
Current static ARP:
  ip          mac          port  vlan
  -----
  10.10.10.42  03:bf:0A:0A:0A:2A      42
  -----
Total number of arp entries : 2
  IP address  Flags  MAC address  VLAN  Age Port
  -----
  10.10.10.1  P      fc:cf:62:9d:74:00  42
  10.10.10.42  P      03:bf:0A:0A:0A:2A  42  0
```

## Limitations

- You must configure the ARP only in the Layer 2/Layer 3 node or the router node but not in the Layer 2-only node. IBM N/OS cannot validate if the node is Layer 2-only.
- The packet is always forwarded to all the ports as specified in the Multicast MAC address configuration. If VLAN membership changes for the ports, you must update this static multicast ARP entry. If not, the ports, whose membership has changed, will report discards.
- ACLs take precedence over static multicast ARP. If an ACL is configured to match and permit ingress of unicast traffic, the traffic will be forwarded based on the ACL rule, and the static multicast ARP will be ignored.



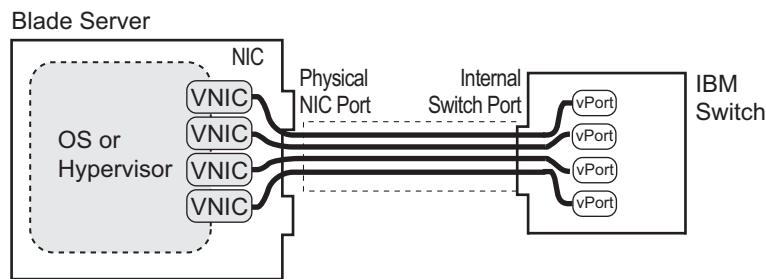
## Chapter 18. Unified Fabric Port

Unified Fabric Port (UFP) is a cost-effective way to allocate, share and dynamically control network bandwidth between a server and a switch. UFP lets you create multiple virtual connections. The UFP protocol is a link-level protocol that runs a separate instance for each physical communication link established between a server NIC and a switch port. Virtualizing the ports allows you to separate or aggregate port traffic by applying the network policies defined on the switch. Virtualization lessens bottlenecks and provides higher bandwidth while consolidating equipment use.

UFP provides a switch fabric component to control NICs. The server operating system (OS) or hypervisor recognizes each subdivided link (channel) as an independent physical NIC. Each channel has a unique identity and profile that defines its properties and functionality. The server communicates with the switch over the channel as defined in the channel profile. The channels share the high-speed physical link bandwidth.

For each channel, the vNIC on the server side communicates with virtual port on the switch side. Any 10 Gbps internal server port can be configured as an UFP port.

Figure 30. UFP vPorts



The UFP protocol has the following operation categories:

- **Channel Initialization:** The server NIC and the switch port negotiate the number of channels and establish channel identifiers. Each UFP channel has a data component and a control component. The two components have the same UFP channel ID.
- **Channel Control:** For an established channel, the switch can modify channel properties by sending a control message on the UFP channel. While the channel ID is the same for the control and data components, the destination MAC address of the control message frame is a well-known address.
- **Discovery Capability:** UFP can discover other ports that are UFP enabled. Once you enable UFP, you can check the information statistics for established channels.

---

## UFP Limitations

The following limitations apply when configuring UFP:

- FCoE must be configured only on vPort 2 of the physical NIC.
- UFP port in FCoE mode cannot operate with FIP auto-VLAN feature.
- VLANs that have member vPorts configured in trunk- or access-modes cannot have member vPorts configured in tunnel mode.
- vPorts on a physical port must be members of separate VLANs.
- VLANs 4002-4005 are reserved for outer tagging.
- VLAN translation is not applied on packets that egress an UFP port.
- UFP bandwidth is guaranteed lossless only for unicast traffic.
- UFP vPorts can support up to 256 VLANs.
- When CEE is turned on, FCoE vPort must be used for lossless priority traffic. For loss-tolerant priority traffic, a non-FCoE UFP vPort must be used. The lossless property of FCoE vPort is not guaranteed, if lossless and loss-tolerant traffic are combined.
- When the vPort is enabled and the channel link state is up, the system does not support changing vPort VLAN type from private/non-private to non-private/private.
- When integrating with private VLANs, UFP is not supported in stacking mode.
- A maximum of four vPorts can be configured for each physical switch port.
- UFP and vNIC cannot be configured on the same port.
- VMReady Local Group configuration is not supported by UFP.

---

## Virtual Ports Modes

A single physical switch port is configured with virtual ports (vPorts). Each UFP channel connects the server NIC with a switch vPort. Properties that are defined for a vPort, such as native VLAN and bandwidth, are applied to the traffic that belongs to the vPort.

**Note:** A maximum of four vPorts can be configured for each physical switch port.

### vPort-S-Tag Mapping

A vPort can also be identified with an S-tag (service tag or outer tag). When a vPort is initialized, the switch communicates the UFP channel ID of the vPort to the server NIC. When the server NIC or switch transmit frames, they add this S-tag to indicate the vPort or vNIC to which the packet is being transmitted. No VLAN mapping is required. Such packets can be single tagged or double tagged (with S-tag).

### vPort-VLAN Mapping

In port-VLAN mapping, the NIC can identify a particular vPort or vNIC based on port and VLAN tags from incoming/outgoing packets on a UFP port, because two different vPorts on the same UFP port will not carry the same VLAN traffic. A port paired with a VLAN can uniquely map to a particular vPort.

In local domain data path type, the switch and server identify the vPort and vNIC by the port and VLAN tag in the incoming and outgoing packets. Because no two vPorts carry traffic for the same VLAN, the port-and-VLAN combination must be uniquely mapped to a vPort.

## UFP vPort Mode

The UFP mode is configured based on the type of switching domain (single VLAN or multiple VLANs) where the vPort is connected.

- Use local domain data path types for trunk or access mode.
- Use pass-through domain data path types for tunnel mode. In tunnel mode, a vPort can belong to only one VLAN.

Use the following command to configure UFP vPort mode:

```
VFSM(config)# ufp port <num> vport <num>
VFSM(config_ufp_vport)# network mode {access|trunk|tunnel|fcoe}

Default mode is 'tunnel'
```

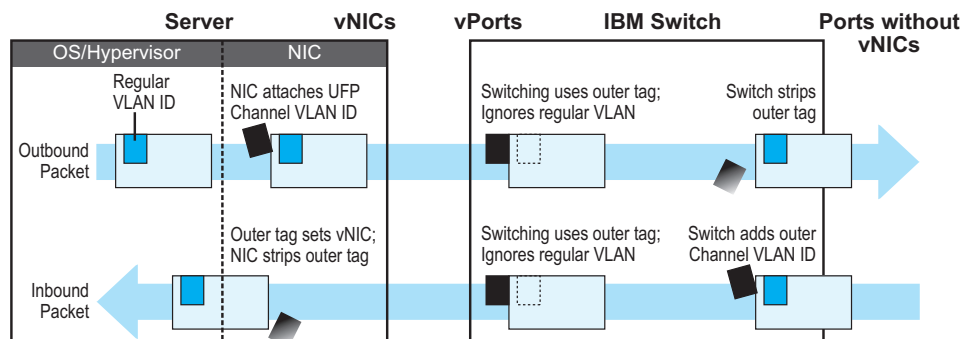
## Tunnel Mode

In tunnel mode, a vPort can belong to only one VLAN. An outer tag with the vPort's VLAN ID is inserted in packets that egress the vPort. The inner VLAN tag remains unchanged. The switch processes packets based on the outer tag. When all the ports or vPorts that belong to a particular VLAN are placed in tunnel mode, they belong to one pass-through domain.

Use tunnel mode to send all VM data traffic to an upstream switch, for Layer 2 or Layer 3 processing, in one domain. In such cases, the UFP port or vPort must be in tunnel mode and the upstream switch port must be in 802.1Q trunk mode.

**Note:** Two vPorts on a physical port cannot be members of the same VLAN.

Figure 31. Packet pass-through in Tunnel Mode

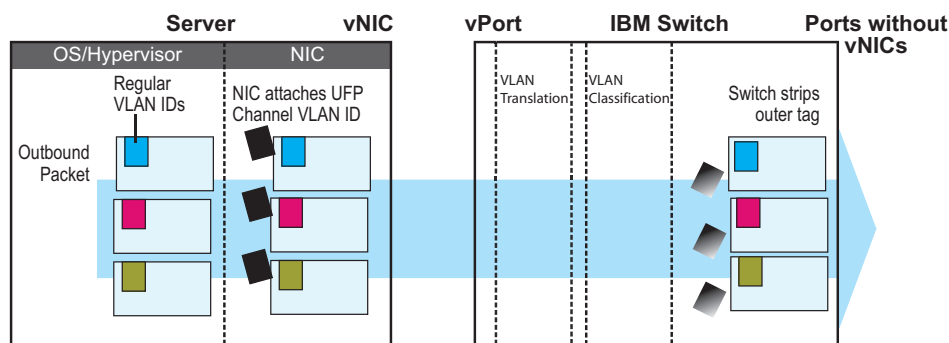


## 802.1Q Trunk Mode

In trunk mode, a vPort can carry packets that have inner tags that belong to up to 256 VLANs. A maximum of 2048 VLANs for all UFP vPorts can be configured on the switch in stand-alone mode. Each VLAN in the inner tag requires a VLAN translation entry.

**Note:** Two vPorts operating in trunk mode on the same physical port cannot carry the same set of VLANs in the inner tag.

Figure 32. Packet passing through in Trunk Mode



## Access Mode

In access mode, a vPort carries packets with inner tags that belong to one VLAN. The vPort is associated with the VLAN defined by the command:

```
VFSM(config_ufp_vport)# network default-vlan <2-4094>
```

**Note:** VLANs 4002-4005 are reserved for outer tagging.

## FCoE Mode

FCoE traffic is carried by a vPort. The server-side endpoint of this virtual port will be represented through a FC vHBA. Setting a virtual port in FCoE mode will enable Priority-based Flow Control (PFC) on the physical port.

A vPort configured in FCoE mode can only be attached to a Fibre Channel (FC) VLAN. Only vPort 2 on a physical port can be configured in FCoE mode. A vPort in FCoE mode operates as a local domain data path type with packets being single tagged.

---

## UFP Bandwidth Provisioning

UFP provides one mode of bandwidth provisioning for vPort: Strict Bandwidth Provisioning Mode.

### UFP Strict Bandwidth Provisioning Mode

Strict bandwidth provisioning mode configures the switch and NIC apply bidirectional bandwidth control on the vPort as per the defined configuration. By default, a bandwidth of 2.5 Gbps per vPort is guaranteed. If other vPorts are idle, the bandwidth of a vPort can be up to 10 Gbps. A minimum bandwidth of 1 Gbps is provisioned, which can be raised by 100 Mbps increments. The sum of the minimum bandwidth guaranteed for all vPorts together cannot exceed the capacity of the physical link. A vPort can also be configured with a maximum bandwidth.

This mode works with the port scheduler to avoid unintended packet drops due to policing through EFP metering block. If flow control is enabled, the switch provides a no-drop packet forwarding behavior, which improves end-to-end TCP-throughput performance.

**Note:** If a vPort is configured with low upper limit, it might lead to head-of-line congestion on the egress port.

By default, uplink ports have a separate traffic class for storage traffic with guaranteed bandwidth. The rest of the bandwidth is shared equally among other traffic.

Use the following command to configure strict bandwidth provisioning:

```
VFSM(config_ufp_vport)# qos bandwidth {max|min} <10-100>

min - Set minimum guaranteed bandwidth
max - Set maximum allowed bandwidth
```

---

## Using UFP with Other Virtual Fabric Switch Module Features

UFP works with other VFSM features, as described with limitations and details.

### Layer 2 Failover

UFP failover can be configured with auto-monitoring or manual monitoring. In auto-monitoring, a vPort is automatically associated with a Failover trigger if it has any VLAN in common with the monitor ports.

### Increased VLAN Limits

Configured with UFP and VLANs, a vPort can support maximum 256 VLANs. A UFP port supports 256 VLANs.

For more information on VLAN configuration, see [“VLANs” on page 123](#).

---

## UFP Configuration Examples

Following is an example configuration of UFP vPorts in access mode.

### Example 1: Access Mode

1. Turn on UFP.

```
VFSM(config)# ufp enable
```

2. Configure internal port as UFP.

```
VFSM(config)# ufp port INT1 enable  
  
Warning: "Tagging/Trunk-mode" is enabled on UFP port INTA1
```

3. Configure virtual port.

```
VFSM(config)# ufp port INT1 vport 1
```

4. Configure vPort access mode.

```
VFSM(config_ufp_vport)# network mode access
```

5. Configure vPort default VLAN.

```
VFSM(config_ufp_vport)# network default-vlan 100
```

6. Ensure tagging is disabled on vPort.

```
VFSM(config_ufp_vport)# no network default-tag
```

7. Specify QoS parameters for the vPort.

```
VFSM(config_ufp_vport)# qos bandwidth min 25 (in percentage)  
VFSM(config_ufp_vport)# qos bandwidth max 100 (in percentage)  
VFSM(config_ufp_vport)# enable  
VFSM(config_ufp_vport)# exit
```

8. Configure PVID/Native VLAN for external port 1.

```
VFSM(config)# interface port EXT1  
VFSM(config-if)# switchport mode access  
VFSM(config-if)# switchport access vlan 100
```

## Example 2: Trunk Mode

Following is an example configuration of UFP vPorts in trunk mode.

1. Turn on UFP.

```
VFSM(config)# ufp enable
```

2. Configure internal port 1 as UFP.

```
VFSM(config)# ufp port INT1 enable  
Warning: "Tagging/Trunk-mode" is enabled on UFP port INTA1
```

3. Configure virtual port.

```
VFSM(config)# ufp port INT1 vport 1
```

4. Configure vPort trunk mode.

```
VFSM(config_ufp_vport)# network mode trunk
```

5. Configure vPort default VLAN.

```
VFSM(config_ufp_vport)# network default-vlan 100
```

6. Ensure tagging is disabled on vPort.

```
VFSM(config_ufp_vport)# no network default-tag
```

7. Specify QoS parameters for the vPort.

```
VFSM(config_ufp_vport)# qos bandwidth min 25 (in percentage)  
VFSM(config_ufp_vport)# qos bandwidth max 100 (in percentage)  
VFSM(config_ufp_vport)# enable  
VFSM(config_ufp_vport)# exit
```

8. Configure internal port 2 as UFP.

```
VFSM(config)# ufp port INT2 enable  
Warning: "Tagging/Trunk-mode" is enabled on UFP port INTA2
```

9. Configure virtual port.

```
VFSM(config)# ufp port INT2 vport 3
```

10. Configure vPort trunk mode.

```
VFSM(config_ufp_vport)# network mode trunk
```

11. Configure vPort default VLAN.

```
VFSM(config_uvp_vport)# network default-vlan 100
```

12. Ensure tagging is disabled on vPort.

```
VFSM(config_uvp_vport)# no network default-tag
```

13. Specify QoS parameters for the vPort.

```
VFSM(config_uvp_vport)# qos bandwidth min 25 (in percentage)
VFSM(config_uvp_vport)# qos bandwidth max 100 (in percentage)
VFSM(config_uvp_vport)# enable
VFSM(config_uvp_vport)# exit
```

14. Enable tagging/trunk mode on external port 1.

```
VFSM(config)# interface port EXT1
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk native vlan 100
VFSM(config-if)# switchport trunk allowed vlan add 200,300
VFSM(config-if)# exit
```

15. Configure VLAN 200 parameters.

```
VFSM(config)# vlan 200
VFSM(config-vlan)# vmember INT1.3
VFSM(config-vlan)# vmember INT2.3
VFSM(config-vlan)# exit
```

16. Configure VLAN 300 parameters.

```
VFSM(config)# vlan 300
VFSM(config-vlan)# vmember INT1.3
VFSM(config-vlan)# vmember INT2.3
VFSM(config-vlan)# exit
```

## Example 3: Tunnel Mode

Following is an example configuration of UFP vPorts in tunnel mode.

1. Turn on UFP.

```
VFSM(config)# ufp enable
```

2. Configure internal port as UFP.

```
VFSM(config)# ufp port INT1 enable
Warning: "Tagging/Trunk-mode" is enabled on UFP port INTA1
```

3. Configure virtual port.

```
VFSM(config)# ufp port INT1 vport 1
```

4. Configure vPort tunnel mode.

```
VFSM(config_ufp_vport)# network mode tunnel
```

5. Configure vPort default VLAN.

```
VFSM(config_ufp_vport)# network default-vlan 4000
```

6. Ensure tagging is disabled on vPort.

```
VFSM(config_ufp_vport)# no network default-tag
```

7. Specify QoS parameters for the vPort.

```
VFSM(config_ufp_vport)# qos bandwidth min 25 (in percentage)
VFSM(config_ufp_vport)# qos bandwidth max 100 (in percentage)
VFSM(config_ufp_vport)# enable
VFSM(config_ufp_vport)# exit
```

8. Configure tagging on external port 1.

```
VFSM(config)# interface port EXT1
VFSM(config-if)# tagpvid-ingress
VFSM(config-if)# no vlan dot1q tag native
VFSM(config-if)# switchport access vlan 4000
VFSM(config-if)# exit
```

## Example 4: FCoE Mode

Following is an example configuration of UFP vPorts in FCoE mode.

1. Enable CEE.

```
VFSM(config)# cee enable
```

2. Enable FIPs.

```
VFSM(config)# fcoe fips enable
```

3. Turn on UFP.

```
VFSM(config)# ufp enable
```

4. Configure internal port as UFP.

```
VFSM(config)# ufp port INT1 enable
Warning: "Tagging/Trunk-mode" is enabled on UFP port INTA1
```

5. Configure virtual port.

```
VFSM(config)# ufp port INT1 vport 2
```

6. Configure vPort FCoE mode.

```
VFSM(config_ufp_vport)# network mode fcoe
```

7. Configure vPort default VLAN.

```
VFSM(config_ufp_vport)# network default-vlan 1102
```

8. Ensure tagging is disabled on vPort.

```
VFSM(config_ufp_vport)# no network default-tag
```

9. Specify QoS parameters for the vPort.

```
VFSM(config_ufp_vport)# qos bandwidth min 25 (in percentage)
VFSM(config_ufp_vport)# qos bandwidth max 100 (in percentage)
VFSM(config_ufp_vport)# enable
VFSM(config_ufp_vport)# exit
```

10. Enable tagging/trunk mode on external port.

```
VFSM(config)# interface port EXT1
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk native vlan 1102
VFSM(config-if)# tagpvid-ingress
VFSM(config-if)# exit
```

# Part 5: IP Routing

This section discusses Layer 3 switching functions. In addition to switching traffic at near line rates, the application switch can perform multi-protocol routing. This section discusses basic routing and advanced routing protocols:

- Basic Routing
- IPv6 Host Management
- Routing Information Protocol (RIP)
- Internet Group Management Protocol (IGMP)
- Border Gateway Protocol (BGP)
- Open Shortest Path First (OSPF)



---

## Chapter 19. Basic IP Routing

This chapter provides configuration background and examples for using the Virtual Fabric Switch Module (VFSM) to perform IP routing functions. The following topics are addressed in this chapter:

- [“IP Routing Benefits” on page 318](#)
- [“Routing Between IP Subnets” on page 319](#)
- [“Subnet Routing Example” on page 321](#)
- [“Dynamic Host Configuration Protocol” on page 325](#)

---

## IP Routing Benefits

The VFSM uses a combination of configurable IP switch interfaces and IP routing options. The switch IP routing capabilities provide the following benefits:

- Connects the server IP subnets to the rest of the backbone network.
- Provides the ability to route IP traffic between multiple Virtual Local Area Networks (VLANs) configured on the switch.

---

## Routing Between IP Subnets

The physical layout of most corporate networks has evolved over time. Classic hub/router topologies have given way to faster switched topologies, particularly now that switches are increasingly intelligent. The VFSM is intelligent and fast enough to perform routing functions on par with wire-speed Layer 2 switching.

The combination of faster routing and switching in a single device provides another service—it allows you to build versatile topologies that account for legacy configurations.

For example, consider a corporate campus that has migrated from a router-centric topology to a faster, more powerful, switch-based topology. As is often the case, the legacy of network growth and redesign has left the system with a mix of illogically distributed subnets.

This is a situation that switching alone cannot cure. Instead, the router is flooded with cross-subnet communication. This compromises efficiency in two ways:

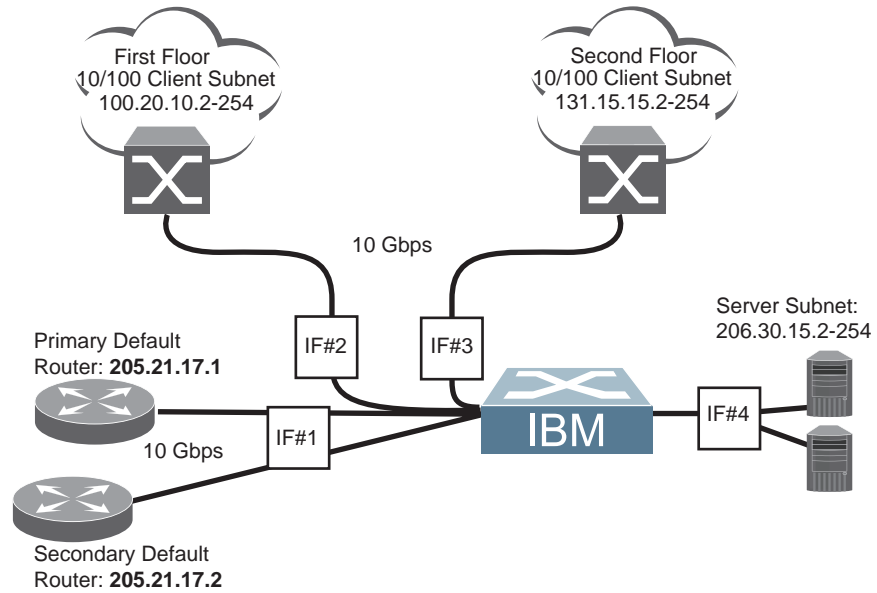
- Routers can be slower than switches. The cross-subnet side trip from the switch to the router and back again adds two hops for the data, slowing throughput considerably.
- Traffic to the router increases, increasing congestion.

Even if every end-station could be moved to better logical subnets (a daunting task), competition for access to common server pools on different subnets still burdens the routers.

This problem is solved by using VFSMs with built-in IP routing capabilities. Cross-subnet LAN traffic can now be routed within the switches with wire speed Layer 2 switching performance. This not only eases the load on the router but saves the network administrators from reconfiguring each and every end-station with new IP addresses.

Take a closer look at the BladeCenter's VFSM in the following configuration example:

Figure 33. Switch-Based Routing Topology



The VFSM connects the Gigabit Ethernet and Fast Ethernet trunks from various switched subnets throughout one building. Common servers are placed on another subnet attached to the switch. A primary and backup router are attached to the switch on yet another subnet.

Without Layer 3 IP routing on the switch, cross-subnet communication is relayed to the default gateway (in this case, the router) for the next level of routing intelligence. The router fills in the necessary address information and sends the data back to the switch, which then relays the packet to the proper destination subnet using Layer 2 switching.

With Layer 3 IP routing in place on the VFSM, routing between different IP subnets can be accomplished entirely within the switch. This leaves the routers free to handle inbound and outbound traffic for this group of subnets.

## Subnet Routing Example

Prior to configuring, you must be connected to the switch Command Line Interface (CLI) as the administrator.

**Note:** For details about accessing and using any of the menu commands described in this example, see the *IBM Networking OS 7.8 Command Reference*.

1. Assign an IP address (or document the existing one) for each router and client workstation.

In the example topology in [Figure 33 on page 320](#), the following IP addresses are used:

Table 28. Subnet Routing Example: IP Address Assignments

Subnet	Devices	IP Addresses
1	Primary and Secondary Default Routers	205.21.17.1 and 205.21.17.2
2	First Floor Client Workstations	100.20.10.2-254
3	Second Floor Client Workstations	131.15.15.2-254
4	Common Servers	206.30.15.2-254

2. Assign an IP interface for each subnet attached to the switch.

Since there are four IP subnets connected to the switch, four IP interfaces are needed:

Table 29. Subnet Routing Example: IP Interface Assignments

Interface	Devices	IP Interface Address
IF 1	Primary and Secondary Default Routers	205.21.17.3
IF 2	First Floor Client Workstations	100.20.10.1
IF 3	Second Floor Client Workstations	131.15.15.1
IF 4	Common Servers	206.30.15.1

IP interfaces are configured using the following commands at the CLI:

```
VFSM(config)# interface ip 1 (Select IP interface 1)
VFSM(config-ip-if)# ip address 205.21.17.3 255.255.255.0 enable
VFSM(config-vlan)# exit
VFSM(config)# interface ip 2 (Select IP interface 2)
VFSM(config-ip-if)# ip address 100.20.10.1 255.255.255.0 enable
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 3 (Select IP interface 3)
VFSM(config-ip-if)# ip address 131.15.15.1 255.255.255.0 enable
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 4 (Select IP interface 4)
VFSM(config-ip-if)# ip address 206.30.15.1 255.255.255.0 enable
VFSM(config-ip-if)# exit
```

3. Set each server and workstation's default gateway to the appropriate switch IP interface (the one in the same subnet as the server or workstation).
4. Configure the default gateways to the routers' addresses.

Configuring the default gateways allows the switch to send outbound traffic to the routers:

```
VFSM(config)# ip gateway 1 address 205.21.17.1 enable
VFSM(config)# ip gateway 2 address 205.21.17.2 enable
```

5. Verify the configuration.

```
VFSM(config)# show interface ip
```

Examine the resulting information. If any settings are incorrect, make the appropriate changes.

## Using VLANs to Segregate Broadcast Domains

In the previous example, devices that share a common IP network are all in the same broadcast domain. If you want to limit the broadcasts on your network, you could use VLANs to create distinct broadcast domains. For example, as shown in the following procedure, you could create one VLAN for the client trunks, one for the routers, and one for the servers.

In this example, you are adding to the previous configuration.

1. Determine which switch ports and IP interfaces belong to which VLANs.

The following table adds port and VLAN information:

*Table 30. Subnet Routing Example: Optional VLAN Ports*

VLAN	Devices	IP Interface	Switch Port	VLAN #
1	First Floor Client Workstations	2	EXT1	1
	Second Floor Client Workstations	3	EXT2	1
2	Primary Default Router	1	EXT3	2
	Secondary Default Router	1	EXT4	2
3	Common Servers 1	4	INT5	3
	Common Servers 2	4	INT6	3

2. Add the switch ports to their respective VLANs.

The VLANs shown in [Table 30](#) are configured as follows:

```
VFSM(config)# vlan 1
VFSM(config-vlan)# exit
VFSM(config)# interface port ext1,ext2      (Add ports to VLAN 1)
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk allowed vlan add 1
VFSM(config-if)# exit

VFSM(config)# vlan 2
VFSM(config-vlan)# exit
VFSM(config)# interface port ext3,ext4      (Add ports to VLAN 2)
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk allowed vlan add 2
VFSM(config-if)# exit

VFSM(config)# vlan 3
VFSM(config-vlan)# exit
VFSM(config)# interface port inet5,int6     (Add ports to VLAN 3)
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk allowed vlan add 3
VFSM(config-if)# exit
```

Each time you add a port to a VLAN, you may get the following prompt:

```
Port 4 is an Untagged/Access-mode port and its current
PVID/Native-VLAN is 1.
Confirm changing PVID/Native-VLAN from 1 to 2 [y/n]?
```

Enter *y* to set the default Port VLAN ID (PVID)/Native VLAN for the port.

3. Add each IP interface to the appropriate VLAN.

Now that the ports are separated into three VLANs, the IP interface for each subnet must be placed in the appropriate VLAN. From [Table 30 on page 322](#), the settings are made as follows:

```
VFSM(config)# interface ip 1                (Select IP interface 1)
VFSM(config-ip-if)# vlan 2                  (Add VLAN 2)
VFSM(config-vlan)# exit
VFSM(config)# interface ip 2                (Select IP interface 2)
VFSM(config-ip-if)# vlan 1                  (Add VLAN 1)
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 3                (Select IP interface 3)
VFSM(config-ip-if)# vlan 1                  (Add VLAN 1)
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 4                (Select IP interface 4)
VFSM(config-ip-if)# vlan 3                  (Add VLAN 3)
VFSM(config-ip-if)# exit
```

4. Apply and verify the configuration.

```
VFSM(config)# show vlan
VFSM(config)# show interface information
VFSM(config)# show interface ip
```

Examine the resulting information. If any settings are incorrect, make the appropriate changes.

---

## BOOTP Relay Agent

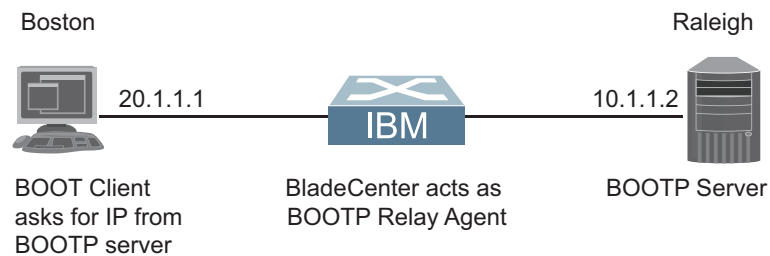
### BOOTP Relay Agent Overview

The VFSM can function as a Bootstrap Protocol relay agent, enabling the switch to forward a client request for an IP address up to two BOOTP servers with IP addresses that have been configured on the switch.

When a switch receives a BOOTP request from a BOOTP client requesting an IP address, the switch acts as a proxy for the client. The request is then forwarded as a UDP Unicast MAC layer message to two BOOTP servers whose IP addresses are configured on the switch. The servers respond to the switch with a Unicast reply that contains the default gateway and IP address for the client. The switch then forwards this reply back to the client.

Figure 34 shows a basic BOOTP network example.

Figure 34. BOOTP Relay Agent Configuration



The use of two servers provide failover redundancy. The client request is forwarded to both BOOTP servers configured on the switch. However, no health checking is supported.

### BOOTP Relay Agent Configuration

To enable the VFSM to be the BOOTP forwarder, you need to configure the BOOTP server IP addresses on the switch, and enable BOOTP relay on the interface(s) on which the BOOTP requests are received.

Generally, you should configure the command on the switch IP interface that is closest to the client, so that the BOOTP server knows from which IP subnet the newly allocated IP address should come.

Use the following commands to configure the switch as a BOOTP relay agent:

```
VFSM(config)# ip bootp-relay enable
VFSM(config)# ip bootp-relay server <1-5> address <IPv4 address>
```

Use the following command to enable the Relay functionality on an IP interface:

```
VFSM(config)# interface ip <interface number>
VFSM(config-ip-if)# relay
VFSM(config-ip-if)# exit
```

---

## Dynamic Host Configuration Protocol

Dynamic Host Configuration Protocol (DHCP) is a transport protocol that provides a framework for automatically assigning IP addresses and configuration information to other IP hosts or clients in a large TCP/IP network. Without DHCP, the IP address must be entered manually for each network device. DHCP allows a network administrator to distribute IP addresses from a central point and automatically send a new IP address when a device is connected to a different place in the network.

DHCP is an extension of another network IP management protocol, Bootstrap Protocol (BOOTP), with an additional capability of being able to dynamically allocate reusable network addresses and configuration parameters for client operation.

Built on the client/server model, DHCP allows hosts or clients on an IP network to obtain their configurations from a DHCP server, thereby reducing network administration. The most significant configuration the client receives from the server is its required IP address; (other optional parameters include the “generic” file name to be booted, the address of the default gateway, and so forth).

DHCP relay agent eliminates the need to have DHCP/BOOTP servers on every subnet. It allows the administrator to reduce the number of DHCP servers deployed on the network and to centralize them. Without the DHCP relay agent, there must be at least one DHCP server deployed at each subnet that has hosts needing to perform the DHCP request.

## DHCP Relay Agent

DHCP is described in RFC 2131, and the DHCP relay agent supported on VFSMs is described in RFC 1542. DHCP uses UDP as its transport protocol. The client sends messages to the server on port 67 and the server sends messages to the client on port 68.

DHCP defines the methods through which clients can be assigned an IP address for a finite lease period and allowing reassignment of the IP address to another client later. Additionally, DHCP provides the mechanism for a client to gather other IP configuration parameters it needs to operate in the TCP/IP network.

In the DHCP environment, the VFSM acts as a relay agent. The DHCP relay feature enables the switch to forward a client request for an IP address to two BOOTP servers with IP addresses that have been configured on the switch.

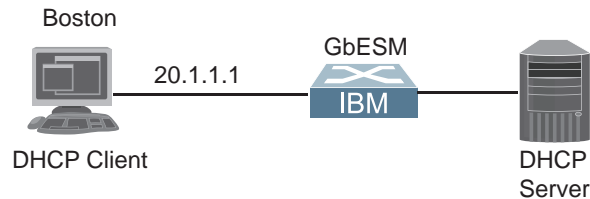
When a switch receives a UDP broadcast on port 67 from a DHCP client requesting an IP address, the switch acts as a proxy for the client, replacing the client source IP (SIP) and destination IP (DIP) addresses. The request is then forwarded as a UDP Unicast MAC layer message to two BOOTP servers whose IP addresses are configured on the switch. The servers respond as a UDP Unicast message back to the switch, with the default gateway and IP address for the client. The destination IP address in the server response represents the interface address on the switch that received the client request. This interface address tells the switch on which VLAN to send the server response to the client.

## DHCP Relay Agent Configuration

To enable the VFSM to be the BOOTP forwarder, you need to configure the DHCP/BOOTP server IP addresses on the switch. Generally, you should configure the switch IP interface on the client side to match the client's subnet, and configure VLANs to separate client and server subnets. The DHCP server knows from which IP subnet the newly allocated IP address should come.

The following figure shows a basic DHCP network example:

Figure 35. DHCP Relay Agent Configuration



In VFSM implementation, there is no need for primary or secondary servers. The client request is forwarded to the BOOTP servers configured on the switch. The use of two servers provide failover redundancy. However, no health checking is supported.

Use the following commands to configure the switch as a DHCP relay agent:

```
VFSM(config)# ip bootp-relay server 1 <IP address>
VFSM(config)# ip bootp-relay server 2 <IP address>
VFSM(config)# ip bootp-relay server 3 <IP address>
VFSM(config)# ip bootp-relay server 4 <IP address>
VFSM(config)# ip bootp-relay server 5 <IP address>
VFSM(config)# ip bootp-relay enable
VFSM(config)# show ip bootp-relay
```

Additionally, DHCP Relay functionality can be assigned on a per interface basis. Use the following command to enable the Relay functionality:

```
VFSM(config)# interface ip <Interface number>
VFSM(config-ip-if)# relay
```



---

## Chapter 20. Internet Protocol Version 6

Internet Protocol version 6 (IPv6) is a network layer protocol intended to expand the network address space. IPv6 is a robust and expandable protocol that meets the need for increased physical address space. The switch supports the following RFCs for IPv6-related features:

- RFC 1981
- RFC 2404
- RFC 2410
- RFC 2451
- RFC 2460
- RFC 2474
- RFC 2526
- RFC 2711
- RFC 2740
- RFC 3289
- RFC 3306
- RFC 3307
- RFC 3411
- RFC 3412
- RFC 3413
- RFC 3414
- RFC 3484
- RFC 3602
- RFC 3810
- RFC 3879
- RFC 4007
- RFC 4213
- RFC 4291
- RFC 4292
- RFC 4293
- RFC 4301
- RFC 4302
- RFC 4303
- RFC 4306
- RFC 4307
- RFC 4443
- RFC 4552
- RFC 4718
- RFC 4835
- RFC 4861
- RFC 4862
- RFC 5095
- RFC 5114
- RFC 5340

This chapter describes the basic configuration of IPv6 addresses and how to manage the switch via IPv6 host management.

---

## IPv6 Limitations

The following IPv6 features are not supported in this release.

- Dynamic Host Control Protocol for IPv6 (DHCPv6)
- Border Gateway Protocol for IPv6 (BGP)
- Routing Information Protocol for IPv6 (RIPng)

Most other IBM Networking OS 7.8 features permit IP addresses to be configured using either IPv4 or IPv6 address formats. However, the following switch features support IPv4 only:

- Default switch management IP address
- Bootstrap Protocol (BOOTP) and DHCP
- RADIUS, TACACS+ and LDAP
- QoS metering and re-marking ACLs for out-profile traffic
- Stacking
- VMware Virtual Center (vCenter) for VMready
- Routing Information Protocol (RIP)
- Internet Group Management Protocol (IGMP)
- Border Gateway Protocol (BGP)
- Virtual Router Redundancy Protocol (VRRP)
- sFLOW

---

## IPv6 Address Format

The IPv6 address is 128 bits (16 bytes) long and is represented as a sequence of eight 16-bit hex values, separated by colons.

Each IPv6 address has two parts:

- Subnet prefix representing the network to which the interface is connected
- Local identifier, either derived from the MAC address or user-configured

The preferred hexadecimal format is as follows:

XXXX : XXXX : XXXX : XXXX : XXXX : XXXX : XXXX : XXXX

Example IPv6 address:

FEDC : BA98 : 7654 : BA98 : FEDC : 1234 : ABCD : 5412

Some addresses can contain long sequences of zeros. A single contiguous sequence of zeros can be compressed to :: (two colons). For example, consider the following IPv6 address:

FE80 : 0 : 0 : 0 : 2AA : FF : FA : 4CA2

The address can be compressed as follows:

FE80 : : 2AA : FF : FA : 4CA2

Unlike IPv4, a subnet mask is not used for IPv6 addresses. IPv6 uses the subnet prefix as the network identifier. The prefix is the part of the address that indicates the bits that have fixed values or are the bits of the subnet prefix. An IPv6 prefix is written in address/prefix-length notation. For example, in the following address, 64 is the network prefix:

21DA : D300 : 0000 : 2F3C : : /64

IPv6 addresses can be either user-configured or automatically configured. Automatically configured addresses always have a 64-bit subnet prefix and a 64-bit interface identifier. In most implementations, the interface identifier is derived from the switch's MAC address, using a method called EUI-64.

Most IBM N/OS 7.8 features permit IP addresses to be configured using either IPv4 or IPv6 address formats. Throughout this manual, *IP address* is used in places where either an IPv4 or IPv6 address is allowed. In places where only one type of address is allowed, the type (*IPv4* or *IPv6*) is specified.

---

## IPv6 Address Types

IPv6 supports three types of addresses: unicast (one-to-one), multicast (one-to-many), and anycast (one-to-nearest). Multicast addresses replace the use of broadcast addresses.

### Unicast Address

Unicast is a communication between a single host and a single receiver. Packets sent to a unicast address are delivered to the interface identified by that address. IPv6 defines the following types of unicast address:

- **Global Unicast address:** An address that can be reached and identified globally. Global Unicast addresses use the high-order bit range up to FF00, therefore all non-multicast and non-link-local addresses are considered to be global unicast. A manually configured IPv6 address must be fully specified. Autoconfigured IPv6 addresses are comprised of a prefix combined with the 64-bit EUI. RFC 4291 defines the IPv6 addressing architecture.

The interface ID must be unique within the same subnet.

- **Link-local unicast address:** An address used to communicate with a neighbor on the same link. Link-local addresses use the format FE80 : : EUI

Link-local addresses are designed to be used for addressing on a single link for purposes such as automatic address configuration, neighbor discovery, or when no routers are present.

Routers must not forward any packets with link-local source or destination addresses to other links.

### Multicast

Multicast is communication between a single host and multiple receivers. Packets are sent to all interfaces identified by that address. An interface may belong to any number of multicast groups.

A multicast address (FF00 - FFFF) is an identifier for a group interface. The multicast address most often encountered is a solicited-node multicast address using prefix FF02 : : 1 : FF00 : 0000 / 104 with the low-order 24 bits of the unicast or anycast address.

The following well-known multicast addresses are pre-defined. The group IDs defined in this section are defined for explicit scope values, as follows:

FF00 : : : : : 0 through FFOF : : : : : 0

### Anycast

Packets sent to an anycast address or list of addresses are delivered to the nearest interface identified by that address. Anycast is a communication between a single sender and a list of addresses.

Anycast addresses are allocated from the unicast address space, using any of the defined unicast address formats. Thus, anycast addresses are syntactically indistinguishable from unicast addresses. When a unicast address is assigned to more than one interface, thus turning it into an anycast address, the nodes to which the address is assigned must be explicitly configured to know that it is an anycast address.

---

## IPv6 Address Autoconfiguration

IPv6 supports the following types of address autoconfiguration:

- **Stateful address configuration**

Address configuration is based on the use of a stateful address configuration protocol, such as DHCPv6, to obtain addresses and other configuration options.

- **Stateless address configuration**

Address configuration is based on the receipt of Router Advertisement messages that contain one or more Prefix Information options.

N/OS 7.8 supports stateless address configuration.

Stateless address configuration allows hosts on a link to configure themselves with link-local addresses and with addresses derived from prefixes advertised by local routers. Even if no router is present, hosts on the same link can configure themselves with link-local addresses and communicate without manual configuration.

---

## IPv6 Interfaces

Each IPv6 interface supports multiple IPv6 addresses. You can manually configure up to two IPv6 addresses for each interface, or you can allow the switch to use stateless autoconfiguration. By default, the switch automatically configures the IPv6 address of its management interface.

You can manually configure two IPv6 addresses for each interface, as follows:

- Initial IPv6 address is a global unicast or anycast address.

```
VFSM(config)# interface ip <interface number>
VFSM(config-ip-if)# ipv6 address <IPv6 address>
```

Note that you cannot configure both addresses as anycast. If you configure an anycast address on the interface you must also configure a global unicast address on that interface.

- Second IPv6 address can be a unicast or anycast address.

```
VFSM(config-ip-if)# ipv6 secaddr6 <IPv6 address>
VFSM(config-ip-if)# exit
```

You cannot configure an IPv4 address on an IPv6 management interface. Each interface can be configured with only one address type: either IPv4 or IPv6, but not both. When changing between IPv4 and IPv6 address formats, the prior address settings for the interface are discarded.

Each IPv6 interface can belong to only one VLAN. Each VLAN can support only one IPv6 interface. Each VLAN can support multiple IPv4 interfaces.

Interface 127 is reserved for IPv6 host support. This interface is included in management VLAN 4095. Use the following commands to configure the IPv6 gateway:

```
VFSM(config)# ip gateway6 1 address <IPv6 address>
VFSM(config)# ip gateway6 1 enable
```

IPv6 gateway 1 is reserved for IPv6 data interfaces. IPv6 gateway 132 is the default IPv6 management gateway.

---

## Neighbor Discovery

### Neighbor Discovery Overview

The switch uses Neighbor Discovery protocol (ND) to gather information about other router and host nodes, including the IPv6 addresses. Host nodes use ND to configure their interfaces and perform health detection. ND allows each node to determine the link-layer addresses of neighboring nodes, and to keep track of each neighbor's information. A neighboring node is a host or a router that is linked directly to the switch. The switch supports Neighbor Discovery as described in RFC 4861.

Neighbor Discover messages allow network nodes to exchange information, as follows:

- *Neighbor Solicitations* allow a node to discover information about other nodes.
- *Neighbor Advertisements* are sent in response to Neighbor Solicitations. The Neighbor Advertisement contains information required by nodes to determine the link-layer address of the sender, and the sender's role on the network.
- IPv6 hosts use *Router Solicitations* to discover IPv6 routers. When a router receives a Router Solicitation, it responds immediately to the host.
- Routers use *Router Advertisements* to announce its presence on the network, and to provide its address prefix to neighbor devices. IPv6 hosts listen for Router Advertisements, and use the information to build a list of default routers. Each host uses this information to perform autoconfiguration of IPv6 addresses.
- *Redirect messages* are sent by IPv6 routers to inform hosts of a better first-hop address for a specific destination. Redirect messages are only sent by routers for unicast traffic, are only unicast to originating hosts, and are only processed by hosts.

ND configuration for various advertisements, flags, and interval settings is performed on a per-interface basis using the following command path:

```
VFSM(config)# interface ip <interface number>
VFSM(config-ip-if)# [no] ipv6 nd ?
VFSM(config-ip-if)# exit
```

To add or remove entries in the static neighbor cache, use the following command path:

```
VFSM(config)# [no] ip neighbors ?
```

## Host vs. Router

Each IPv6 interface can be configured as a router node or a host node, as follows:

- A router node's IP address is configured manually. Router nodes can send Router Advertisements.
- A host node's IP address is autoconfigured. Host nodes listen for Router Advertisements that convey information about devices on the network.

**Note:** When IP forwarding is turned on, all IPv6 interfaces configured on the switch can forward packets.

You can configure each IPv6 interface as either a host node or a router node. You can manually assign an IPv6 address to an interface in host mode, or the interface can be assigned an IPv6 address by an upstream router, using information from router advertisements to perform stateless auto-configuration.

To set an interface to host mode, use the following command:

```
VFSM(config)# interface ip <interface number>
VFSM(config-ip-if)# ipv6host
VFSM(config-ip-if)# exit
```

By default, host mode is enabled on the management interface, and disabled on data interfaces.

The VFSM supports up to 1880 IPv6 routes.

---

## Supported Applications

The following applications have been enhanced to provide IPv6 support.

- **Ping**

The `ping` command supports IPv6 addresses. Use the following format to ping an IPv6 address:

```
ping <host name> | <IPv6 address> [-n <tries (0-4294967295)>]
[-w <msec delay (0-4294967295)>] [-l <length (0/32-65500/2080)>]
[-s <IP source>] [-v <TOS (0-255)>] [-f] [-t]
```

To ping a link-local address (begins with FE80), provide an interface index, as follows:

```
ping <IPv6 address>%<Interface index> [-n <tries (0-4294967295)>]
[-w <msec delay (0-4294967295)>] [-l <length (0/32-65500/2080)>]
[-s <IP source>] [-v <TOS (0-255)>] [-f] [-t]
```

- **Traceroute**

The `traceroute` command supports IPv6 addresses (but not link-local addresses).

Use the following format to perform a traceroute to an IPv6 address:

```
traceroute <host name> | <IPv6 address> [<max-hops (1-32)>]
[<msec delay (1-4294967295)>]]
```

- **Telnet server**

The `telnet` command supports IPv6 addresses (but not link-local addresses).

Use the following format to Telnet into an IPv6 interface on the switch:

```
telnet <host name> | <IPv6 address> [<port>]
```

- **Telnet client**

The `telnet` command supports IPv6 addresses (but not link-local addresses).

Use the following format to Telnet to an IPv6 address:

```
telnet <host name> | <IPv6 address> [<port>]
```

- **HTTP/HTTPS**

The HTTP/HTTPS servers support both IPv4 and IPv6 connections.

- **SSH**

Secure Shell (SSH) connections over IPv6 are supported, (but not link-local addresses). The following syntax is required from the client:

```
ssh -u <IPv6 address>
```

Example:

```
ssh -u 2001:2:3:4:0:0:0:142
```

- **TFTP**

The TFTP commands support both IPv4 and IPv6 addresses. Link-local addresses are not supported.

- **FTP**

The FTP commands support both IPv4 and IPv6 addresses. Link-local addresses are not supported.

- **DNS client**

DNS commands support both IPv4 and IPv6 addresses. Link-local addresses are not supported. Use the following command to specify the type of DNS query to be sent first:

```
VFSM(config)# ip dns ipv6 request-version {ipv4|ipv6}
```

If you set the request version to `v4`, the DNS application sends an `A` query first, to resolve the hostname with an IPv4 address. If no `A` record is found for that hostname (no IPv4 address for that hostname) an `AAAA` query is sent to resolve the hostname with a IPv6 address.

If you set the request version to `v6`, the DNS application sends an `AAAA` query first, to resolve the hostname with an IPv6 address. If no `AAAA` record is found for that hostname (no IPv6 address for that hostname) an `A` query is sent to resolve the hostname with an IPv4 address.

---

## Configuration Guidelines

When you configure an interface for IPv6, consider the following guidelines:

- Support for subnet router anycast addresses is not available.
- Interface 127 is reserved for IPv6 management.
- A single interface can accept either IPv4 or IPv6 addresses, but not both IPv4 and IPv6 addresses.
- A single interface can accept multiple IPv6 addresses.
- A single interface can accept only one IPv4 address.
- If you change the IPv6 address of a configured interface to an IPv4 address, all IPv6 settings are deleted.
- A single VLAN can support only one IPv6 interface.
- Health checks are not supported for IPv6 gateways.
- IPv6 interfaces support Path MTU Discovery. The CPU's MTU is fixed at 1500 bytes.
- Support for jumbo frames (1,500 to 9,216 byte MTUs) is limited. Any jumbo frames intended for the CPU must be fragmented by the remote node. The switch can re-assemble fragmented packets up to 9k. It can also fragment and transmit jumbo packets received from higher layers.

---

## IPv6 Configuration Examples

This section provides steps to configure IPv6 on the switch.

### IPv6 Example 1

The following example uses IPv6 host mode to autoconfigure an IPv6 address for the interface. By default, the interface is assigned to VLAN 1.

1. Enable IPv6 host mode on an interface.

```
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip6host
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# exit
```

2. Configure the IPv6 default gateway.

```
VFSM(config)# ip gateway6 1 address 2001:BA98:7654:BA98:FEDC:1234:ABCD:5412
VFSM(config)# ip gateway6 1 enable
```

3. Verify the interface address.

```
VFSM(config)# show interface ip 2 (Display interface information)
```

### IPv6 Example 2

Use the following example to manually configure IPv6 on an interface.

1. Assign an IPv6 address and prefix length to the interface.

```
VFSM(config)# interface ip 3
VFSM(config-ip-if)# ipv6 address 2001:BA98:7654:BA98:FEDC:1234:ABCD:5214
VFSM(config-ip-if)# ipv6 prefixlen 64
VFSM(config-ip-if)# ipv6 seccaddr6 2003::1 32
VFSM(config-ip-if)# vlan 2
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# exit
```

The secondary IPv6 address is compressed, and the prefix length is 32.

2. Configure the IPv6 default gateway.

```
VFSM(config)# ip gateway6 1 address 2001:BA98:7654:BA98:FEDC:1234:ABCD:5412
VFSM(config)# ip gateway6 1 enable (Enable default gateway)
```

3. Configure Neighbor Discovery advertisements for the interface (optional)

```
VFSM(config)# interface ip 3
VFSM(config-ip-if)# no ipv6 nd suppress-ra
```

4. Verify the configuration.

```
VFSM(config-ip-if)# show layer3
```

---

## Chapter 21. Using IPsec with IPv6

Internet Protocol Security (IPsec) is a protocol suite for securing Internet Protocol (IP) communications by authenticating and encrypting each IP packet of a communication session. IPsec also includes protocols for establishing mutual authentication between agents at the beginning of the session and negotiation of cryptographic keys to be used during the session.

Since IPsec was implemented in conjunction with IPv6, all implementations of IPv6 must contain IPsec. To support the National Institute of Standards and Technology (NIST) recommendations for IPv6 implementations, IBM Networking OS IPv6 feature compliance has been extended to include the following IETF RFCs, with an emphasis on IP Security (IPsec) and Internet Key Exchange version 2, and authentication/confidentiality for OSPFv3:

- RFC 4301 for IPv6 security
- RFC 4302 for the IPv6 Authentication Header
- RFCs 2404, 2410, 2451, 3602, and 4303 for IPv6 Encapsulating Security Payload (ESP), including NULL encryption, CBC-mode 3DES and AES ciphers, and HMAC-SHA-1-96.
- RFCs 4306, 4307, 4718, and 4835 for IKEv2 and cryptography
- RFC 4552 for OSPFv3 IPv6 authentication
- RFC 5114 for Diffie-Hellman groups

**Note:** This implementation of IPsec supports DH groups 1, 2, 5, 14, and 24.

The following topics are discussed in this chapter:

- [“IPsec Protocols” on page 342](#)
- [“Using IPsec with the VFSM” on page 343](#)

---

## IPsec Protocols

The IBM N/OS implementation of IPsec supports the following protocols:

- **Authentication Header (AH)**  
AHs provide connectionless integrity and data origin authentication for IP packets, and provide protection against replay attacks. In IPv6, the AH protects the AH itself, the Destination Options extension header after the AH, and the IP payload. It also protects the fixed IPv6 header and all extension headers before the AH, except for the mutable fields DSCP, ECN, Flow Label, and Hop Limit. AH is defined in RFC 4302.
- **Encapsulating Security Payload (ESP)**  
ESPs provide confidentiality, data origin authentication, integrity, an anti-replay service (a form of partial sequence integrity), and some traffic flow confidentiality. ESPs may be applied alone or in combination with an AH. ESP is defined in RFC 4303.
- **Internet Key Exchange Version 2 (IKEv2)**  
IKEv2 is used for mutual authentication between two network elements. An IKE establishes a security association (SA) that includes shared secret information to efficiently establish SAs for ESPs and AHs, and a set of cryptographic algorithms to be used by the SAs to protect the associated traffic. IKEv2 is defined in RFC 4306.

Using IKEv2 as the foundation, IPsec supports ESP for encryption and/or authentication, and/or AH for authentication of the remote partner.

Both ESP and AH rely on security associations. A security association (SA) is the bundle of algorithms and parameters (such as keys) that encrypt and authenticate a particular flow in one direction.

---

## Using IPsec with the VFSM

IPsec supports the fragmentation and reassembly of IP packets that occurs when data goes to and comes from an external device. The IBM Virtual Fabric 10Gb Switch Module acts as an end node that processes any fragmentation and reassembly of packets but does not forward the IPsec traffic. The IKEv2 key must be authenticated before you can use IPsec.

The security protocol for the session key is either ESP or AH. Outgoing packets are labeled with the SA SPI (Security Parameter Index), which the remote device will use in its verification and decryption process.

Every outgoing IPv6 packet is checked against the IPsec policies in force. For each outbound packet, after the packet is encrypted, the software compares the packet size with the MTU size that it either obtains from the default minimum maximum transmission unit (MTU) size (1500) or from path MTU discovery. If the packet size is larger than the MTU size, the receiver drops the packet and sends a message containing the MTU size to the sender. The sender then fragments the packet into smaller pieces and retransmits them using the correct MTU size.

The maximum traffic load for each IPsec packet is limited to the following:

- IKEv2 SAs: 5
- IPsec SAs: 10 (5 SAs in each direction)
- SPDs: 20 (10 policies in each direction)

IPsec is implemented as a software cryptography engine designed for handling control traffic, such as network management. IPsec is not designed for handling data traffic, such as a VPN.

## Setting up Authentication

Before you can use IPsec, you need to have key policy authentication in place. There are two types of key policy authentication:

- **Preshared key (default)**  
The parties agree on a shared, secret key that is used for authentication in an IPsec policy. During security negotiation, information is encrypted before transmission by using a session key created by using a Diffie-Hellman calculation and the shared, secret key. Information is decrypted on the receiving end using the same key. One IPsec peer authenticates the other peer's packet by decryption and verification of the hash inside the packet (the hash inside the packet is a hash of the preshared key). If authentication fails, the packet is discarded.
- **Digital certificate (using RSA algorithms)**  
The peer being validated must hold a digital certificate signed by a trusted Certificate Authority and the private key for that digital certificate. The side performing the authentication only needs a copy of the trusted certificate authorities digital certificate. During IKEv2 authentication, the side being validated sends a copy of the digital certificate and a hash value signed using the private key. The certificate can be either generated or imported.

**Note:** During the IKEv2 negotiation phase, the digital certificate takes precedence over the preshared key.

## Creating an IKEv2 Proposal

With IKEv2, a single policy can have multiple encryption and authentication types, as well as multiple integrity algorithms.

To create an IKEv2 proposal:

1. Enter IKEv2 proposal mode.

```
VFSM(config)# ikev2 proposal
```

2. Set the DES encryption algorithm.

```
VFSM(config-ikev2-prop)# encryption 3des|aes-cbc|des (default: 3des)
```

3. Set the authentication integrity algorithm type.

```
VFSM(config-ikev2-prop)# integrity md5|sha1 (default: sha1)
```

4. Set the Diffie-Hellman group.

```
VFSM(config-ikev2-prop)# group 1|2|5|14|24 (default: 2)
```

## Importing an IKEv2 Digital Certificate

To import an IKEv2 digital certificate for authentication:

1. Import the CA certificate file.

```
VFSM(config)# copy tftp ca-cert address <hostname or IPv4 address>
Enter hostname or IP address of FTP/TFTP server: <hostname or IPv4 address>
Enter name of file on FTP/TFTP server: <path and filename of CA certificate file>
Enter the port to use for down the file
["mgt"|"data"]:
Confirm download operation [y/n]: y
```

2. Import the host key file.

```
VFSM(config)# copy tftp host-key address <hostname or IPv4 address>
Enter name of file on FTP/TFTP server: <path and filename of host private key file>
Enter the port to use for down the file
["mgt"|"data"]:
Confirm download operation [y/n]: y
```

3. Import the host certificate file.

```
VFSM(config)# copy tftp host-cert address <hostname or IPv4 address>
Enter name of file on FTP/TFTP server: <path and filename of host certificate file>
Enter the port to use for down the file
["mgt"|"data"]:
Confirm download operation [y/n]: y
```

**Note:** When prompted for the port to use for download the file, if you used a management port to connect the switch to the server, enter `mgt`, otherwise enter `data`.

## Generating an IKEv2 Digital Certificate

To create an IKEv2 digital certificate for authentication:

1. Create an HTTPS certificate defining the information you want to be used in the various fields.

```
VFSM(config)# access https generate-certificate
Country Name (2 letter code) []: <country code>
State or Province Name (full name) []: <state>
Locality Name (eg, city) []: <city>
Organization Name (eg, company) []: <company>
Organizational Unit Name (eg, section) []: <org. unit>
Common Name (eg, YOUR name) []: <name>
Email (eg, email address) []: <email address>
Confirm generating certificate? [y/n]: y
Generating certificate. Please wait (approx 30 seconds)
restarting SSL agent
```

2. Save the HTTPS certificate.

The certificate is valid only until the switch is rebooted. To save the certificate so that it is retained beyond reboot or power cycles, use the following command:

```
VFSM(config)# access https save-certificate
```

3. Enable IKEv2 RSA-signature authentication:

```
VFSM(config)# access https enable
```

## Enabling IKEv2 Preshared Key Authentication

To set up IKEv2 preshared key authentication:

1. Enter the local preshared key.

```
VFSM(config)# ikev2 preshare-key local <preshared key, a string of 1-256 chars>
```

2. If asymmetric authentication is supported, enter the remote key:

```
VFSM(config)# ikev2 preshare-key remote <preshared key> <IPv6 host>
```

where the following parameters are used:

- *preshared key* A string of 1-256 characters
- *IPv6 host* An IPv6-format host, such as “3000::1”

3. Set up the IKEv2 identification type by entering *one* of the following commands:

```
VFSM(config)# ikev2 identity local address (use an IPv6 address)
VFSM(config)# ikev2 identity local email <email address>
VFSM(config)# ikev2 identity local fqdn <domain name>
```

To disable IKEv2 RSA-signature authentication method and enable preshared key authentication, enter:

```
VFSM(config)# no access https
```

## Setting Up a Key Policy

When configuring IPsec, you must define a key policy. This key policy can be either manual or dynamic. Either way, configuring a policy involves the following steps:

- Create a transform set—This defines which encryption and authentication algorithms are used.
  - Create a traffic selector—This describes the packets to which the policy applies.
  - Establish an IPsec policy.
  - Apply the policy.
1. To define which encryption and authentication algorithms are used, create a transform set:

```
VFSM(config)# ipsec transform-set <transform ID> <encryption method> <integrity algorithm> <AH authentication algorithm>
```

where the following parameters are used:

- *transform ID* A number from 1-10
- *encryption method* One of the following: esp-des | esp-3des | esp-aes-cbc | esp-null
- *integrity algorithm* One of the following: esp-sha1 | esp-md5 | none

– *AH authentication algorithm* One of the following: ah-sha1 | ah-md5 | none

2. Decide whether to use tunnel or transport mode. The default mode is transport.

```
VFSM(config)# ipsec transform-set tunnel|transport
```

3. To describe the packets to which this policy applies, create a traffic selector using the following commands:

```
VFSM(config)# ipsec traffic-selector <traffic selector number> permit|deny any|icmp  
<type/any> |tcp > <source IP address/any> <destination IP address/any> [<prefix length>]
```

where the following parameters are used:

- *traffic selector number* an integer from 1-10
- permit|deny whether or not to permit IPsec encryption of traffic that meets the criteria specified in this command
- proto/any apply the selector to any type of traffic
- proto/icmp type|any only apply the selector only to ICMP traffic of the specified *type* (an integer from 1-255) or to any ICMP traffic
- proto/tcp only apply the selector to TCP traffic
- source IP address|any the source IP address in IPv6 format or “any” source
- destination IP address|any the destination IP address in IPv6 format or “any” destination
- prefix length (Optional) the length of the destination IPv6 prefix; an integer from 1-128

Permitted traffic that matches the policy in force is encrypted, while denied traffic that matches the policy in force is dropped. Traffic that does not match the policy bypasses IPsec and passes through *clear* (unencrypted).

4. Choose whether to use a manual or a dynamic policy.

## Using a Manual Key Policy

A manual policy involves configuring policy and manual SA entries for local and remote peers.

To configure a manual key policy, you need:

- The IP address of the peer in IPv6 format (for example, “3000::1”).
- Inbound/Outbound session keys for the security protocols.

You can then assign the policy to an interface. The peer represents the other end of the security association. The security protocol for the session key can be either ESP or AH.

To create and configure a manual policy:

1. Enter a manual policy to configure.

```
VFSM(config)# ipsec manual-policy <policy number>
```

2. Configure the policy.

```
VFSM(config-ipsec-manual)#peer <peer's IPv6 address>
VFSM(config-ipsec-manual)#traffic-selector <IPsec traffic selector>
VFSM(config-ipsec-manual)#transform-set <IPsec transform set>
VFSM(config-ipsec-manual)#in-ah auth-key <inbound AH IPsec key>
VFSM(config-ipsec-manual)#in-ah auth-spi <inbound AH IPsec SPI>
VFSM(config-ipsec-manual)#in-esp cipher-key <inbound ESP cipher key>
VFSM(config-ipsec-manual)#in-esp auth-spi <inbound ESP SPI>
VFSM(config-ipsec-manual)#in-esp auth-key <inbound ESP authenticator key>
VFSM(config-ipsec-manual)#out-ah auth-key <outbound AH IPsec key>
VFSM(config-ipsec-manual)#out-ah auth-spi <outbound AH IPsec SPI>
VFSM(config-ipsec-manual)#out-esp cipher-key <outbound ESP cipher key>
VFSM(config-ipsec-manual)#out-esp auth-spi <outbound ESP SPI>
VFSM(config-ipsec-manual)#out-esp auth-key <outbound ESP authenticator key>
```

where the following parameters are used:

- |  |  |
|--|--|
| – <i>peer's IPv6 address</i>           | The IPv6 address of the peer (for example, 3000::1)    |
| – <i>IPsec traffic-selector</i>        | A number from 1-10                                     |
| – <i>IPsec of transform-set</i>        | A number from 1-10                                     |
| – <i>inbound AH IPsec key</i>          | The inbound AH key code, in hexadecimal                |
| – <i>inbound AH IPsec SPI</i>          | A number from 256-4294967295                           |
| – <i>inbound ESP cipher key</i>        | The inbound ESP key code, in hexadecimal               |
| – <i>inbound ESP SPI</i>               | A number from 256-4294967295                           |
| – <i>inbound ESP authenticator key</i> | The inbound ESP authenticator key code, in hexadecimal |
| – <i>outbound AH IPsec key</i>         | The outbound AH key code, in hexadecimal               |
| – <i>outbound AH IPsec SPI</i>         | A number from 256-4294967295                           |
| – <i>outbound ESP cipher key</i>       | The outbound ESP key code, in hexadecimal              |
| – <i>outbound ESP SPI</i>              | A number from 256-4294967295                           |

- *outbound ESP authenticator key* The outbound ESP authenticator key code, in hexadecimal

**Note:** When configuring a manual policy ESP, the ESP authenticator key is optional.

**Note:** If using third-party switches, the IPsec manual policy session key must be of fixed length as follows:

For AH key: SHA1 is 20 bytes; MD5 is 16 bytes

For ESP cipher key: 3DES is 24 bytes; AES-cbc is 24 bytes; DES is 8 bytes

For ESP auth key: SHA1 is 20 bytes; MD5 is 16 bytes

3. After you configure the IPsec policy, you need to apply it to the interface to enforce the security policies on that interface and save it to keep it in place after a reboot. To accomplish this, enter:

```
VFSM(config-ip)# interface ip <IP interface number, 1-128>
VFSM(config-ip-if)# address <IPv6 address>
VFSM(config-ip-if)# ipsec manual-policy <policy index, 1-10>
VFSM(config-ip-if)# enable (enable the IP interface)
VFSM(config-ip-if)# exit
VFSM(config-ip-if)# write (save the current configuration)
```

## Using a Dynamic Key Policy

When you use a dynamic key policy, the first packet triggers IKE and sets the IPsec SA and IKEv2 SA. The initial packet negotiation also determines the lifetime of the algorithm, or how long it stays in effect. When the key expires, a new key is automatically created. This helps prevent break-ins.

To configure a dynamic key policy:

1. Choose a dynamic policy to configure.

```
VFSM(config)# ipsec dynamic-policy <policy number>
```

2. Configure the policy.

```
VFSM(config-ipsec-dynamic)# peer <peer's IPv6 address>
VFSM(config-ipsec-dynamic)# traffic-selector <index of traffic selector>
VFSM(config-ipsec-dynamic)# transform-set <index of transform set>
VFSM(config-ipsec-dynamic)# sa-lifetime <SA lifetime, in seconds>
VFSM(config-ipsec-dynamic)# pfs enable|disable
```

where the following parameters are used:

- *peer's IPv6 address* The IPv6 address of the peer (for example, 3000::1)
- *index of traffic-selector* A number from 1-10
- *index of transform-set* A number from 1-10
- *SA lifetime, in seconds* The length of time the SA is to remain in effect; an integer from 120-86400
- *pfs enable|disable* Whether to enable or disable the perfect forward security feature. The default is `disable`.

**Note:** In a dynamic policy, the AH and ESP keys are created by IKEv2.

3. After you configure the IPSec policy, you need to apply it to the interface to enforce the security policies on that interface and save it to keep it in place after a reboot. To accomplish this, enter:

```
VFSM(config-ip)# interface ip <IP interface number, 1-128>
VFSM(config-ip-if)# address <IPv6 address>
VFSM(config-ip-if)# ipsec dynamic-policy <policy index, 1-10>
VFSM(config-ip-if)# enable (enable the IP interface)
VFSM(config-ip-if)# exit
VFSM(config-ip-if)# write (save the current configuration)
```

---

## Chapter 22. Routing Information Protocol

In a routed environment, routers communicate with one another to keep track of available routes. Routers can learn about available routes dynamically using the Routing Information Protocol (RIP). IBM Networking OS software supports RIP version 1 (RIPv1) and RIP version 2 (RIPv2) for exchanging TCP/IPv4 route information with other routers.

**Note:** IBM N/OS 7.8 does not support IPv6 for RIP.

---

## Distance Vector Protocol

RIP is known as a distance vector protocol. The vector is the network number and next hop, and the distance is the cost associated with the network number. RIP identifies network reachability based on metric, and metric is defined as hop count. One hop is considered to be the distance from one switch to the next, which typically is 1.

When a switch receives a routing update that contains a new or changed destination network entry, the switch adds 1 to the metric value indicated in the update and enters the network in the routing table. The IPv4 address of the sender is used as the next hop.

---

## Stability

RIP includes a number of other stability features that are common to many routing protocols. For example, RIP implements the split horizon and hold-down mechanisms to prevent incorrect routing information from being propagated.

RIP prevents routing loops from continuing indefinitely by limiting the number of hops allowed in a path from the source to a destination. The maximum number of hops in a path is 15. The network destination network is considered unreachable if increasing the metric value by 1 causes the metric to be 16 (that is infinity). This limits the maximum diameter of a RIP network to less than 16 hops.

RIP is often used in stub networks and in small autonomous systems that do not have many redundant paths.

---

## Routing Updates

RIP sends routing-update messages at regular intervals and when the network topology changes. Each router “advertises” routing information by sending a routing information update every 30 seconds. If a router doesn’t receive an update from another router for 180 seconds, those routes provided by that router are declared invalid. The routes are removed from the routing table, but they remain in the RIP routes table. After another 120 seconds without receiving an update for those routes, the routes are removed from regular updates.

When a router receives a routing update that includes changes to an entry, it updates its routing table to reflect the new route. The metric value for the path is increased by 1, and the sender is indicated as the next hop. RIP routers maintain only the best route (the route with the lowest metric value) to a destination.

For more information see The Configuration Menu, Routing Information Protocol Configuration in the *IBM Networking OS 7.8 Command Reference*.

---

## RIPv1

RIP version 1 uses broadcast User Datagram Protocol (UDP) data packets for the regular routing updates. The main disadvantage is that the routing updates do not carry subnet mask information. Hence, the router cannot determine whether the route is a subnet route or a host route. It is of limited usage after the introduction of RIPv2. For more information about RIPv1 and RIPv2, refer to RFC 1058 and RFC 2453.

---

## RIPv2

RIPv2 is the most popular and preferred configuration for most networks. RIPv2 expands the amount of useful information carried in RIP messages and provides a measure of security. For a detailed explanation of RIPv2, refer to RFC 1723 and RFC 2453.

RIPv2 improves efficiency by using multicast UDP (address 224.0.0.9) data packets for regular routing updates. Subnet mask information is provided in the routing updates. A security option is added for authenticating routing updates, by using a shared password. N/OS supports using clear password for RIPv2.

---

## RIPv2 in RIPv1 Compatibility Mode

N/OS allows you to configure RIPv2 in RIPv1 compatibility mode, for using both RIPv2 and RIPv1 routers within a network. In this mode, the regular routing updates use broadcast UDP data packet to allow RIPv1 routers to receive those packets. With RIPv1 routers as recipients, the routing updates have to carry natural or host mask. Hence, it is not a recommended configuration for most network topologies.

**Note:** When using both RIPv1 and RIPv2 within a network, use a single subnet mask throughout the network.

---

## RIP Features

N/OS provides the following features to support RIPv1 and RIPv2:

### Poison Reverse

Simple split horizon in RIP omits routes learned from one neighbor in updates sent to that neighbor. That is the most common configuration used in RIP, with the Poison Reverse feature disabled. Split horizon with poisoned reverse enabled includes such routes in updates, but sets their metrics to 16. The disadvantage of using this feature is the increase of size in the routing updates.

### Triggered Updates

Triggered updates are an attempt to speed up convergence. When Triggered Updates is enabled, whenever a router changes the metric for a route, it sends update messages almost immediately, without waiting for the regular update interval. It is recommended to enable Triggered Updates.

### Multicast

RIPv2 messages use IPv4 multicast address (224.0.0.9) for periodic updates. Multicast RIPv2 updates are not processed by RIPv1 routers. IGMP is not needed since these are inter-router messages which are not forwarded.

To configure RIPv2 in RIPv1 compatibility mode, set multicast to `disable`, and set version to `both`.

### Default Route

The RIP router can listen and supply a default route, usually represented as IPv4 0.0.0.0 in the routing table. When a router does not have an explicit route to a destination network in its routing table, it uses the default route to forward those packets.

### Metric

The metric field contains a configurable value between 1 and 15 (inclusive) which specifies the current metric for the interface. The metric value typically indicates the total number of hops to the destination. The metric value of 16 represents an unreachable destination.

### Authentication

RIPv2 authentication uses plain text password for authentication. If configured using Authentication password, then it is necessary to enter an authentication key value.

The following method is used to authenticate a RIP message:

- If the router is not configured to authenticate RIPv2 messages, then RIPv1 and unauthenticated RIPv2 messages are accepted; authenticated RIPv2 messages are discarded.
- If the router is configured to authenticate RIPv2 messages, then RIPv1 and RIPv2 messages which pass authentication testing are accepted; unauthenticated and failed authentication RIPv2 messages are discarded.

For maximum security, RIPv1 messages are ignored when authentication is enabled; otherwise, the routing information from authenticated messages is propagated by RIPv1 routers in an unauthenticated manner.

---

## RIP Configuration Example

**Note:** An interface RIP disabled uses all the default values of the RIP, no matter how the RIP parameters are configured for that interface. RIP sends out RIP regular updates to include an UP interface, but not a DOWN interface.

1. Add VLANs for routing interfaces.

```
VFSM(config)# vlan 2
VFSM(config-vlan)# exit
VFSM(config)# interface port ext2
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk allowed vlan add 2
VFSM(config-if)# exit

Port EXT2 is an Untagged/Access-mode port and its current PVID/Native-VLAN is 1.
Confirm changing PVID/Native-VLAN from 1 to 2 [y/n]: y

VFSM(config)# vlan 3
VFSM(config-vlan)# exit
VFSM(config)# interface port ext3
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk allowed vlan add 3
VFSM(config-if)# exit

Port EXT3 is an Untagged/Access-mode port and its current PVID/Native-VLAN is 1.
Confirm changing PVID/Native-VLAN from 1 to 3 [y/n]: y
```

2. Add IP interfaces with IPv4 addresses to VLANs.

```
VFSM(config)# interface ip 2
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# ip address 102.1.1.1
VFSM(config-ip-if)# vlan 2
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 3
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# ip address 103.1.1.1
VFSM(config-ip-if)# vlan 3
VFSM(config-ip-if)# exit
```

3. Turn on RIP globally and enable RIP for each interface.

```
VFSM(config)# router rip
VFSM(config-router-rip)# enable
VFSM(config-router-rip)# exit
VFSM# interface ip 2
VFSM(config-ip-if)# ip rip enable
VFSM(config-ip-if)# exit
VFSM# interface ip 3
VFSM(config-ip-if)# ip rip enable
VFSM(config-ip-if)# exit
```

Use the following command to check the current valid routes in the routing table of the switch:

```
VFSM# show ip route
```

For those RIP learnt routes within the garbage collection period, that are routes phasing out of the routing table with metric 16, use the following command:

```
VFSM# show ip rip routes
```

Locally configured static routes do not appear in the RIP Routes table.



---

## Chapter 23. Internet Group Management Protocol

Internet Group Management Protocol (IGMP) is used by IPv4 Multicast routers (Mrouters) to learn about the existence of host group members on their directly attached subnet. The IPv4 Mrouters get this information by broadcasting IGMP Membership Queries and listening for IPv4 hosts reporting their host group memberships. This process is used to set up a client/server relationship between an IPv4 multicast source that provides the data streams and the clients that want to receive the data. The switch supports three versions of IGMP:

- IGMPv1: Defines the method for hosts to join a multicast group. However, this version does not define the method for hosts to leave a multicast group. See RFC 1112 for details.
- IGMPv2: Adds the ability for a host to signal its desire to leave a multicast group. See RFC 2236 for details.
- IGMPv3: Adds support for source filtering by which a host can report interest in receiving packets only from specific source addresses, or from all but specific source addresses, sent to a particular multicast address. See RFC 3376 for details.

The Virtual Fabric Switch Module (VFSM) can perform IGMP Snooping, or act as an IGMP Relay (proxy) device.

The following topics are discussed in this chapter:

- [“IGMP Terms” on page 364](#)
- [“How IGMP Works” on page 365](#)
- [“VFSM Capacity and IGMP Default Values” on page 366](#)
- [“IGMP Snooping” on page 367](#)
- [“IGMP Relay” on page 378](#)
- [“Additional IGMP Features” on page 387](#)

---

## IGMP Terms

Listed below are the commonly used IGMP terms:

- Multicast traffic: Flow of data from one source to multiple destinations.
- Group: A multicast stream to which a host can join. Multicast groups have IP addresses in the range: 224.0.1.0 to 239.255.255.255.
- IGMP Querier: A router or switch in the subnet that generates *Membership Queries*.
- IGMP Snooper: A Layer 3 device that forwards multicast traffic only to hosts that are interested in receiving multicast data. This device can be a router or a Layer 3 switch.
- Multicast Router: A router configured to make routing decisions for multicast traffic. The router identifies the type of packet received (unicast or multicast) and forwards the packet to the intended destination.
- IGMP Proxy: A device that filters Join messages and Leave messages sent upstream to the Mrouter to reduce the load on the Mrouter.
- Membership Report: A report sent by the host that indicates an interest in receiving multicast traffic from a multicast group.
- Leave: A message sent by the host when it wants to leave a multicast group.
- FastLeave: A process by which the switch stops forwarding multicast traffic to a port as soon as it receives a Leave message.
- Membership Query: Message sent by the Querier to verify if hosts are listening to a group.
- General Query: A *Membership Query* sent to all hosts. The Group address field for general queries is 0.0.0.0 and the destination address is 224.0.0.1.
- Group-specific Query: A *Membership Query* sent to all hosts in a multicast group.

---

## How IGMP Works

When IGMP is not configured, switches forward multicast traffic through all ports, increasing network load. When IGMPv2 is configured on a switch, multicast traffic flows as follows:

- A server sends multicast traffic to a multicast group.
- The Mrouter sends *Membership Queries* to the switch, which forwards them to all ports in a given VLAN.
- Hosts respond with *Membership Reports* if they want to join a group. The switch forwards these reports to the Mrouter.
- The switch forwards multicast traffic only to hosts that have joined a group.
- The Mrouter periodically sends *Membership Queries* to ensure that a host wants to continue receiving multicast traffic. If a host does not respond, the IGMP Snooper stops sending traffic to the host.
- The host can initiate the Leave process by sending an IGMP Leave packet to the IGMP Snooper.
- When a host sends an IGMPv2 Leave packet, the IGMP Snooper queries to find out if any other host connected to the port is interested in receiving the multicast traffic. If it does not receive a Join message in response, the IGMP Snooper removes the group entry and passes on the information to the Mrouter.

---

## VFSM Capacity and IGMP Default Values

The following table lists the maximum and minimum values of the VFSM variables.

Table 31. VFSM Capacity Table

Variable	Maximum
IGMP Entries - Snoop	2048
IGMP Entries - Relay	1000
VLANs - Snoop	1023
VLANs - Relay	8
Static Mrouters	16
Dynamic Mrouters	16
Number of IGMP Filters	16
IPMC Groups (IGMP Relay)	1000

The following table lists the default settings for IGMP features and variables.

Table 32. IGMP Default Configuration Settings

Field	Default Value
Global IGMP State	Disabled
IGMP Snooping	Disabled
IGMP Querier	Not Supported
IGMP Filtering	Disabled
IP Multicast (IPMC) Flood	Enabled
IGMP FastLeave	Disabled for all VLANs
IGMP Mrouter Timeout	255 Seconds
IGMP Report Timeout Variable	10 Seconds
IGMP Robustness Variable	2
IGMPv3	Disabled
IGMPv3 number of sources	8 (The switch processes only the first eight sources listed in the IGMPv3 group record.) Valid range: 1 - 64
IGMPv3 - allow v1v2 Snooping	Enabled

---

## IGMP Snooping

IGMP Snooping allows a switch to listen to the IGMP conversation between hosts and Mrouters. By default, a switch floods multicast traffic to all ports in a broadcast domain. With IGMP Snooping enabled, the switch learns the ports interested in receiving multicast data and forwards it only to those ports. IGMP Snooping conserves network resources.

IGMP Snooping conserves bandwidth. With IGMP Snooping, the switch learns which ports are interested in receiving multicast data, and forwards multicast data only to those ports. In this way, other ports are not burdened with unwanted multicast traffic.

The switch can sense IGMP *Membership Reports* from attached hosts and act as a proxy to set up a dedicated path between the requesting host and a local IPv4 Mrouter. After the path is established, the switch blocks the IPv4 multicast stream from flowing through any port that does not connect to a host member, thus conserving bandwidth.

The client-server path is set up as follows:

- An IPv4 Multicast Router (Mrouter) sends *Membership Queries* to the switch, which forwards them to all ports in a given VLAN.
- Hosts that want to receive the multicast data stream send *Membership Reports* to the switch, which sends a proxy Membership Report to the Mrouter.
- The switch sets up a path between the Mrouter and the host, and blocks all other ports from receiving the multicast.
- Periodically, the Mrouter sends Membership Queries to ensure that the host wants to continue receiving the multicast. If a host fails to respond with a Membership Report, the Mrouter stops sending the multicast to that path.
- The host can send an IGMP Leave packet to the switch, which responds with an IGMP Groups Specific Query in order to check if there are other clients that want to receive the multicast traffic for the group referenced in the Leave packet. If an IGMP Report is not received, the group is deleted from the port and the multicast path is terminated. The switch then sends a Proxy Leave packet to the Mrouter in order to update it. If the FastLeave option is enabled on a VLAN, the multicast path is terminated immediately and the Leave packet is directly forwarded to the Mrouter.

## IGMP Groups

In both stand-alone and stacking mode, one IGMP entry is allocated for each unique Join request, based on the VLAN and IGMP group address only (regardless of the port). If multiple ports join the same IGMP group using the same VLAN, only a single IGMP entry is used.

## IGMPv3

IGMPv3 includes new Membership Report messages that extend IGMP functionality. The VFSM provides snooping capability for all types of IGMPv3 *Membership Reports*.

IGMPv3 is supported in stand-alone (non-stacking) mode only.

IGMPv3 supports Source-Specific Multicast (SSM). SSM identifies session traffic by both source and group addresses. The VFSM uses *source filtering*, which allows hosts to report interest in receiving multicast packets only from specific source addresses, or from all but specific source addresses.

The VFSM supports the following IGMPv3 filter modes:

- **INCLUDE mode:** The host requests membership to a multicast group and provides a list of IPv4 addresses from which it wants to receive traffic.
- **EXCLUDE mode:** The host requests membership to a multicast group and provides a list of IPv4 addresses from which it does not want to receive traffic. This indicates that the host wants to receive traffic only from sources that are not part of the Exclude list. To disable snooping on EXCLUDE mode reports, use the following command:

```
EN 4093R(config)# no ip igmp snoop igmpv3 exclude
```

By default, the VFSM snoops the first eight sources listed in the IGMPv3 Group Record. Use the following command to change the number of snooping sources:

```
VFSM(config)# ip igmp snoop igmpv3 sources <1-64>
```

IGMPv3 Snooping is compatible with IGMPv1 and IGMPv2 Snooping. To disable snooping on version 1 and version 2 reports, use the following command:

```
VFSM(config)# no ip igmp snoop igmpv3 v1v2
```

## IGMP Snooping Configuration Guidelines

Consider the following guidelines when you configure IGMP Snooping:

- IGMP operation is independent of the routing method. You can use RIP, OSPF, or static routes for Layer 3 routing.
- When multicast traffic flood is disabled, the multicast traffic sent by the multicast server is discarded if no hosts or Mrouter are learned on the switch.
- The Mrouter periodically sends IGMP Queries.
- The switch learns the Mrouter on the port connected to the router when it sees Query messages. The switch then floods the IGMP queries on all other ports including a Trunk Group, if any.
- Multicast hosts send IGMP Reports as a reply to the IGMP Queries sent by the Mrouter.
- The switch can also learn an Mrouter when it receives a PIM Hello packet from another device. However, an Mrouter learned from a PIM packet has a lower priority than an Mrouter learned from an IGMP Query. A switch overwrites an Mrouter learned from a PIM packet when it receives an IGMP Query on the same port.
- A host sends an IGMP Leave message to its multicast group. The expiration timer for this group is updated to IGMP timeout variable (the default is 10 seconds). The L3 switch sends IGMP Group-Specific Query to the host that had sent the Leave message. If the host does not respond with an IGMP Report during the timeout interval, all the groups expire and the switch deletes the host from the IGMP groups table. The switch then proxies the IGMP Leave messages to the Mrouter.

## IGMP Snooping Configuration Example

This section provides steps to configure IGMP Snooping on the VFSM.

1. Configure port and VLAN membership on the switch.
2. Add VLANs to IGMP Snooping and enable the feature.

```
VFSM(config)# ip igmp snoop vlan 1      (Add VLAN 1 to IGMP snooping)
VFSM(config)# ip igmp snoop enable      (Enable IGMP Snooping)
```

3. Enable IGMPv3 Snooping (optional).

```
VFSM(config)# ip igmp snoop igmpv3 enable (Enable IGMPv3 Snooping)
```

4. Enable IGMP.

```
VFSM(config)# ip igmp enable            (Turn on IGMP)
```

5. View dynamic IGMP information.

To display information about IGMP Groups:

```
VFSM(config)# show ip igmp groups

Total entries: 5  Total IGMP groups: 3
Note: The <Total IGMP groups> number is computed as
       the number of unique (Group, Vlan) entries!

Note: Local groups (224.0.0.x) are not snooped/relayed and do not appear.

```

Source	Group	VLAN	Port	Version	Mode	Expires	Fwd
10.1.1.1	232.1.1.1	2	EXT4	V3	INC	4:16	Yes
10.1.1.5	232.1.1.1	2	EXT4	V3	INC	4:16	Yes
*	232.1.1.1	2	EXT4	V3	INC	-	No
10.10.10.43	235.0.0.1	9	EXT1	V3	INC	2:26	Yes
*	236.0.0.1	9	EXT1	V3	EXC	-	Yes

To display information about Mrouters learned by the switch:

```
VFSM(config)# show ip igmp mrouter

Total entries: 2 Total number of dynamic mrouter: 1
Total number of installed static mrouter : 1

```

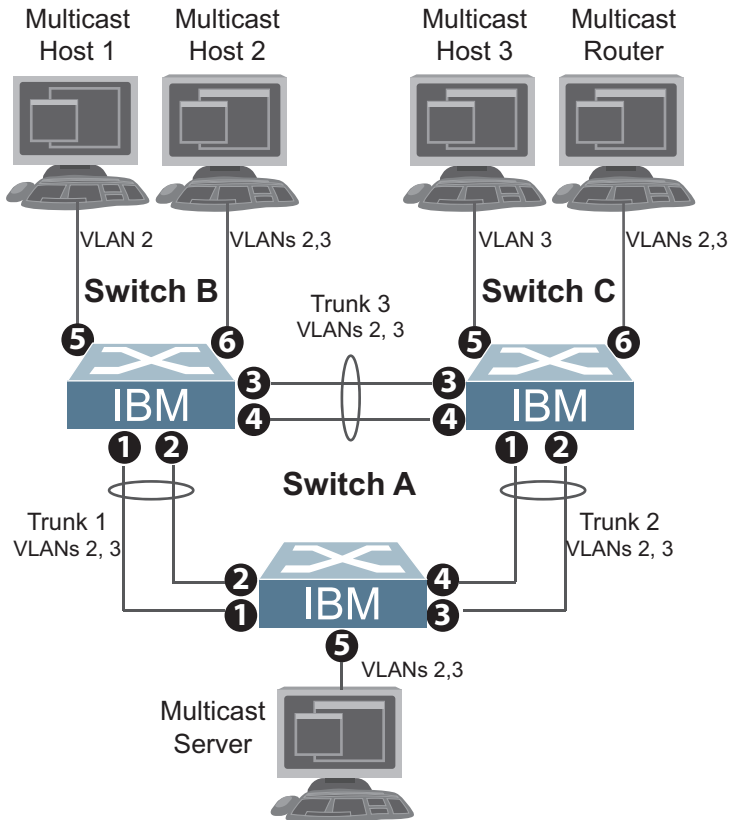
SrcIP	VLAN	Port	Version	Expires	MRT	QRV	QQIC
*	1	EXT4	V2	static	-	-	-
10.10.10.10	2	EXT3	V3	4:09	128	2	125

**Note:** If IGMP Snooping v1/v2 is enabled and IGMPv3 Snooping is disabled, the output of IGMPv3 reports and queries show some items as IGMPv3 (v3), though they retain v2 behavior. For example, the Source IPv4 address is not relevant for v2 entries.

## Advanced Configuration Example: IGMP Snooping

Figure 36 shows an example topology. Switches B and C are configured with IGMP Snooping.

Figure 36. Topology



Devices in the above topology are configured as follows:

- STG 2 includes VLAN 2; STG 3 includes VLAN 3.
- The multicast server sends IP multicast traffic for the following groups:
  - VLAN 2, 225.10.0.11 – 225.10.0.12, Source: 22.10.0.11
  - VLAN 2, 225.10.0.13 – 225.10.0.15, Source: 22.10.0.13
  - VLAN 3, 230.0.2.1 – 230.0.2.2, Source: 22.10.0.1
  - VLAN 3, 230.0.2.3 – 230.0.2.5, Source: 22.10.0.3
- The Mrouter sends IGMP Query packets in VLAN 2 and VLAN 3. The Mrouter's IP address is 10.10.10.10.
- The multicast hosts send the following IGMP Reports:
  - IGMPv2 Report, VLAN 2, Group: 225.10.0.11, Source: \*
  - IGMPv2 Report, VLAN 3, Group: 230.0.2.1, Source: \*
  - IGMPv3 IS\_INCLUDE Report, VLAN 2, Group: 225.10.0.13, Source: 22.10.0.13
  - IGMPv3 IS\_INCLUDE Report, VLAN 3, Group: 230.0.2.3, Source: 22.10.0.3

- The hosts receive multicast traffic as follows:
  - Host 1 receives multicast traffic for groups (\*, 225.10.0.11), (22.10.0.13, 225.10.0.13)
  - Host 2 receives multicast traffic for groups (\*, 225.10.0.11), (\*, 230.0.2.1), (22.10.0.13, 225.10.0.13), (22.10.0.3, 230.0.2.3)
  - Host 3 receives multicast traffic for groups (\*, 230.0.2.1), (22.10.0.3, 230.0.2.3)
- The Mrouter receives all the multicast traffic.

## Prerequisites

Before you configure IGMP Snooping, ensure you have performed the following actions:

- Configured VLANs.
- Enabled IGMP.
- Configured a switch or Mrouter as the Querier.
- Identified the IGMP version(s) you want to enable.
- Disabled IGMP flooding.

## Configuration

This section provides the configuration details of the switches shown in [Figure 36](#).

### Switch A Configuration

1. Configure VLANs 2 and 3, enable tagging/trunk mode on ports EXT1 through EXT5 and add them to the VLANs 2 and 3.

```
VFSM(config)# interface port EXT1-EXT5
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk allowed 2,3
VFSM(config-if)# exit
```

2. Configure an IP interface with IPv4 address, and assign a VLAN.

```
VFSM(config)# interface ip 1
VFSM(config-ip-if)# ip address 10.10.10.1 enable
VFSM(config-ip-if)# vlan 2
VFSM(config-ip-if)# exit
```

3. Configure STP. Assign a bridge priority lower than the default bridge priority to enable the switch to become the STP root in STG 2 and STG 3.

```
VFSM(config)# spanning-tree stp 2 bridge priority 4096
VFSM(config)# spanning-tree stp 3 bridge priority 4096
```

4. Configure LACP dynamic trunk groups (portchannels).

```
VFSM(config)# interface port EXT1
VFSM(config-if)# lacp key 100
VFSM(config-if)# lacp mode active
VFSM(config-if)# exit

VFSM(config)# interface port EXT2
VFSM(config-if)# lacp key 100
VFSM(config-if)# lacp mode active
VFSM(config-if)# exit

VFSM(config)# interface port EXT3
VFSM(config-if)# lacp key 200
VFSM(config-if)# lacp mode active
VFSM(config-if)# exit

VFSM(config)# interface port EXT4
VFSM(config-if)# lacp key 200
VFSM(config-if)# lacp mode active
```

## Switch B Configuration

1. Configure VLANs 2 and 3. Enable tagging/trunk mode on ports EXT1 through EXT4 and port EXT6. Add the ports to VLANs 2 and 3.

```
VFSM(config)# interface port EXT1-EXT4,EXT6
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk allowed 2,3
VFSM(config-if)# exit
```

2. Remove ports EXT1 through EXT6 from VLAN 1.

```
VFSM(config)# vlan 1
VFSM(config-vlan)# no member EXT1-EXT6
VFSM(config-if)# exit
```

3. Configure an IP interface with IPv4 address, and assign a VLAN.

```
VFSM(config)# interface ip 1
VFSM(config-ip-if)# ip address 10.10.10.2 enable
VFSM(config-ip-if)# vlan 2
VFSM(config-ip-if)# exit
```

4. Configure STP. Reset the ports to make the edge configuration operational.

```
VFSM(config)# interface port EXT5,EXT6
VFSM(config-if)# spanning-tree edge
VFSM(config-if)# shutdown
VFSM(config-if)# no shutdown
VFSM(config-if)# exit
```

5. Configure an LACP dynamic trunk group (portchannel).

```
VFSM(config)# interface port EXT1,EXT2
VFSM(config-if)# lacp key 300
VFSM(config-if)# lacp mode active
VFSM(config-if)# exit
```

6. Configure a static trunk group (portchannel).

```
VFSM(config)# portchannel 1 port EXT3,EXT4 enable
```

7. Configure IGMP Snooping.

```
VFSM(config)# ip igmp enable
VFSM(config)# ip igmp snoop vlan 2,3
VFSM(config)# ip igmp snoop source-ip 10.10.10.2
VFSM(config)# ip igmp snoop igmpv3 enable
VFSM(config)# ip igmp snoop igmpv3 sources 64
VFSM(config)# ip igmp snoop enable

VFSM(config)# vlan 2
VFSM(config-vlan)# no flood
VFSM(config-vlan)# exit

VFSM(config)# vlan 3
VFSM(config-vlan)# no flood
VFSM(config-vlan)# exit
```

## Switch C Configuration

1. Configure VLAN 2. Add ports EXT1 through EXT4 and port EXT6 to VLAN 2. Enable tagging/trunk mode.

```
VFSM(config)# interface port EXT1-EXT4,EXT6
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk allowed 2
VFSM(config-if)# exit
```

2. Configure VLAN 3. Add ports EXT1 through EXT6 to VLAN 3.

```
VFSM(config)# vlan 3
VFSM(config-vlan)# enable
VFSM(config-vlan)# member EXT1-EXT6
VFSM(config-vlan)# exit
```

3. Remove ports EXT1 through EXT6 from VLAN 1.

```
VFSM(config)# vlan 1
VFSM(config-vlan)# no member EXT1-EXT6
VFSM(config-vlan)# exit
```

4. Configure an IP interface with IPv4 address, and assign a VLAN.

```
VFSM(config)# interface ip 1
VFSM(config-ip-if)# ip address 10.10.10.3 enable
VFSM(config-ip-if)# vlan 2
VFSM(config-ip-if)# exit
```

5. Configure STP. Reset the ports to make the edge configuration operational.

```
VFSM(config)# interface port EXT5,EXT6
VFSM(config-if)# spanning-tree edge
VFSM(config-if)# shutdown
VFSM(config-if)# no shutdown
VFSM(config-if)# exit
```

6. Configure an LACP dynamic trunk group (portchannel).

```
VFSM(config)# interface port EXT1,EXT2
VFSM(config-if)# lacp key 400
VFSM(config-if)# lacp mode active
VFSM(config-if)# exit
```

7. Configure a static trunk group (portchannel).

```
VFSM(config)# portchannel EXT1 port EXT3,EXT4 enable
```

8. Configure IGMP Snooping.

```
VFSM(config)# ip igmp enable
VFSM(config)# ip igmp snoop vlan 2,3
VFSM(config)# ip igmp snoop source-ip 10.10.10.3
VFSM(config)# ip igmp snoop igmpv3 enable
VFSM(config)# ip igmp snoop igmpv3 sources 64
VFSM(config)# ip igmp snoop enable

VFSM(config)# vlan 2
VFSM(config-vlan)# no flood
VFSM(config-vlan)# exit

VFSM(config)# vlan 3
VFSM(config-vlan)# no flood
VFSM(config-vlan)# exit
```

## Troubleshooting

This section provides the steps to resolve common IGMP Snooping configuration issues. The topology described in [Figure 36](#) is used as an example.

### Issue: Multicast traffic from non-member groups reaches the host or Mrouter

- Check if traffic is unregistered. For unregistered traffic, an IGMP entry is not displayed in the IGMP groups table.

```
VFSM# show ip igmp groups
```

- Ensure IPMC flooding is disabled and CPU is enabled.

```
VFSM(config)# vlan <vlan id>
VFSM(config-vlan)# no flood
VFSM(config-vlan)# cpu
```

- Check the egress port's VLAN membership. The ports to which the hosts and Mrouter are connected must be used only for VLAN 2 and VLAN 3.

```
VFSM# show vlan
```

**Note:** To avoid such a scenario, disable IPMC flooding for all VLANs enabled on the switches (if this is an acceptable configuration).

- Check IGMP Reports on switches B and C for information about the IGMP groups.

```
VFSM# show ip igmp groups
```

If the non-member IGMP groups are displayed in the table, close the application that may be sending the IGMP Reports for these groups.

Identify the traffic source by using a sniffer on the hosts and reading the source IP/MAC address. If the source IP/MAC address is unknown, check the port statistics to find the ingress port.

```
VFSM# show interface port <port id> interface-counters
```

- Ensure no static multicast MACs, static multicast groups, or static Mrouters are configured.

```
VFSM# show oam
VFSM# show ip igmp ipmcgrp
VFSM# show ip igmp mrouter
```

- Ensure IGMP Relay is not configured.

```
VFSM# show ip igmp relay
```

### Issue: Not all multicast traffic reaches the appropriate receivers

- Ensure hosts are sending IGMP Reports for all the groups. Check the VLAN on which the groups are learned.

```
VFSM# show ip igmp groups
```

If some of the groups are not displayed, ensure the multicast application is running on the host device and the generated IGMP Reports are correct.

- Ensure multicast traffic reaches the switch to which the host is connected. Close the application sending the IGMP Reports. Clear the IGMP groups by flapping (disabling, then re-enabling) the port.

**Note:** To clear all IGMP groups, use the following command:

```
VFSM(config)# clear ip igmp groups
```

However, this will clear all the IGMP groups and will influence other hosts.

Check if the multicast traffic reaches the switch.

```
VFSM# show ip igmp ipmcgrp
```

If the multicast traffic group is not displayed in the table, check the link state, VLAN membership, and STP convergence.

- Ensure multicast server is sending all the multicast traffic.
- Ensure no static multicast MACs, static multicast groups, or static multicast routes are configured.

### Issue: IGMP queries sent by the Mrouter do not reach the host

- Ensure the Mrouter is learned on switches B and C.

```
VFSM# show ip igmp mrouter
```

If it is not learned on switch B but is learned on switch C, check the link state of the trunk group, VLAN membership, and STP convergence.

If it is not learned on any switch, ensure the multicast application is running and is sending correct IGMP Query packets.

If it is learned on both switches, check the link state, VLAN membership, and STP port states for the ports connected to the hosts.

### Issue: IGMP Reports/Leaves sent by the hosts do not reach the Mrouter

- Ensure IGMP Queries sent by the Mrouter reach the hosts.
- Ensure the Mrouter is learned on both switches. Note that the Mrouter may not be learned on switch B immediately after a trunk group failover/failback.

```
VFSM# show ip igmp mrouter
```

- Ensure the host's multicast application is started and is sending correct IGMP Reports/Leaves.

```
VFSM# show ip igmp groups
VFSM# show ip igmp counters
```

#### **Issue: Multicast traffic reaches incorrect VLAN**

- Check port VLAN membership.
- Check IGMP Reports sent by the host.
- Check multicast data sent by the server.

#### **Issue: The Mrouter is learned on the incorrect trunk group**

- Check link state. Trunk group 1 might be down or in STP discarding state.
- Check STP convergence.
- Check port VLAN membership.

#### **Issue: Hosts receive multicast traffic at a lower rate than normal**

**Note:** This behavior is expected if IPMC flood is disabled and CPU is enabled. As soon as the IGMP/IPMC entries are installed on ASIC, the IPMC traffic recovers and is forwarded at line rate. This applies to unregistered IPMC traffic.

- Ensure a multicast threshold is not configured on the trunks.

```
VFSM(config)# interface port <port id>
VFSM(config-ip)# no multicast-threshold
```

- Check link speeds and network congestion.

---

## IGMP Relay

The VFSM can act as an IGMP Relay (or IGMP Proxy) device that relays IGMP multicast messages and traffic between an Mrouter and end stations. IGMP Relay allows the VFSM to participate in network multicasts with no configuration of the various multicast routing protocols, so you can deploy it in the network with minimal effort.

To an IGMP host connected to the VFSM, IGMP Relay appears to be an IGMP Mrouter. IGMP Relay sends *Membership Queries* to hosts, which respond by sending an IGMP response message. A host can also send an unsolicited Join message to the IGMP Relay.

To an Mrouter, IGMP Relay appears as a host. The Mrouter sends IGMP host queries to IGMP Relay, and IGMP Relay responds by forwarding IGMP host reports and unsolicited Join messages from its attached hosts.

IGMP Relay also forwards multicast traffic between the Mrouter and end stations, similar to IGMP Snooping.

You can configure up to two Mrouters to use with IGMP Relay. One Mrouter acts as the primary Mrouter, and one is the backup Mrouter. The VFSM uses health checks to select the primary Mrouter.

## Configuration Guidelines

Consider the following guidelines when you configure IGMP Relay:

- IGMP Relay is supported in stand-alone (non-stacking) mode only.
- IGMP Relay and IGMP Snooping are mutually exclusive—if you enable IGMP Relay, you must turn off IGMP Snooping.
- Add the upstream Mrouter VLAN to the IGMP Relay list, using the following command:

```
VFSM(config)# ip igmp relay vlan <VLAN number>
```

- If IGMP hosts reside on different VLANs, you must:
  - Disable IGMP flooding.

```
VFSM(config)# vlan <vlan id>  
VFSM(config-vlan)# no flood
```

- Enable CPU forwarding to ensure that multicast data is forwarded across the VLANs.

```
VFSM(config)# vlan <vlan id>  
VFSM(config-vlan)# cpu
```

## Configure IGMP Relay

Use the following procedure to configure IGMP Relay.

1. Configure IP interfaces with IPv4 addresses, and assign VLANs.

```
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip address 10.10.1.1 255.255.255.0 enable
VFSM(config-ip-if)# vlan 2
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 3
VFSM(config-ip-if)# ip address 10.10.2.1 255.255.255.0 enable
VFSM(config-ip-if)# vlan 3
VFSM(config-ip-if)# exit
```

2. Turn on IGMP.

```
VFSM(config)# ip igmp enable
```

3. Configure the upstream Mrouters with IPv4 addresses.

```
VFSM(config)# ip igmp relay mrouter 1 address 100.0.1.2
VFSM(config)# ip igmp relay mrouter 1 enable
VFSM(config)# ip igmp relay mrouter 2 address 100.0.2.4
VFSM(config)# ip igmp relay mrouter 2 enable
```

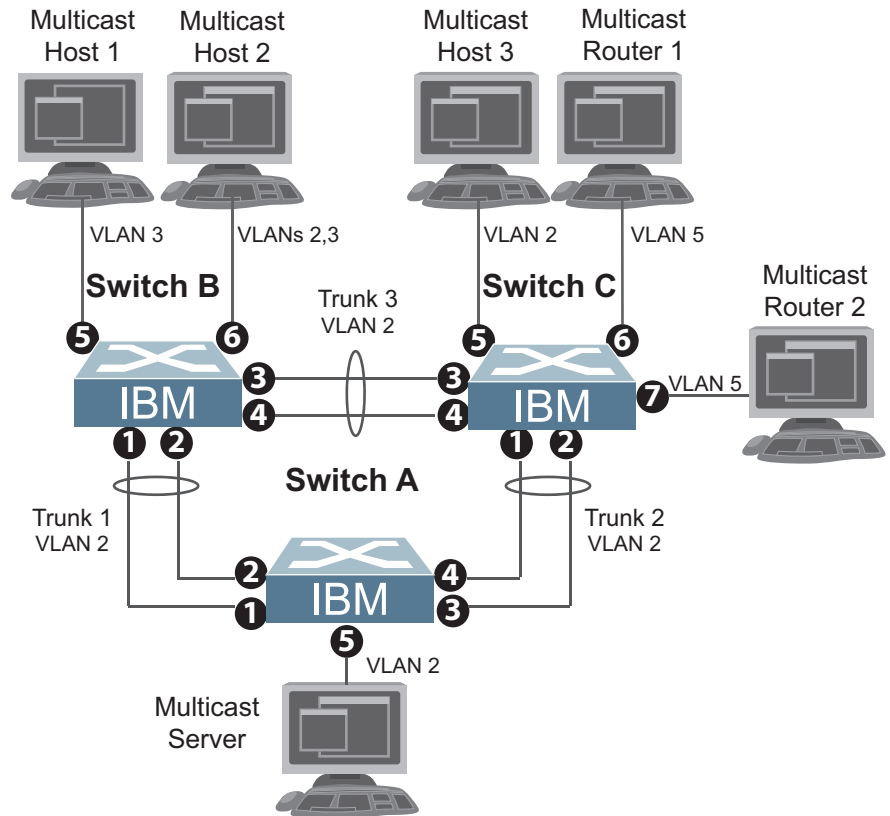
4. Add VLANs to the downstream network and enable IGMP Relay

```
VFSM(config)# ip igmp relay vlan 2
VFSM(config)# ip igmp relay vlan 3
VFSM(config)# ip igmp relay enable
```

## Advanced Configuration Example: IGMP Relay

Figure 37 shows an example topology. Switches B and C are configured with IGMP Relay.

Figure 37. Topology



Devices in the above topology are configured as follows:

- The IP address of Multicast Router 1 is 5.5.5.5
- The IP address of Multicast Router 2 is 5.5.5.6
- STG 2 includes VLAN2; STG 3 includes VLAN 3; STG 5 includes VLAN 5.
- The multicast server sends IP multicast traffic for the following groups:
  - VLAN 2, 225.10.0.11 – 225.10.0.15
- The multicast hosts send the following IGMP Reports:
  - Host 1: 225.10.0.11 – 225.10.0.12, VLAN 3
  - Host 2: 225.10.0.12 – 225.10.0.13, VLAN 2; 225.10.0.14 – 225.10.0.15, VLAN 3
  - Host 3: 225.10.0.13 – 225.10.0.14, VLAN 2
- The Mrouter receives all the multicast traffic.

### Prerequisites

Before you configure IGMP Snooping, ensure you have performed the following actions:

- Configure VLANs.
- Enable IGMP.
- Identify the IGMP version(s) you want to enable.
- Disable IGMP flooding.
- Disable IGMP Snooping.

## Configuration

This section provides the configuration details of the switches in [Figure 37](#).

### Switch A Configuration

1. Configure a VLAN.

```
VFSM(config)# vlan 2
VFSM(config-vlan)# enable
VFSM(config-vlan)# member EXT1-EXT5
VFSM(config-vlan)# exit
```

2. Configure an IP interface with IPv4 address, and assign a VLAN..

```
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip address 2.2.2.10 enable
VFSM(config-ip-if)# vlan 2
VFSM(config-ip-if)# exit
```

3. Configure the PVRST root bridge. Assign a bridge priority lower than the default bridge priority to enable the switch to become the STP root in STG 2 and 3.

```
VFSM(config)# spanning-tree stp 2 bridge priority 4096
```

4. Configure LACP dynamic trunk groups (portchannels).

```
VFSM(config)# interface port EXT1,EXT2
VFSM(config-if)# lacp key 100
VFSM(config-if)# lacp mode active
VFSM(config-if)# exit

VFSM(config)# interface port EXT3,EXT4
VFSM(config-if)# lacp key 200
VFSM(config-if)# lacp mode active
```

## Switch B Configuration

### 1. Configure VLANs and tagging.

```
VFSM(config)# vlan 2
VFSM(config-vlan)# enable
VFSM(config-vlan)# member EXT1,EXT2,EXT3,EXT4,EXT6
VFSM(config-vlan)# exit

VFSM(config)# interface port EXT6
VFSM(config-ip-if)# tagging
VFSM(config-ip-if)# exit

VFSM(config)# vlan 3
VFSM(config-vlan)# enable
VFSM(config-vlan)# member EXT5,EXT6
VFSM(config-vlan)# exit
```

### 2. Configure IP interfaces with IPv4 addresses, and assign VLANs.

```
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip address 2.2.2.20 enable
VFSM(config-ip-if)# vlan 2
VFSM(config-ip-if)# exit

VFSM(config)# interface ip 3
VFSM(config-ip-if)# ip address 3.3.3.20 enable
VFSM(config-ip-if)# vlan 3
VFSM(config-ip-if)# exit

VFSM(config)# ip gateway 2 address 2.2.2.30 enable
```

### 3. Configure STP.

```
VFSM(config)# interface port EXT5,EXT6
VFSM(config-if)# spanning-tree edge
VFSM(config-if)# shutdown
VFSM(config-if)# no shutdown
VFSM(config-if)# exit
```

### 4. Configure an LACP dynamic trunk group (portchannel).

```
VFSM(config)# interface port EXT1,EXT2
VFSM(config-if)# lacp key 300
VFSM(config-if)# lacp mode active
VFSM(config-if)# exit
```

### 5. Configure a static trunk group (portchannel).

```
VFSM(config)# portchannel 1 port EXT3,EXT4 enable
```

## 6. Configure IGMP Relay.

```
VFSM(config)# ip igmp enable
VFSM(config)# ip igmp relay vlan 2,3
VFSM(config)# ip igmp relay mrouter 1 address 5.5.5.5 enable
VFSM(config)# ip igmp relay mrouter 2 address 5.5.5.6 enable
VFSM(config)# ip igmp relay enable

VFSM(config)# vlan 2
VFSM(config-vlan)# no flood
VFSM(config-vlan)# exit

VFSM(config)# vlan 3
VFSM(config-vlan)# no flood
VFSM(config-vlan)# exit
```

## Switch C Configuration

### 1. Configure VLANs.

```
VFSM(config)# vlan 2
VFSM(config-vlan)# enable
VFSM(config-vlan)# member EXT1-EXT5
VFSM(config-vlan)# exit

VFSM(config)# vlan 5
VFSM(config-vlan)# enable
VFSM(config-vlan)# member EXT6,EXT7
VFSM(config-vlan)# exit
```

### 2. Configure IP interfaces with IPv4 addresses and assign VLANs.

```
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip address 2.2.2.30 enable
VFSM(config-ip-if)# vlan 2
VFSM(config-ip-if)# exit

VFSM(config)# interface ip 3
VFSM(config-ip-if)# ip address 5.5.5.30 enable
VFSM(config-ip-if)# vlan 5
VFSM(config-ip-if)# exit

VFSM(config)# ip gateway 2 address 2.2.2.20 enable
```

### 3. Configure STP.

```
VFSM(config)# interface port EXT5,EXT6,EXT7
VFSM(config-if)# spanning-tree edge
VFSM(config-if)# shutdown
VFSM(config-if)# no shutdown
VFSM(config-if)# exit
```

### 4. Configure LACP dynamic trunk group (portchannel).

```
VFSM(config)# interface port EXT1,EXT2
VFSM(config-if)# lacp key 400
VFSM(config-if)# lacp mode active
VFSM(config-if)# exit
```

5. Configure a static trunk group (portchannel).

```
VFSM(config)# portchannel 1 port EXT3,EXT4 enable
```

6. Enable IGMP.

```
VFSM(config)# ip igmp enable
```

7. Configure IGMP Relay.

```
VFSM(config)# ip igmp relay vlan 2,5
VFSM(config)# ip igmp relay mrouter 1 address 5.5.5.5 enable
VFSM(config)# ip igmp relay mrouter 2 address 5.5.5.6 enable
VFSM(config)# ip igmp relay enable

VFSM(config)# vlan 2
VFSM(config-vlan)# no flood
VFSM(config-vlan)# exit

VFSM(config)# vlan 5
VFSM(config-vlan)# no flood
VFSM(config-vlan)# exit
```

## Troubleshooting

This section provides the steps to resolve common IGMP Relay configuration issues. The topology described in [Figure 37](#) is used as an example.

### Issue: Multicast traffic from non-member groups reaches the hosts or Router

- Ensure IPMC flood is disabled.

```
VFSM(config)# vlan <vlan id>
VFSM(config-vlan)# no flood
```

- Check the egress port's VLAN membership. The ports to which the hosts and Mrouter are connected must be used only for VLAN 2, VLAN 3, or VLAN 5.

```
VFSM(config)# show vlan
```

**Note:** To avoid such a scenario, disable IPMC flooding for all VLANs enabled on the switches (if this is an acceptable configuration).

- Check IGMP Reports on switches B and C for information about IGMP groups.

```
VFSM(config)# show ip igmp groups
```

If non-member IGMP groups are displayed in the table, close the application that may be sending the IGMP Reports for these groups.

Identify the traffic source by using a sniffer on the hosts and reading the source IP address/MAC address. If the source IP address/MAC address is unknown, check the port statistics to find the ingress port.

```
VFSM(config)# show interface port <port id> interface-counters
```

- Ensure no static multicast MACs and static Mroouters are configured.

### Issue: Not all multicast traffic reaches the appropriate receivers

- Ensure hosts are sending IGMP Reports for all the groups. Check the VLAN on which the groups are learned.

```
VFSM(config)# show ip igmp groups
```

If some of the groups are not displayed, ensure the multicast application is running on the host device and the generated IGMP Reports are correct.

- Ensure the multicast traffic reaches the switch to which the host is connected. Close the application sending the IGMP Reports. Clear the IGMP groups by flapping (disabling, then re-enabling) the port.

**Note:** To clear all IGMP groups, use the following commands:

```
VFSM(config)# clear ip igmp groups
```

However, this will clear all the IGMP groups and will influence other hosts.

Check if the multicast traffic reaches the switch.

```
VFSM(config)# show ip igmp ipmcgrp
```

If the multicast traffic group is not displayed in the table, check the link state, VLAN membership, and STP convergence.

- Ensure the multicast server is sending all the multicast traffic.
- Ensure no static multicast MACs or static multicast routes are configured.
- Ensure PIM is not enabled on the switches.

### Issue: IGMP Reports/Leaves sent by the hosts do not reach the Mrouter

- Ensure one of the Mrouters is learned on both switches. If not, the IGMP Reports/Leaves are not forwarded. Note that the Mrouter may not be learned on switch B immediately after a trunk group failover/failback.

```
VFSM(config)# show ip igmp mrouter
```

- Ensure the host's multicast application is started and is sending correct IGMP Reports/Leaves.

```
VFSM(config)# show ip igmp groups
VFSM(config)# show ip igmp counters
```

### Issue: The Mrouter is learned on the incorrect trunk group

- Check link state. Trunk group 1 may be down or in STP discarding state.
- Check STP convergence.
- Check port VLAN membership.

**Issue: Hosts receive multicast traffic at a lower rate than normal.**

- Ensure a multicast threshold is not configured on the trunk groups.

```
VFSM(config)# interface port <port id>  
VFSM(config-if)# no multicast-threshold
```

- Check link speeds and network congestion.

---

## Additional IGMP Features

The following topics are discussed in this section:

- [“FastLeave” on page 387](#)
- [“IGMP Filtering” on page 387](#)
- [“Static Multicast Router” on page 388](#)

### FastLeave

In normal IGMP operation, when the switch receives an IGMPv2 Leave message, it sends a Group-Specific Query to determine if any other devices in the same group (and on the same port) are still interested in the specified multicast group traffic. The switch removes the affiliated port from that particular group if the switch does not receive an IGMP Membership Report within the query-response-interval.

With FastLeave enabled on the VLAN, a port can be removed immediately from the port list of the group entry when the IGMP Leave message is received.

**Note:** Only IGMPv2 supports FastLeave. Enable FastLeave on ports that have only one host connected. If more than one host is connected to a port, you may lose some hosts unexpectedly.

Use the following command to enable FastLeave.

```
VFSM(config)# ip igmp fastleave <VLAN number>
```

### IGMP Filtering

With IGMP filtering, you can allow or deny certain IGMP groups to be learned on a port. IGMP filtering works with all versions of IGMP.

If access to a multicast group is denied, IGMP *Membership Reports* from the port are dropped, and the port is not allowed to receive IPv4 multicast traffic from that group. If access to the multicast group is allowed, *Membership Reports* from the port are forwarded for normal processing.

To configure IGMP filtering, you must globally enable IGMP filtering, define an IGMP filter, assign the filter to a port, and enable IGMP filtering on the port. To define an IGMP filter, you must configure a range of IPv4 multicast groups, choose whether the filter will allow or deny multicast traffic for groups within the range, and enable the filter.

#### Configuring the Range

Each IGMP filter allows you to set a start and end point that defines the range of IPv4 addresses upon which the filter takes action. Each IPv4 address in the range must be between 224.0.0.0 and 239.255.255.255.

#### Configuring the Action

Each IGMP filter can allow or deny IPv4 multicasts to the range of IPv4 addresses configured. If you configure the filter to deny IPv4 multicasts, then IGMP *Membership Reports* from multicast groups within the range are dropped. You can configure another filter with a numerically lower ID to allow IPv4 multicasts to a small

range of addresses within a larger range that a filter with a numerically higher ID is configured to deny. The two filters work together to allow IPv4 multicasts to a small subset of addresses within the larger range of addresses.

**Note:** Lower-numbered filters take precedence over higher-number filters. For example, the action defined for IGMP filter 1 supersedes the action defined for IGMP filter 2.

### Configure IGMP Filtering

1. Enable IGMP filtering on the switch.

```
VFSM(config)# ip igmp filtering
```

2. Define an IGMP filter with IPv4 information.

```
VFSM(config)# ip igmp profile 1 range 224.0.0.0 226.0.0.0
VFSM(config)# ip igmp profile 1 action deny
VFSM(config)# ip igmp profile 1 enable
```

3. Assign the IGMP filter to a port.

```
VFSM(config)# interface port 3
VFSM(config-if)# ip igmp profile 1
VFSM(config-if)# ip igmp filtering
```

## Static Multicast Router

A static Mrouter can be configured for a particular port on a particular VLAN. A static Mrouter does not have to be learned through IGMP Snooping.

A total of 16 static Mrouters can be configured on the VFSM. Both internal and external ports can accept a static Mrouter.

**Note:** When static Mrouters are used, the switch will continue learning dynamic Mrouters via IGMP snooping. However, dynamic Mrouters will not replace static Mrouters. If a dynamic Mrouter has the same port and VLAN combination as a static Mrouter, the dynamic Mrouter will not be learned.

Following is an example of configuring a static multicast router:

1. For each Mrouter, configure a port, VLAN, and IGMP version of the multicast router.

```
VFSM(config)# ip igmp mrouter EXT5 1 2
```

2. Verify the configuration.

```
VFSM(config)# show ip igmp mrouter
```

---

## Chapter 24. Multicast Listener Discovery

Multicast Listener Discovery (MLD) is an IPv6 protocol that a host uses to request multicast data for a multicast group. An IPv6 router uses MLD to discover the presence of multicast listeners (nodes that want to receive multicast packets) on its directly attached links, and to discover specifically the multicast addresses that are of interest to those neighboring nodes.

MLD version 1 is derived from Internet Group Management Protocol version 2 (IGMPv2) and MLDv2 is derived from IGMPv3. MLD uses ICMPv6 (IP Protocol 58) message types. See RFC 2710 and RFC 3810 for details.

MLDv2 protocol, when compared to MLDv1, adds support for source filtering—the ability for a node to report interest in listening to packets only from specific source addresses, or from all but specific source addresses, sent to a particular multicast address. MLDv2 is interoperable with MLDv1. See RFC 3569 for details on Source-Specific Multicast (SSM).

The following topics are discussed in this chapter:

- [“MLD Terms” on page 390](#)
- [“How MLD Works” on page 391](#)
- [“MLD Capacity and Default Values” on page 394](#)
- [“Configuring MLD” on page 395](#)

---

## MLD Terms

Following are the commonly used MLD terms:

- Multicast traffic: Flow of data from one source to multiple destinations.
- Group: A multicast stream to which a host can join.
- Multicast Router (Mrouter): A router configured to make routing decisions for multicast traffic. The router identifies the type of packet received (unicast or multicast) and forwards the packet to the intended destination.
- Querier: An Mrouter that sends periodic query messages. Only one Mrouter on the subnet can be elected as the Querier.
- Multicast Listener Query: Messages sent by the Querier. There are three types of queries:
  - General Query: Sent periodically to learn multicast address listeners from an attached link. VFSM uses these queries to build and refresh the Multicast Address Listener state. General Queries are sent to the link-scope all-nodes multicast address (FF02::1), with a multicast address field of 0, and a maximum response delay of *query response interval*.
  - Multicast Address Specific Query: Sent to learn if a specific multicast address has any listeners on an attached link. The multicast address field is set to the IPv6 multicast address.
  - Multicast Address and Source Specific Query: Sent to learn if, for a specified multicast address, there are nodes still listening to a specific set of sources. Supported only in MLDv2.

**Note:** Multicast Address Specific Queries and Multicast Address and Source Specific Queries are sent only in response to State Change Reports, and never in response to Current State Reports.

- Multicast Listener Report: Sent by a host when it joins a multicast group, or in response to a Multicast Listener Query sent by the Querier. Hosts use these reports to indicate their current multicast listening state, or changes in the multicast listening state of their interfaces. These reports are of two types:
  - Current State Report: Contains the current Multicast Address Listening State of the host.
  - State Change Report: If the listening state of a host changes, the host immediately reports these changes through a State Change Report message. These reports contain either Filter Mode Change records and/or Source List Change records. State Change Reports are retransmitted several times to ensure all Mrouters receive it.
- Multicast Listener Done: Sent by a host when it wants to leave a multicast group. This message is sent to the link-scope all-routers IPv6 destination address of FF02::2. When an Mrouter receives a Multicast Listener Done message from the last member of the multicast address on a link, it stops forwarding traffic to this multicast address.

---

## How MLD Works

The software uses the information obtained through MLD to maintain a list of multicast group memberships for each interface and forwards the multicast traffic only to interested listeners.

Without MLD, the switch forwards IPv6 multicast traffic through all ports, increasing network load. Following is an overview of operations when MLD is configured on VFSM:

- The switch acts as an Mrouter when MLDv1/v2 is configured and enabled on each of its directly attached links. If the switch has multiple interfaces connected to the same link, it operates the protocol on any one of the interfaces.
- If there are multiple Mrouters on the subnet, the Mrouter with the numerically lowest IPv6 address is elected as the Querier.
- The Querier sends general queries at short intervals to learn multicast address listener information from an attached link.
- Hosts respond to these queries by reporting their per-interface Multicast Address Listening state, through Current State Report messages sent to a specific multicast address that all MLD routers on the link listen to.
- If the listening state of a host changes, the host immediately reports these changes through a State Change Report message.
- The Querier sends a Multicast Address Specific Query to verify if hosts are listening to a specified multicast address or not. Similarly, if MLDv2 is configured, the Querier sends a Multicast Address and Source Specific Query to verify, for a specified multicast address, if hosts are listening to a specific set of sources, or not. MLDv2 listener report messages consists of Multicast Address Records:
  - INCLUDE: to receive packets from source specified in the MLDv2 message
  - EXCLUDE: to receive packets from all sources except the ones specified in the MLDv2 message
- A host can send a State Change Report to indicate its desire to stop listening to a particular multicast address (or source in MLDv2). The Querier then sends a multicast address specific query to verify if there are other listeners of the multicast address. If there aren't any, the Mrouter deletes the multicast address from its Multicast Address Listener state and stops sending multicast traffic. Similarly in MLDv2, the Mrouter sends a Multicast Address and Source Specific Query to verify if, for a specified multicast address, there are hosts still listening to a specific set of sources.

VFSM supports MLD versions 1 and 2.

**Note:** MLDv2 operates in version 1 compatibility mode when, in a specific network, not all hosts are configured with MLDv2.

### How Flooding Impacts MLD

When `flood` option is disabled, the unknown multicast traffic is discarded if no Mrouters are learned on the switch. You can set the flooding behavior by configuring the `flood` and `cpu` options. You can optimize the flooding to ensure that unknown IP multicast (IPMC) data packets are not dropped during the learning phase.

The flooding options include:

- `flood`: Enable hardware flooding in VLAN for the unregistered IPMC; This option is enabled by default.

- `cpu`: Enable sending unregistered IPMC to the Mrouter ports. However, during the learning period, there will be some packet loss. The `cpu` option is enabled by default. You must ensure that the `flood` and `optflood` options are disabled.
- `optflood`: Enable optimized flooding to allow sending the unregistered IPMC to the Mrouter ports without having any packet loss during the learning period; This option is disabled by default; When `optflood` is enabled, the `flood` and `cpu` settings are ignored.

The flooding parameters must be configured per VLAN. Enter the following command to set the `flood` or `cpu` options:

```
VFSM(config)# vlan <vlan number>
VFSM(config-vlan)# [no] flood
VFSM(config-vlan)# [no] cpu
VFSM(config-vlan)# [no] optflood
```

## MLD Querier

An Mrouter acts as a Querier and periodically (at short query intervals) sends query messages in the subnet. If there are multiple Mrouters in the subnet, only one can be the Querier. All Mrouters on the subnet listen to the messages sent by the multicast address listeners, and maintain the same multicast listening information state.

All MLDv2 queries are sent with the FE80::/64 link-local source address prefix.

### Querier Election

Only one Mrouter can be the Querier per subnet. All other Mrouters will be non-Queriers. MLD versions 1 and 2 elect the Mrouter with the numerically lowest IPv6 address as the Querier.

If the switch is configured as an Mrouter on a subnet, it also acts as a Querier by default and sends multiple general queries. If the switch receives a general query from another Querier with a numerically lower IPv6 address, it sets the *other querier present timer* to the *other querier present timeout*, and changes its state to non-Querier. When the *other querier present timer* expires, it regains the Querier state and starts sending general queries.

**Note:** When MLD Querier is enabled on a VLAN, the switch performs the role of an MLD Querier only if it meets the MLD Querier election criteria.

## Dynamic Mrouters

The switch learns Mrouters on the ingress VLANs of the MLD-enabled interface. All report or done messages are forwarded to these Mrouters. By default, the option of dynamically learning Mrouters is disabled. To enable it, use the following commands:

```
VFSM(config)# interface ip <interface number>
VFSM(config-ip-if)# ipv6 mld dmrtr enable
```

---

## MLD Capacity and Default Values

Table 33 lists the maximum and minimum values of the VFSM variables.

Table 33. VFSM Capacity Table

Variable	Maximum Value
IPv6 Multicast Entries	256
IPv6 Interfaces for MLD	8

Table 34 lists the default settings for MLD features and variables.

Table 34. MLD Timers and Default Values

Field	Default Value
Robustness Variable (RV)	2
Query Interval (QI)	125 seconds
Query Response Interval (QRI)	10 seconds
Multicast Address Listeners Interval (MALI)	260 seconds [derived: $RV \cdot QI + QRI$ ]
Other Querier Present Interval [OQPT]	255 seconds [derived: $RV \cdot QI + \frac{1}{2} QRI$ ]
Start up Query Interval [SQI]	31.25 seconds [derived: $\frac{1}{4} \cdot QI$ ]
Startup Query Count [SQC]	2 [derived: RV]
Last Listener Query Interval [LLQI]	1 second
Last Listener Query Count [LLQC]	2 [derived: RV]
Last Listener Query Time [LLQT]	2 seconds [derived: $LLQI \cdot LLQT$ ]
Older Version Querier Present Timeout: [OVQPT]	260 seconds [derived: $RV \cdot QI + QRI$ ]
Older Version Host Present Interval [OVHPT]	260 seconds [derived: $RV \cdot QI + QRI$ ]

---

## Configuring MLD

Following are the steps to enable MLD and configure the interface parameters:

1. Turn on MLD globally.

```
VFSM(config)# ipv6 mld
VFSM(config-router-mlD)# enable
VFSM(config-router-mlD)# exit
```

2. Create an IPv6 interface.

```
VFSM(config)# interface ip 2
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# ipv6 address 2002:1:0:0:0:0:3
VFSM(config-ip-if)# ipv6 prefixlen 64
```

3. Enable MLD on the IPv6 interface.

```
VFSM(config-ip-if)# ipv6 mld enable
```

4. Configure the MLD parameters on the interface: version, robustness, query response interval, MLD query interval, and last listener query interval.

```
VFSM(config-ip-if)# ipv6 mld version <1-2> (MLD version)
VFSM(config-ip-if)# ipv6 mld robust <1-10> (Robustness)
VFSM(config-ip-if)# ipv6 mld gri <1-256> (In seconds)
VFSM(config-ip-if)# ipv6 mld qintrval <1-608>(In seconds)
VFSM(config-ip-if)# ipv6 mld llistnr <1-32> (In seconds)
```



---

## Chapter 25. Border Gateway Protocol

Border Gateway Protocol (BGP) is an Internet protocol that enables routers on an IPv4 network to share and advertise routing information with each other about the segments of the IPv4 address space they can access within their network and with routers on external networks. BGP allows you to decide what is the “best” route for a packet to take from your network to a destination on another network rather than simply setting a default route from your border router(s) to your upstream provider(s). BGP is defined in RFC 1771.

Virtual Fabric Switch Modules (VFSMs) can advertise their IP interfaces and IPv4 addresses using BGP and take BGP feeds from as many as 16 BGP router peers. This allows more resilience and flexibility in balancing traffic from the Internet.

**Note:** IBM Networking OS 7.8 does not support IPv6 for BGP.

The following topics are discussed in this section:

- [“Internal Routing Versus External Routing” on page 398](#)
- [“Forming BGP Peer Routers” on page 399](#)
- [“Loopback Interfaces” on page 400](#)
- [“What is a Route Map?” on page 401](#)
- [“Aggregating Routes” on page 404](#)
- [“Redistributing Routes” on page 405](#)
- [“BGP Attributes” on page 406](#)
- [“Selecting Route Paths in BGP” on page 407](#)
- [“BGP Failover Configuration” on page 408](#)
- [“Default Redistribution and Route Aggregation Example” on page 410](#)

---

## Internal Routing Versus External Routing

To ensure effective processing of network traffic, every router on your network needs to know how to send a packet (directly or indirectly) to any other location/destination in your network. This is referred to as *internal routing* and can be done with static routes or using active, internal dynamic routing protocols, such as RIP, RIPv2, and OSPF.

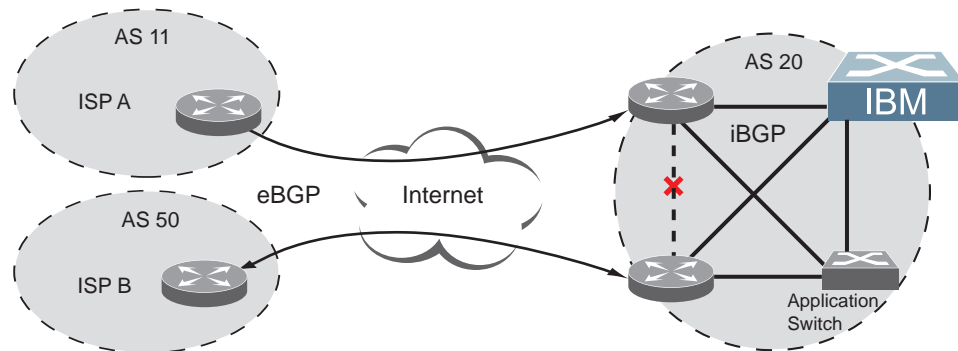
Static routes should have a higher degree of precedence than dynamic routing protocols. If the destination route is not in the route cache, then the packets are forwarded to the default gateway which may be incorrect if a dynamic routing protocol is enabled.

It is also useful to tell routers outside your network (upstream providers or *peers*) about the routes you can access in your network. External networks (those outside your own) that are under the same administrative control are referred to as *autonomous systems* (AS). Sharing of routing information between autonomous systems is known as *external routing*.

External BGP (eBGP) is used to exchange routes between different autonomous systems whereas internal BGP (iBGP) is used to exchange routes within the same autonomous system. An iBGP is a type of internal routing protocol you can use to do active routing inside your network. It also carries AS path information, which is important when you are an ISP or doing BGP transit.

The iBGP peers have to maintain reciprocal sessions to every other iBGP router in the same AS (in a full-mesh manner) in order to propagate route information throughout the AS. If the iBGP session shown between the two routers in AS 20 was not present (as indicated in [Figure 38](#)), the top router would not learn the route to AS 50, and the bottom router would not learn the route to AS 11, even though the two AS 20 routers are connected via the BladeCenter and the Application Switch.

Figure 38. iBGP and eBGP



Typically, an AS has one or more *border routers*—peer routers that exchange routes with other ASs—and an internal routing scheme that enables routers in that AS to reach every other router and destination within that AS. When you *advertise* routes to border routers on other autonomous systems, you are effectively committing to carry data to the IPv4 space represented in the route being advertised. For example, if you advertise 192.204.4.0/24, you are declaring that if another router sends you data destined for any address in 192.204.4.0/24, you know how to carry that data to its destination.

---

## Forming BGP Peer Routers

Two BGP routers become peers or neighbors once you establish a TCP connection between them. For each new route, if a peer is interested in that route (for example, if a peer would like to receive your static routes and the new route is static), an update message is sent to that peer containing the new route. For each route removed from the route table, if the route has already been sent to a peer, an update message containing the route to withdraw is sent to that peer.

For each Internet host, you must be able to send a packet to that host, and that host has to have a path back to you. This means that whoever provides Internet connectivity to that host must have a path to you. Ultimately, this means that they must “hear a route” which covers the section of the IPv4 space you are using; otherwise, you will not have connectivity to the host in question.

---

## Loopback Interfaces

In many networks, multiple connections may exist between network devices. In such environments, it may be useful to employ a loopback interface for a common BGP router address, rather than peering the switch to each individual interface.

When a loopback interface is created for BGP, the switch automatically uses the loopback interface as the BGP peer ID, instead of the switch's local IP interface address.

**Note:** To ensure that the loopback interface is reachable from peer devices, it should be advertised using an interior routing protocol (such as OSPF), or a static route should be configured on the peer.

To configure an existing loopback interface for BGP neighbor use the following commands:

```
VFSM(config)# router bgp
VFSM(config-router-bgp)# neighbor <#> update-source loopback <1-5>
VFSM(config-router-bgp)# exit
```

---

## What is a Route Map?

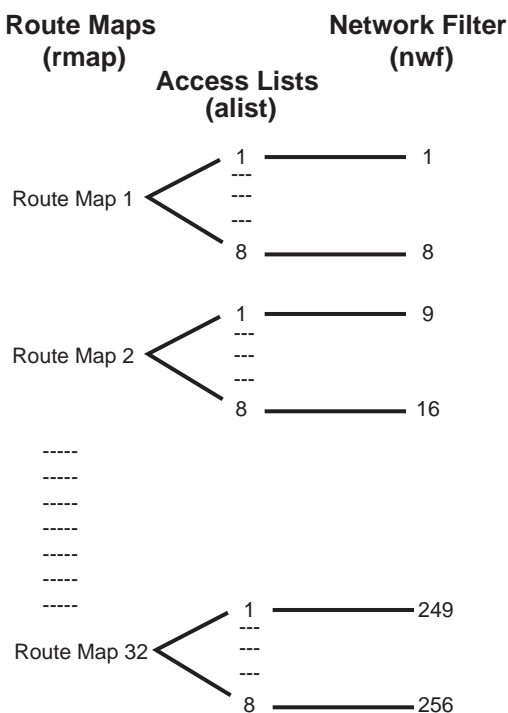
A route map is used to control and modify routing information. Route maps define conditions for redistributing routes from one routing protocol to another or controlling routing information when injecting it in and out of BGP. For example, a route map is used to set a preference value for a specific route from a peer router and another preference value for all other routes learned via the same peer router. For example, the following commands are used to define a route map:

VFSM(config)# route-map <map number>	(Select a route map)
VFSM(config-route-map)# ?	(List available commands)

A route map allows you to match attributes, such as metric, network address, and AS number. It also allows users to overwrite the local preference metric and to append the AS number in the AS route. See [“BGP Failover Configuration” on page 408](#).

IBM N/OS allows you to configure 32 route maps. Each route map can have up to eight access lists. Each access list consists of a network filter. A network filter defines an IPv4 address and subnet mask of the network that you want to include in the filter. [Figure 39](#) illustrates the relationship between route maps, access lists and network filters.

Figure 39. Distributing Network Filters in Access Lists and Route Maps



## Incoming and Outgoing Route Maps

You can have two types of route maps: incoming and outgoing. A BGP peer router can be configured to support up to eight route maps in the incoming route map list and outgoing route map list.

If a route map is not configured in the incoming route map list, the router imports all BGP updates. If a route map is configured in the incoming route map list, the router ignores all unmatched incoming updates. If you set the action to `deny`, you must add another route map to permit all unmatched updates.

Route maps in an outgoing route map list behave similar to route maps in an incoming route map list. If a route map is not configured in the outgoing route map list, all routes are advertised or permitted. If a route map in the outgoing route map list is set to `permit`, matched routes are advertised and unmatched routes are ignored.

## Precedence

You can set a priority to a route map by specifying a precedence value with the following commands:

```
VFSM(config)# route-map <map number>           (Select a route map)
VFSM(config-route-map)# precedence <1-255>      (Specify a precedence)
VFSM(config-route-map)# exit
```

The smaller the value the higher the precedence. If two route maps have the same precedence value, the smaller number has higher precedence.

## Configuration Overview

To configure route maps, you need to do the following:

1. Define network filter.

```
VFSM(config)# ip match-address 1 <IPv4 address> <IPv4 subnet mask>
VFSM(config)# ip match-address 1 enable
```

1. Enter a filter number from 1 to 256. Specify the IPv4 address and subnet mask of the network that you want to match. Enable the network filter. You can distribute up to 256 network filters among 32 route maps each containing eight access lists.
2. (Optional) Define the criteria for the access list and enable it.

Specify the access list and associate the network filter number configured in Step 1.

```
VFSM(config)# route-map 1
VFSM(config-route-map)# access-list 1 match-address 1
VFSM(config-route-map)# access-list 1 metric <metric value>
VFSM(config-route-map)# access-list 1 action deny
VFSM(config-route-map)# access-list 1 enable
```

Steps 2 and 3 are optional, depending on the criteria that you want to match. In Step 2, the network filter number is used to match the subnets defined in the network filter. In Step 3, the autonomous system number is used to match the subnets. Or, you can use both (Step 2 and Step 3) criteria: access list (network filter) and access path (AS filter) to configure the route maps.

3. (Optional) Configure the attributes in the AS filter menu.

```
VFSM(config-route-map)# as-path-list 1 as 1
VFSM(config-route-map)# as-path-list 1 action deny
VFSM(config-route-map)# as-path-list 1 enable
```

4. Set up the BGP attributes.

If you want to overwrite the attributes that the peer router is sending, define the following BGP attributes:

- Specify up to three AS numbers that you want to prepend to a matched route and the local preference for the matched route.
- Specify the metric [Multi Exit Discriminator (MED)] for the matched route.

```
VFSM(config-route-map)# as-path-preference <AS number> [<AS number>] [<AS number>]
VFSM(config-route-map)# local-preference <local preference value>
VFSM(config-route-map)# metric <metric value>
```

5. Enable the route .

```
VFSM(config-route-map)# enable
VFSM(config-route-map)# exit
```

6. Turn BGP on.

```
VFSM(config)# router bgp
VFSM(config-router-bgp)# enable
```

7. Assign the route map to a peer router.

Select the peer router and then add the route map to the incoming route map list,

```
VFSM(config-router-bgp)# neighbor 1 route-map in <1-32>
```

or to the outgoing route map list.

```
VFSM(config-router-bgp)# neighbor 1 route-map out <1-32>
```

8. Exit Router BGP mode.

```
VFSM(config-router-bgp)# exit
```

---

## Aggregating Routes

Aggregation is the process of combining several different routes in such a way that a single route can be advertised, which minimizes the size of the routing table. 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.

To define an aggregate route in the BGP routing table, use the following commands:

```
VFSM(config)# router bgp
VFSM(config-router-bgp)# aggregate-address <1-16> <IPv4 address> <mask>
VFSM(config-router-bgp)# aggregate-address <1-16> enable
VFSM(config-router-bgp)# exit
```

An example of creating a BGP aggregate route is shown in [“Default Redistribution and Route Aggregation Example” on page 410](#).

---

## Redistributing Routes

In addition to running multiple routing protocols simultaneously, N/OS software can redistribute information from one routing protocol to another. For example, you can instruct the switch to use BGP to re-advertise static routes. This applies to all of the IP-based routing protocols.

You can also conditionally control the redistribution of routes between routing domains by defining a method known as route maps between the two domains. For more information on route maps, see [“What is a Route Map?” on page 401](#). Redistributing routes is another way of providing policy control over whether to export OSPF routes, fixed routes, and static routes. For an example configuration, see [“Default Redistribution and Route Aggregation Example” on page 410](#).

Default routes can be configured using the following methods:

- Import
- Originate—The router sends a default route to peers if it does not have any default routes in its routing table.
- Redistribute—Default routes are either configured through the default gateway or learned via other protocols and redistributed to peer routers. If the default routes are from the default gateway, enable the static routes because default routes from the default gateway are static routes. Similarly, if the routes are learned from another routing protocol, make sure you enable that protocol for redistribution.
- None

---

## BGP Attributes

The following two BGP attributes are discussed in this section: Local preference and metric (Multi-Exit Discriminator).

### Local Preference Attribute

When there are multiple paths to the same destination, the local preference attribute indicates the preferred path. The path with the higher preference is preferred (the default value of the local preference attribute is 100). Unlike the weight attribute, which is only relevant to the local router, the local preference attribute is part of the routing update and is exchanged among routers in the same AS.

The local preference attribute can be set in one of two ways:

- Using the BGP default local preference method, affecting the outbound direction only.

```
VFSM(config)# router bgp
VFSM(config_router_bgp)# local-preference
VFSM(config_router_bgp)# exit
```

- Using the route map local preference method, which affects both inbound and outbound directions.

```
VFSM(config)# route-map 1
VFSM(config_route_map)# local-preference
VFSM(config_router_map)# exit
```

### Metric (Multi-Exit Discriminator) Attribute

This attribute is a hint to external neighbors about the preferred path into an AS when there are multiple entry points. A lower metric value is preferred over a higher metric value. The default value of the metric attribute is 0.

Unlike local preference, the metric attribute is exchanged between ASs; however, a metric attribute that comes into an AS does not leave the AS.

When an update enters the AS with a certain metric value, that value is used for decision making within the AS. When BGP sends that update to another AS, the metric is reset to 0.

Unless otherwise specified, the router compares metric attributes for paths from external neighbors that are in the same AS.

---

## Selecting Route Paths in BGP

BGP selects only one path as the best path. It does not rely on metric attributes to determine the best path. When the same network is learned via more than one BGP peer, BGP uses its policy for selecting the best route to that network. The BGP implementation on the VFSM uses the following criteria to select a path when the same route is received from multiple peers.

1. Local fixed and static routes are preferred over learned routes.
2. With iBGP peers, routes with higher local preference values are selected.
3. In the case of multiple routes of equal preference, the route with lower AS path weight is selected.

AS path weight =  $128 \times \text{AS path length (number of autonomous systems traversed)}$ .

4. In the case of equal weight and routes learned from peers that reside in the same AS, the lower metric is selected.

**Note:** A route with a metric is preferred over a route without a metric.

5. The lower cost to the next hop of routes is selected.
6. In the case of equal cost, the eBGP route is preferred over iBGP.
7. If all routes have same route type (eBGP or iBGP), the route with the lower router ID is selected.

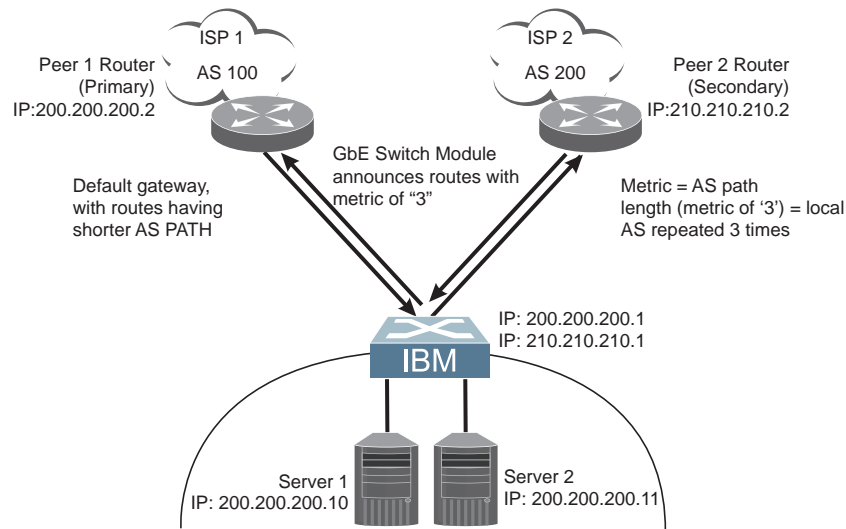
When the path is selected, BGP puts the selected path in its routing table and propagates the path to its neighbors.

## BGP Failover Configuration

Use the following example to create redundant default gateways for a VFSM at a Web Host/ISP site, eliminating the possibility, should one gateway go down, that requests will be forwarded to an upstream router unknown to the switch.

As shown in [Figure 40](#), the switch is connected to ISP 1 and ISP 2. The customer negotiates with both ISPs to allow the switch to use their peer routers as default gateways. The ISP peer routers will then need to announce themselves as default gateways to the VFSM.

Figure 40. BGP Failover Configuration Example



On the VFSM, one peer router (the secondary one) is configured with a longer AS path than the other, so that the peer with the shorter AS path will be seen by the switch as the primary default gateway. ISP 2, the secondary peer, is configured with a metric of "3," thereby appearing to the switch to be three router *hops* away.

### 1. Define the VLANs.

For simplicity, both default gateways are configured in the same VLAN in this example. The gateways could be in the same VLAN or different VLANs.

```
VFSM(config)# vlan 1
```

### 2. Define the IP interfaces with IPv4 addresses.

The switch will need an IP interface for each default gateway to which it will be connected. Each interface must be placed in the appropriate VLAN. These interfaces will be used as the primary and secondary default gateways for the switch.

```
VFSM(config)# interface ip 1 address 200.200.200.1 255.255.255.0 enable
VFSM(config-ip-if)# exit

VFSM(config)# interface ip 2 address 210.210.210.1 255.255.255.0 enable
VFSM(config-ip-if)# exit
```

3. Enable IP forwarding.

IP forwarding is turned on by default and is used for VLAN-to-VLAN (non-BGP) routing. Make sure IP forwarding is on if the default gateways are on different subnets or if the switch is connected to different subnets and those subnets need to communicate through the switch (which they almost always do).

```
VFSM(config)# ip routing
```

*(Enable IP forwarding)*

**Note:** To help eliminate the possibility for a Denial of Service (DoS) attack, the forwarding of directed broadcasts is disabled by default.

4. Configure BGP peer router 1 and 2.

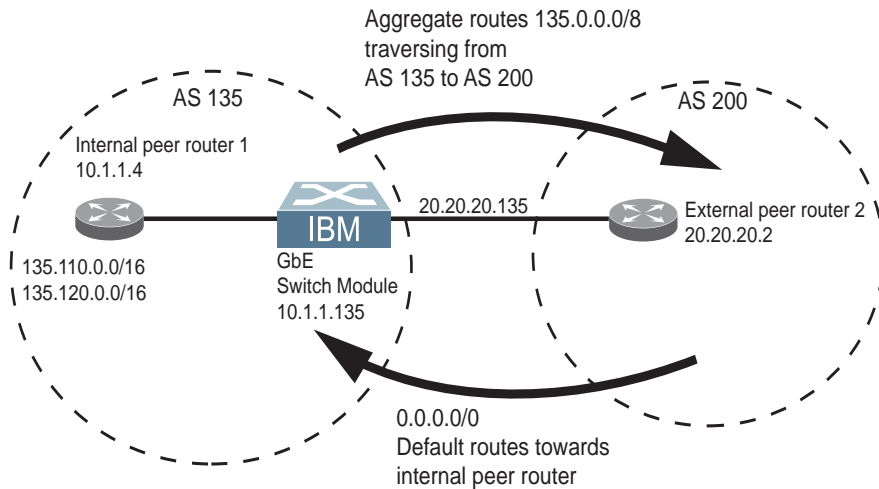
```
VFSM(config)# router bgp
VFSM(config-router-bgp)# neighbor 1 remote-address 200.200.200.2
VFSM(config-router-bgp)# neighbor 1 remote-as 100
VFSM(config-router-bgp)# neighbor 2 remote-address 210.210.210.2
VFSM(config-router-bgp)# neighbor 2 remote-as 200
```

## Default Redistribution and Route Aggregation Example

This example shows you how to configure the switch to redistribute information from one routing protocol to another and create an aggregate route entry in the BGP routing table to minimize the size of the routing table.

As illustrated in [Figure 41](#), you have two peer routers: an internal and an external peer router. Configure the VFSM to redistribute the default routes from AS 200 to AS 135. At the same time, configure for route aggregation to allow you to condense the number of routes traversing from AS 135 to AS 200.

Figure 41. Route Aggregation and Default Route Redistribution



1. Configure the IP interface.
2. Configure the AS number (AS 135) and router ID number (10.1.1.135).

```
VFSM(config)# router bgp
VFSM(config-router-bgp)# as 135
VFSM(config-router-bgp)# exit
VFSM(config)# ip router-id 10.1.1.135
```

3. Configure internal peer router 1 and external peer router 2.

```
VFSM(config)# router bgp
VFSM(config-router-bgp)# neighbor 1 remote-address 10.1.1.4
VFSM(config-router-bgp)# neighbor 1 remote-as 135
VFSM(config-router-bgp)# neighbor 2 remote-address 20.20.20.2
VFSM(config-router-bgp)# neighbor 2 remote-as 200
```

4. Configure redistribution for Peer 1.

```
VFSM(config-router-bgp)# neighbor 1 redistribute default-action redistribute
VFSM(config-router-bgp)# neighbor 1 redistribute fixed
```

5. Configure aggregation policy control.

Configure the routes that you want aggregated.

```
VFSM(config-router-bgp)# aggregate-address 1 135.0.0.0 255.0.0.0  
VFSM(config-router-bgp)# aggregate-address 1 enable
```



---

## Chapter 26. OSPF

IBM Networking OS supports the Open Shortest Path First (OSPF) routing protocol. The IBM N/OS implementation conforms to the OSPF version 2 specifications detailed in Internet RFC 1583, and OSPF version 3 specifications in RFC 5340. The following sections discuss OSPF support for the Virtual Fabric Switch Module (VFSM):

- [“OSPFv2 Overview” on page 414](#). This section provides information on OSPFv2 concepts, such as types of OSPF areas, types of routing devices, neighbors, adjacencies, link state database, authentication, and internal versus external routing.
- [“OSPFv2 Implementation in IBM N/OS” on page 418](#). This section describes how OSPFv2 is implemented in N/OS, such as configuration parameters, electing the designated router, summarizing routes, defining route maps and so forth.
- [“OSPFv2 Configuration Examples” on page 427](#). This section provides step-by-step instructions on configuring different OSPFv2 examples:
  - Creating a simple OSPF domain
  - Creating virtual links
  - Summarizing routes
- [“OSPFv3 Implementation in IBM N/OS” on page 435](#). This section describes differences and additional features found in OSPFv3.

---

## OSPFv2 Overview

OSPF is designed for routing traffic within a single IP domain called an Autonomous System (AS). The AS can be divided into smaller logical units known as *areas*.

All routing devices maintain link information in their own Link State Database (LSDB). The LSDB for all routing devices within an area is identical but is not exchanged between different areas. Only routing updates are exchanged between areas, thereby significantly reducing the overhead for maintaining routing information on a large, dynamic network.

The following sections describe key OSPF concepts.

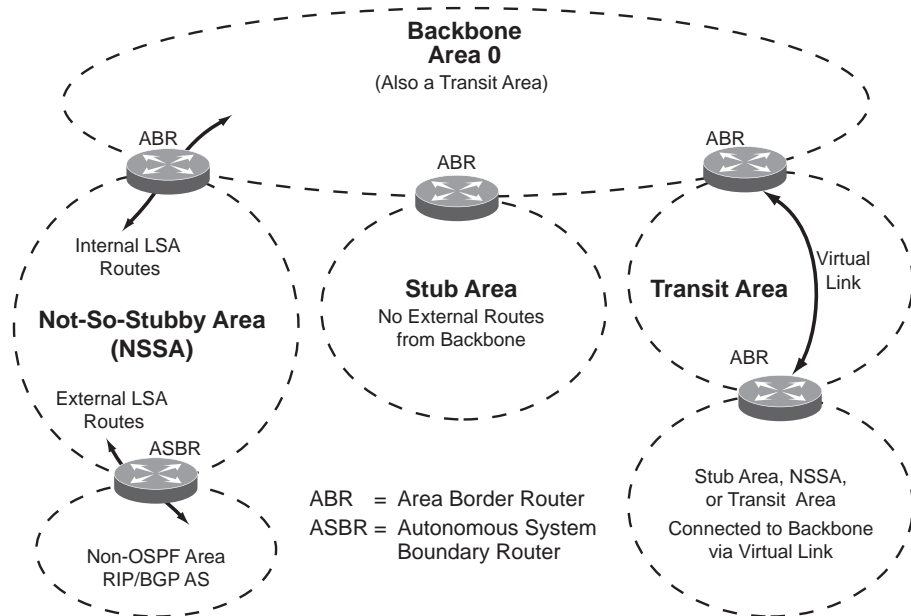
### Types of OSPF Areas

An AS can be broken into logical units known as *areas*. In any AS with multiple areas, one area must be designated as area 0, known as the *backbone*. The backbone acts as the central OSPF area. All other areas in the AS must be connected to the backbone. Areas inject summary routing information into the backbone, which then distributes it to other areas as needed.

As shown in [Figure 42](#), OSPF defines the following types of areas:

- **Stub Area**—an area that is connected to only one other area. External route information is not distributed into stub areas.
- **Not-So-Stubby-Area (NSSA)**—similar to a stub area with additional capabilities. Routes originating from within the NSSA can be propagated to adjacent transit and backbone areas. External routes from outside the AS can be advertised within the NSSA but can be configured to not be distributed into other areas.
- **Transit Area**—an area that carries data traffic which neither originates nor terminates in the area itself.

Figure 42. OSPF Area Types

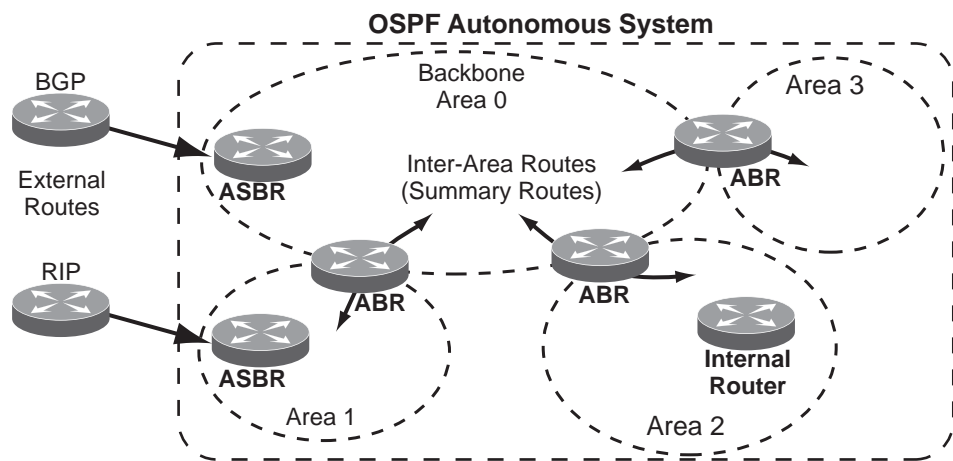


## Types of OSPF Routing Devices

As shown in [Figure 43](#), OSPF uses the following types of routing devices:

- Internal Router (IR)—a router that has all of its interfaces within the same area. IRs maintain LSDBs identical to those of other routing devices within the local area.
- Area Border Router (ABR)—a router that has interfaces in multiple areas. ABRs maintain one LSDB for each connected area and disseminate routing information between areas.
- Autonomous System Boundary Router (ASBR)—a router that acts as a gateway between the OSPF domain and non-OSPF domains, such as RIP, BGP, and static routes.

Figure 43. OSPF Domain and an Autonomous System



## Neighbors and Adjacencies

In areas with two or more routing devices, *neighbors* and *adjacencies* are formed.

*Neighbors* are routing devices that maintain information about each others' health. To establish neighbor relationships, routing devices periodically send hello packets on each of their interfaces. All routing devices that share a common network segment, appear in the same area, and have the same health parameters (`hello` and `dead` intervals) and authentication parameters respond to each other's hello packets and become neighbors. Neighbors continue to send periodic hello packets to advertise their health to neighbors. In turn, they listen to hello packets to determine the health of their neighbors and to establish contact with new neighbors.

The hello process is used for electing one of the neighbors as the area's Designated Router (DR) and one as the area's Backup Designated Router (BDR). The DR is adjacent to all other neighbors and acts as the central contact for database exchanges. Each neighbor sends its database information to the DR, which relays the information to the other neighbors.

The BDR is adjacent to all other neighbors (including the DR). Each neighbor sends its database information to the BDR just as with the DR, but the BDR merely stores this data and does not distribute it. If the DR fails, the BDR will take over the task of distributing database information to the other neighbors.

## The Link-State Database

OSPF is a link-state routing protocol. A *link* represents an interface (or routable path) from the routing device. By establishing an adjacency with the DR, each routing device in an OSPF area maintains an identical Link-State Database (LSDB) describing the network topology for its area.

Each routing device transmits a Link-State Advertisement (LSA) on each of its *active* interfaces. LSAs are entered into the LSDB of each routing device. OSPF uses *flooding* to distribute LSAs between routing devices. Interfaces may also be *passive*. Passive interfaces send LSAs to active interfaces, but do not receive LSAs, hello packets, or any other OSPF protocol information from active interfaces. Passive interfaces behave as stub networks, allowing OSPF routing devices to be aware of devices that do otherwise participate in OSPF (either because they do not support it, or because the administrator chooses to restrict OSPF traffic exchange or transit).

When LSAs result in changes to the routing device's LSDB, the routing device forwards the changes to the adjacent neighbors (the DR and BDR) for distribution to the other neighbors.

OSPF routing updates occur only when changes occur, instead of periodically. For each new route, if an adjacency is interested in that route (for example, if configured to receive static routes and the new route is indeed static), an update message containing the new route is sent to the adjacency. For each route removed from the route table, if the route has already been sent to an adjacency, an update message containing the route to withdraw is sent.

## The Shortest Path First Tree

The routing devices use a link-state algorithm (Dijkstra's algorithm) to calculate the shortest path to all known destinations, based on the cumulative cost required to reach the destination.

The cost of an individual interface in OSPF is an indication of the overhead required to send packets across it. The cost is inversely proportional to the bandwidth of the interface. A lower cost indicates a higher bandwidth.

## Internal Versus External Routing

To ensure effective processing of network traffic, every routing device on your network needs to know how to send a packet (directly or indirectly) to any other location/destination in your network. This is referred to as *internal routing* and can be done with static routes or using active internal routing protocols, such as OSPF, RIP, or RIPv2.

It is also useful to tell routers outside your network (upstream providers or *peers*) about the routes you have access to in your network. Sharing of routing information between autonomous systems is known as *external routing*.

Typically, an AS will have one or more border routers (peer routers that exchange routes with other OSPF networks) as well as an internal routing system enabling every router in that AS to reach every other router and destination within that AS.

When a routing device *advertises* routes to boundary routers on other autonomous systems, it is effectively committing to carry data to the IP space represented in the route being advertised. For example, if the routing device advertises 192.204.4.0/24, it is declaring that if another router sends data destined for any address in the 192.204.4.0/24 range, it will carry that data to its destination.

---

## OSPFv2 Implementation in IBM N/OS

N/OS supports a single instance of OSPF and up to 4K routes on the network. The following sections describe OSPF implementation in N/OS:

- [“Configurable Parameters” on page 418](#)
- [“Defining Areas” on page 419](#)
- [“Interface Cost” on page 421](#)
- [“Electing the Designated Router and Backup” on page 421](#)
- [“Summarizing Routes” on page 421](#)
- [“Default Routes” on page 422](#)
- [“Virtual Links” on page 422](#)
- [“Router ID” on page 423](#)
- [“Authentication” on page 423](#)

### Configurable Parameters

In N/OS, OSPF parameters can be configured through the Command Line Interfaces (CLI/ISCLI), Browser-Based Interface (BBI), or through SNMP. For more information, see [“Switch Administration” on page 27.](#)

The CLI supports the following parameters: interface output cost, interface priority, dead and hello intervals, retransmission interval, and interface transmit delay.

In addition to the above parameters, you can also specify the following:

- Shortest Path First (SPF) interval—Time interval between successive calculations of the shortest path tree using the Dijkstra’s algorithm.
- Stub area metric—A stub area can be configured to send a numeric metric value such that all routes received via that stub area carry the configured metric to potentially influence routing decisions.
- Default routes—Default routes with weight metrics can be manually injected into transit areas. This helps establish a preferred route when multiple routing devices exist between two areas. It also helps route traffic to external networks.
- Passive—When enabled, the interface sends LSAs to upstream devices, but does not otherwise participate in OSPF protocol exchanges.
- Point-to-Point—For LANs that have only two OSPF routing agents (the VFSM and one other device), this option allows the switch to significantly reduce the amount of routing information it must carry and manage.

## Defining Areas

If you are configuring multiple areas in your OSPF domain, one of the areas must be designated as area 0, known as the *backbone*. The backbone is the central OSPF area and is usually physically connected to all other areas. The areas inject routing information into the backbone which, in turn, disseminates the information into other areas.

Since the backbone connects the areas in your network, it must be a contiguous area. If the backbone is partitioned (possibly as a result of joining separate OSPF networks), parts of the AS will be unreachable, and you will need to configure *virtual links* to reconnect the partitioned areas (see [“Virtual Links” on page 422](#)).

Up to three OSPF areas can be connected to the VFSM with N/OS software. To configure an area, the OSPF number must be defined and then attached to a network interface on the switch. The full process is explained in the following sections.

An OSPF area is defined by assigning **two** pieces of information: an *area index* and an *area ID*. The commands to define and enable an OSPF area are as follows:

```
VFSM(config)# router ospf
VFSM(config-router-ospf)# area <area index> area-id <n.n.n.n>
VFSM(config-router-ospf)# area <area index> enable
VFSM(config-router-ospf)# exit
```

The *area <area index>* option above is an arbitrary index used only on the switch and does not represent the actual OSPF area number. The actual OSPF area number is defined in the *area-id* portion of the command as explained in the following sections.

## Assigning the Area Index

The *area <area index>* option is actually just an arbitrary index (0–2) used only by the VFSM. This index does not necessarily represent the OSPF area number, though for configuration simplicity, it should where possible.

For example, both of the following sets of commands define OSPF area 0 (the backbone) and area 1 because that information is held in the area ID portion of the command. However, the first set of commands is easier to maintain because the arbitrary area indexes agree with the area IDs:

- Area index and area ID agree

```
area 0 area-id 0.0.0.0
```

*(Use index 0 to set area 0 in ID octet format)*

```
area 1 area-id 0.0.0.1
```

*(Use index 1 to set area 1 in ID octet format)*

- Area index set to an arbitrary value

```
area 1 area-id 0.0.0.0
```

*(Use index 1 to set area 0 in ID octet format)*

```
area 2 area-id 0.0.0.1
```

*(Use index 2 to set area 1 in ID octet format)*

## Using the Area ID to Assign the OSPF Area Number

The OSPF area number is defined in the `areaid <IP address>` option. The octet format is used to be compatible with two different systems of notation used by other OSPF network vendors. There are two valid ways to designate an area ID:

- Single Number

Most common OSPF vendors express the area ID number as a single number. For example, the Cisco IOS-based router command “`network 1.1.1.0 0.0.0.255 area 1`” defines the area number simply as “area 1.”

- Multi-octet (*IP address*): Placing the area number in the last octet (0.0.0.*n*)

Some OSPF vendors express the area ID number in multi-octet format. For example, “area 0.0.0.2” represents OSPF area 2 and can be specified directly on the VFSM as “area-id 0.0.0.2”.

On the VFSM, using the last octet in the area ID, “area 1” is equivalent to “area-id 0.0.0.1”.

**Note:** Although both types of area ID formats are supported, be sure that the area IDs are in the same format throughout an area.

## Attaching an Area to a Network

Once an OSPF area has been defined, it must be associated with a network. To attach the area to a network, you must assign the OSPF area index to an IP interface that participates in the area. The format for the command is as follows:

```
VFSM(config)# interface ip <interface number>
VFSM(config-ip-if)# ip ospf area <area index>
VFSM(config-ip-if)# exit
```

For example, the following commands could be used to configure IPv4 interface 14 for a presence on the IPv4 10.10.10.1/24 network, to define OSPF area 1, and to attach the area to the network:

```
VFSM(config)# router ospf
VFSM(config-router-ospf)# area 1 area-id 0.0.0.1
VFSM(config-router-ospf)# area 1 enable
VFSM(config-router-ospf)# enable
VFSM(config-router-ospf)# exit
VFSM(config)# interface ip 14
VFSM(config-ip-if)# ip address 10.10.10.1 255.255.255.0 enable
VFSM(config-ip-if)# ip ospf area 1
VFSM(config-ip-if)# ip ospf enable
```

**Note:** OSPFv2 supports IPv4 only. IPv6 is supported in OSPFv3 (see [“OSPFv3 Implementation in IBM N/OS” on page 435](#)).

## Interface Cost

The OSPF link-state algorithm (Dijkstra's algorithm) places each routing device at the root of a tree and determines the cumulative *cost* required to reach each destination. Usually, the cost is inversely proportional to the bandwidth of the interface. Low cost indicates high bandwidth. You can manually enter the cost for the output route with the following command:

```
VFSM(config-ip-if)# ip ospf cost <cost value (1-65535)>
```

## Electing the Designated Router and Backup

In any area with more than two routing devices, a Designated Router (DR) is elected as the central contact for database exchanges among neighbors, and a Backup Designated Router (BDR) is elected in case the DR fails.

DR and BDR elections are made through the hello process. The election can be influenced by assigning a priority value to the OSPF interfaces on the VFSM. The command is as follows:

```
VFSM(config-ip-if)# ip ospf priority <priority value (0-255)>
```

A priority value of 255 is the highest, and 1 is the lowest. A priority value of 0 specifies that the interface cannot be used as a DR or BDR. In case of a tie, the routing device with the highest router ID wins. Interfaces configured as *passive* do not participate in the DR or BDR election process:

```
VFSM(config-ip-if)# ip ospf passive-interface  
VFSM(config-ip-if)# exit
```

## Summarizing Routes

Route summarization condenses routing information. Without summarization, each routing device in an OSPF network would retain a route to every subnet in the network. With summarization, routing devices can reduce some sets of routes to a single advertisement, reducing both the load on the routing device and the perceived complexity of the network. The importance of route summarization increases with network size.

Summary routes can be defined for up to 16 IP address ranges using the following command:

```
VFSM(config)# router ospf  
VFSM(config-router-ospf)# area-range <range number> address <IP address> <mask>
```

where <range number> is a number 1 to 16, <IPv4 address> is the base IP address for the range, and <subnet mask> is the IPv4 address mask for the range. For a detailed configuration example, see [“Example 3: Summarizing Routes” on page 433](#).

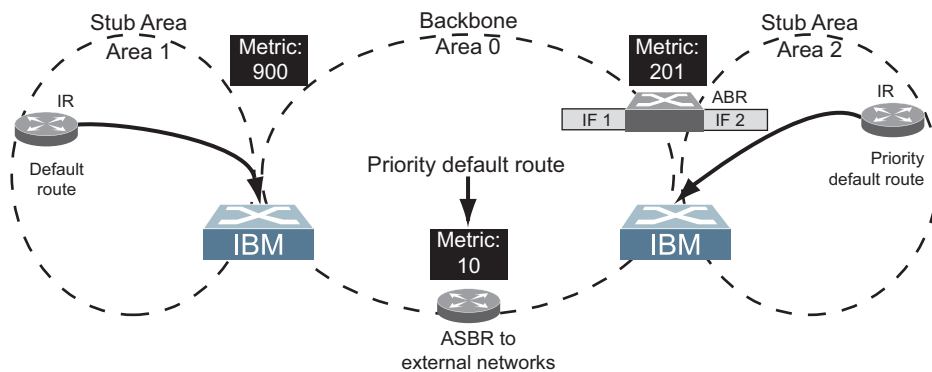
**Note:** OSPFv2 supports IPv4 only. IPv6 is supported in OSPFv3 (see [“OSPFv3 Implementation in IBM N/OS” on page 435](#)).

## Default Routes

When an OSPF routing device encounters traffic for a destination address it does not recognize, it forwards that traffic along the *default route*. Typically, the default route leads upstream toward the backbone until it reaches the intended area or an external router.

Each VFSM acting as an ABR automatically inserts a default route into each attached area. In simple OSPF stub areas or NSSAs with only one ABR leading upstream (see Area 1 in [Figure 44](#)), any traffic for IP address destinations outside the area is forwarded to the switch's IP interface, and then into the connected transit area (usually the backbone). Since this is automatic, no further configuration is required for such areas.

Figure 44. Injecting Default Routes



If the switch is in a transit area and has a configured default gateway, it can inject a default route into rest of the OSPF domain. Use the following command to configure the switch to inject OSPF default routes:

```
VFSM(config-router-ospf)# default-information <metric value> <metric type (1 or 2)>
```

In the preceding command, *<metric value>* sets the priority for choosing this switch for default route. The value `none` sets no default and 1 sets the highest priority for default route. Metric type determines the method for influencing routing decisions for external routes.

When the switch is configured to inject a default route, an AS-external LSA with link state ID 0.0.0.0 is propagated throughout the OSPF routing domain. This LSA is sent with the configured metric value and metric type.

The OSPF default route configuration can be removed with the command:

```
VFSM(config-router-ospf)# no default-information
```

## Virtual Links

Usually, all areas in an OSPF AS are physically connected to the backbone. In some cases where this is not possible, you can use a *virtual link*. Virtual links are created to connect one area to the backbone through another non-backbone area (see [Figure 42 on page 414](#)).

The area which contains a virtual link must be a transit area and have full routing information. Virtual links cannot be configured inside a stub area or NSSA. The area type must be defined as `transit` using the following command:

```
VFSM(config-router-ospf)# area <area index> type transit
```

The virtual link must be configured on the routing devices at each endpoint of the virtual link, though they may traverse multiple routing devices. To configure a VFSM as one endpoint of a virtual link, use the following command:

```
VFSM(config-router-ospf)# area-virtual-link <link number> neighbor-router <router ID>
```

where *<link number>* is a value between 1 and 3, *<area index>* is the OSPF area index of the transit area, and *<router ID>* is the IP address of the virtual neighbor (nbr), the routing device at the target endpoint. Another router ID is needed when configuring a virtual link in the other direction. To provide the VFSM with a router ID, see the following section, [Router ID](#).

For a detailed configuration example on Virtual Links, see [“Example 2: Virtual Links” on page 430](#).

## Router ID

Routing devices in OSPF areas are identified by a router ID, expressed in IP address format. The router ID is not required to be part of any IP interface range or in any OSPF area, and may even use the VFSM loopback interface (see [“Loopback Interfaces in OSPF” on page 426](#)).

The router ID can be configured in one of the following two ways:

- Dynamically (the default)—OSPF protocol configures the router ID as the lowest IP loopback interface IP address, if available, or else the lowest IP interface IP address, if available. Once dynamically configured, the router ID does not normally undergo further updates.
- Statically—Use the following command to manually configure the router ID:

```
VFSM(config-router-ospf)# ip router-id <IPv4 address>
```

To change the router ID from static to dynamic, set the router ID to 0.0.0.0, save the configuration, and reboot the VFSM. To view the router ID, enter:

```
VFSM(config-router-ospf)# show ip ospf
```

## Authentication

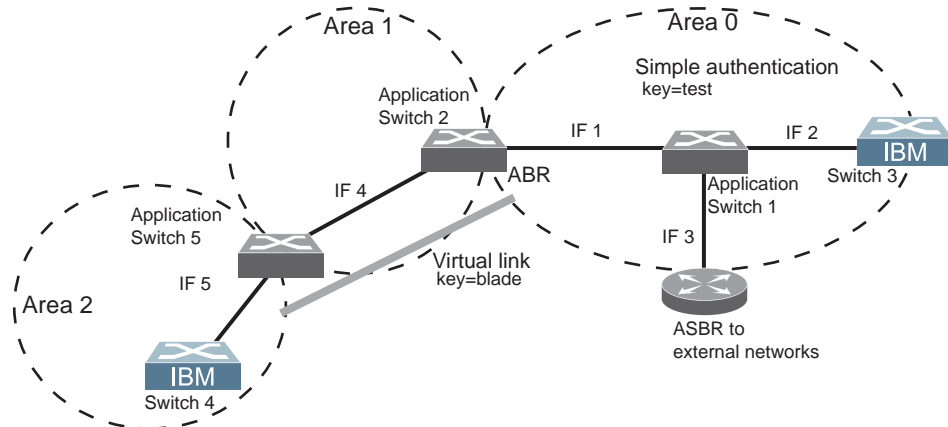
OSPF protocol exchanges can be authenticated so that only trusted routing devices can participate. This ensures less processing on routing devices that are not listening to OSPF packets.

OSPF allows packet authentication and uses IP multicast when sending and receiving packets. Routers participate in routing domains based on pre-defined passwords. N/OS supports simple password (type 1 plain text passwords) and MD5 cryptographic authentication. This type of authentication allows a password to be configured per area.

We strongly recommend that you implement MD5 cryptographic authentication as a best practice.

Figure 45 shows authentication configured for area 0 with the password `test`. Simple authentication is also configured for the virtual link between area 2 and area 0. Area 1 is not configured for OSPF authentication.

Figure 45. OSPF Authentication



## Configuring Plain Text OSPF Passwords

To configure plain text OSPF passwords as shown in Figure 45 use the following commands:

1. Enable OSPF authentication for Area 0 on switches 1, 2, and 3.

```
VFSM(config-router-ospf)# area 0 authentication-type password
VFSM(config-router-ospf)# exit
```

2. Configure a simple text password up to eight characters for each OSPF IP interface in Area 0 on switches 1, 2, and 3.

```
VFSM(config)# interface ip 1
VFSM(config-ip-if)# ip ospf key test
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip ospf key test
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 3
VFSM(config-ip-if)# ip ospf key test
VFSM(config-ip-if)# exit
```

3. Enable OSPF authentication for Area 2 on switch 4.

```
VFSM(config)# router ospf
VFSM(config-router-ospf)# area 2 authentication-type password
```

4. Configure a simple text password up to eight characters for the virtual link between Area 2 and Area 0 on switches 2 and 4.

```
VFSM(config-router-ospf)# area-virtual-link 1 key IBM
```

## Configuring MD5 Authentication

Use the following commands to configure MD5 authentication on the switches shown in [Figure 45](#):

1. Enable OSPF MD5 authentication for Area 0 on switches 1, 2, and 3.

```
VFSM(config-router-ospf)# area 0 authentication-type md5
```

2. Configure MD5 key ID for Area 0 on switches 1, 2, and 3.

```
VFSM(config-router-ospf)# message-digest-key 1 md5-key test  
VFSM(config-router-ospf)# exit
```

3. Assign MD5 key ID to OSPF interfaces on switches 1, 2, and 3.

```
VFSM(config)# interface ip 1  
VFSM(config-ip-if)# ip ospf message-digest-key 1  
VFSM(config-ip-if)# exit  
VFSM(config)# interface ip 2  
VFSM(config-ip-if)# ip ospf message-digest-key 1  
VFSM(config-ip-if)# exit  
VFSM(config)# interface ip 3  
VFSM(config-ip-if)# ip ospf message-digest-key 1  
VFSM(config-ip-if)# exit
```

4. Enable OSPF MD5 authentication for Area 2 on switch 4.

```
VFSM(config)# router ospf  
VFSM(config-router-ospf)# area 1 authentication-type md5
```

5. Configure MD5 key for the virtual link between Area 2 and Area 0 on switch 2 and switch 4.

```
VFSM(config-router-ospf)# message-digest-key 2 md5-key test
```

6. Assign MD5 key ID to OSPF virtual link on switches 2 and 4.

```
VFSM(config-router-ospf)# area-virtual-link 1 message-digest-key 2  
VFSM(config-router-ospf)# exit
```

## Host Routes for Load Balancing

N/OS implementation of OSPF includes host routes. Host routes are used for advertising network device IP addresses to external networks, accomplishing the following goals:

- ABR Load Sharing

As a form of load balancing, host routes can be used for dividing OSPF traffic among multiple ABRs. To accomplish this, each switch provides identical services but advertises a host route for a different IP address to the external network. If each IP address serves a different and equal portion of the external world, incoming traffic from the upstream router should be split evenly among ABRs.

- ABR Failover

Complementing ABR load sharing, identical host routes can be configured on each ABR. These host routes can be given different costs so that a different ABR is selected as the preferred route for each server and the others are available as backups for failover purposes.

- Equal Cost Multipath (ECMP)

With equal cost multipath, a router potentially has several available next hops towards any given destination. ECMP allows separate routes to be calculated for each IP Type of Service. All paths of equal cost to a given destination are calculated, and the next hops for all equal-cost paths are inserted into the routing table.

If redundant routes via multiple routing processes (such as OSPF, RIP, BGP, or static routes) exist on your network, the switch defaults to the OSPF-derived route.

## Loopback Interfaces in OSPF

A loopback interface is an IP interface which has an IP address, but is not associated with any particular physical port. The loopback interface is thus always available to the general network, regardless of which specific ports are in operation. Because loopback interfaces are always available on the switch, loopback interfaces may present an advantage when used as the router ID.

If dynamic router ID selection is used (see [“Router ID” on page 423](#)), loopback interfaces can be used to force router ID selection. If a loopback interface is configured, its IP address is automatically selected as the router ID, even if other IP interfaces have lower IP addresses. If more than one loopback interface is configured, the lowest loopback interface IP address is selected.

Loopback interfaces can be advertised into the OSPF domain by specifying an OSPF host route with the loopback interface IP address.

**Note:** Loopback interfaces are not advertised via the OSPF route redistribution of fixed routes.

To enable OSPF on an existing loopback interface:

```
VFSM(config)# interface loopback <I-5>
VFSM(config-ip-loopback)# ip ospf area <area ID> enable
VFSM(config-ip-loopback)# exit
```

## OSPF Features Not Supported in This Release

The following OSPF features are not supported in this release:

- Summarizing external routes
- Filtering OSPF routes
- Using OSPF to forward multicast routes
- Configuring OSPF on non-broadcast multi-access networks (such as frame relay, X.25, or ATM)

---

## OSPFv2 Configuration Examples

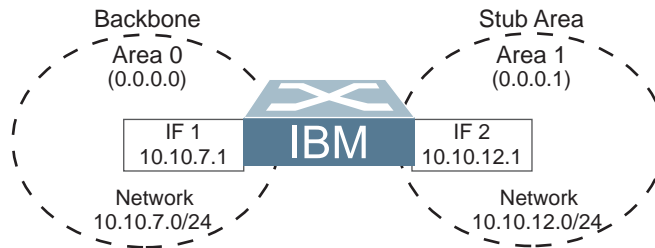
A summary of the basic steps for configuring OSPF on the VFSM is listed here. Detailed instructions for each of the steps is covered in the following sections:

1. Configure IP interfaces.  
One IP interface is required for each desired network (range of IP addresses) being assigned to an OSPF area on the switch.
2. (Optional) Configure the router ID.  
The router ID is required only when configuring virtual links on the switch.
3. Enable OSPF on the switch.
4. Define the OSPF areas.
5. Configure OSPF interface parameters.  
IP interfaces are used for attaching networks to the various areas.
6. (Optional) Configure route summarization between OSPF areas.
7. (Optional) Configure virtual links.
8. (Optional) Configure host routes.

## Example 1: Simple OSPF Domain

In this example, two OSPF areas are defined—one area is the backbone and the other is a stub area. A stub area does not allow advertisements of external routes, thus reducing the size of the database. Instead, a default summary route of IP address 0.0.0.0 is automatically inserted into the stub area. Any traffic for IP address destinations outside the stub area will be forwarded to the stub area's IP interface, and then into the backbone.

Figure 46. A Simple OSPF Domain



Follow this procedure to configure OSPF support as shown in [Figure 46](#):

1. Configure IP interfaces on each network that will be attached to OSPF areas.  
In this example, two IP interfaces are needed:
  - Interface 1 for the backbone network on 10.10.7.0/24
  - Interface 2 for the stub area network on 10.10.12.0/24

```
VFSM(config)# interface ip 1
VFSM(config-ip-if)# ip address 10.10.7.1 255.255.255.0 enable
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip address 10.10.12.1 255.255.255.0 enable
VFSM(config-ip-if)# exit
```

**Note:** OSPFv2 supports IPv4 only. IPv6 is supported in OSPFv3 (see [“OSPFv3 Implementation in IBM N/OS” on page 435](#)).

2. Enable OSPF.

```
VFSM(config)# router ospf
VFSM(config-router-ospf)# enable
```

3. Define the backbone.

The backbone is always configured as a transit area using areaid 0.0.0.0.

```
VFSM(config-router-ospf)# area 0 area-id 0.0.0.0
VFSM(config-router-ospf)# area 0 type transit
VFSM(config-router-ospf)# area 0 enable
```

4. Define the stub area.

```
VFSM(config-router-ospf)# area 1 area-id 0.0.0.1
VFSM(config-router-ospf)# area 1 type stub
VFSM(config-router-ospf)# area 1 enable
VFSM(config-router-ospf)# exit
```

5. Attach the network interface to the backbone.

```
VFSM(config)# interface ip 1
VFSM(config-ip-if)# ip ospf area 0
VFSM(config-ip-if)# ip ospf enable
VFSM(config-ip-if)# exit
```

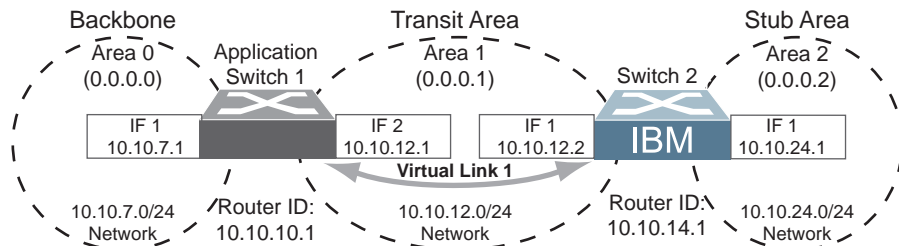
6. Attach the network interface to the stub area.

```
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip ospf area 1
VFSM(config-ip-if)# ip ospf enable
VFSM(config-ip-if)# exit
```

## Example 2: Virtual Links

In the example shown in [Figure 47](#), area 2 is not physically connected to the backbone as is usually required. Instead, area 2 will be connected to the backbone via a virtual link through area 1. The virtual link must be configured at each endpoint.

Figure 47. Configuring a Virtual Link



**Note:** OSPFv2 supports IPv4 only. IPv6 is supported in OSPFv3 (see [“OSPFv3 Implementation in IBM N/OS” on page 435](#)).

### Configuring OSPF for a Virtual Link on Switch #1

1. Configure IP interfaces on each network that will be attached to the switch.

In this example, two IP interfaces are needed:

- Interface 1 for the backbone network on 10.10.7.0/24
- Interface 2 for the transit area network on 10.10.12.0/24

```
VFSM(config)# interface ip 1
VFSM(config-ip-if)# ip address 10.10.7.1 255.255.255.0 enable
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip address 10.10.12.1 255.255.255.0 enable
VFSM(config-ip-if)# exit
```

2. Configure the router ID.

A router ID is required when configuring virtual links. Later, when configuring the other end of the virtual link on Switch 2, the router ID specified here will be used as the target virtual neighbor (nbr) address.

```
VFSM(config)# ip router-id 10.10.10.1
```

3. Enable OSPF.

```
VFSM(config)# router ospf
VFSM(config-router-ospf)# enable
```

4. Define the backbone.

```
VFSM(config-router-ospf)# area 0 area-id 0.0.0.0
VFSM(config-router-ospf)# area 0 type transit
VFSM(config-router-ospf)# area 0 enable
```

5. Define the transit area.

The area that contains the virtual link must be configured as a transit area.

```
VFSM(config-router-ospf)# area 1 area-id 0.0.0.1
VFSM(config-router-ospf)# area 1 type transit
VFSM(config-router-ospf)# area 1 enable
VFSM(config-router-ospf)# exit
```

6. Attach the network interface to the backbone.

```
VFSM(config)# interface ip 1
VFSM(config-ip-if)# ip ospf area 0
VFSM(config-ip-if)# ip ospf enable
VFSM(config-ip-if)# exit
```

7. Attach the network interface to the transit area.

```
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip ospf area 1
VFSM(config-ip-if)# ip ospf enable
VFSM(config-ip-if)# exit
```

8. Configure the virtual link.

The `nbr` router ID configured in this step must be the same as the router ID that will be configured for Switch #2 in [Step 2 on page 431](#).

```
VFSM(config)# router ospf
VFSM(config-router-ospf)# area-virtual-link 1 area 1
VFSM(config-router-ospf)# area-virtual-link 1 neighbor-router 10.10.14.1
VFSM(config-router-ospf)# area-virtual-link 1 enable
```

## Configuring OSPF for a Virtual Link on Switch #2

1. Configure IP interfaces on each network that will be attached to OSPF areas.

In this example, two IP interfaces are needed:

- Interface 1 for the transit area network on 10.10.12.0/24
- Interface 2 for the stub area network on 10.10.24.0/24

```
VFSM(config)# interface ip 1
VFSM(config-ip-if)# ip address 10.10.12.2 255.255.255.0 enable
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip address 10.10.24.1 255.255.255.0 enable
VFSM(config-ip-if)# exit
```

2. Configure the router ID.

A router ID is required when configuring virtual links. This router ID should be the same one specified as the target virtual neighbor on switch 1 in [Step 8 on page 431](#).

```
VFSM(config)# ip router-id 10.10.14.1
```

3. Enable OSPF.

```
VFSM(config)# router ospf
VFSM(config-router-ospf)# enable
```

4. Define the backbone.

This version of IBM N/OS requires that a backbone index be configured on the non-backbone end of the virtual link as follows:

```
VFSM(config-router-ospf)# area 0 area-id 0.0.0.0
VFSM(config-router-ospf)# area 0 enable
```

5. Define the transit area.

```
VFSM(config-router-ospf)# area 1 area-id 0.0.0.1
VFSM(config-router-ospf)# area 1 type transit
VFSM(config-router-ospf)# area 1 enable
```

6. Define the stub area.

```
VFSM(config-router-ospf)# area 2 area-id 0.0.0.2
VFSM(config-router-ospf)# area 2 type stub
VFSM(config-router-ospf)# area 2 enable
VFSM(config-router-ospf)# exit
```

7. Attach the network interface to the transmit area:

```
VFSM(config)# interface ip 1
VFSM(config-ip-if)# ip ospf area 1
VFSM(config-ip-if)# ip ospf enable
VFSM(config-ip-if)# exit
```

8. Attach the network interface to the stub area.

```
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip ospf area 2
VFSM(config-ip-if)# ip ospf enable
VFSM(config-ip-if)# exit
```

9. Configure the virtual link.

The `nbr` router ID configured in this step must be the same as the router ID that was configured for switch #1 in [Step 2 on page 430](#).

```
VFSM(config)# router ospf
VFSM(config-router-ospf)# area-virtual-link 1 area 1
VFSM(config-router-ospf)# area-virtual-link 1 neighbor-router 10.10.10.1
VFSM(config-router-ospf)# area-virtual-link 1 enable
```

## Other Virtual Link Options

- You can use redundant paths by configuring multiple virtual links.
- Only the endpoints of the virtual link are configured. The virtual link path may traverse multiple routers in an area as long as there is a routable path between the endpoints.

## Example 3: Summarizing Routes

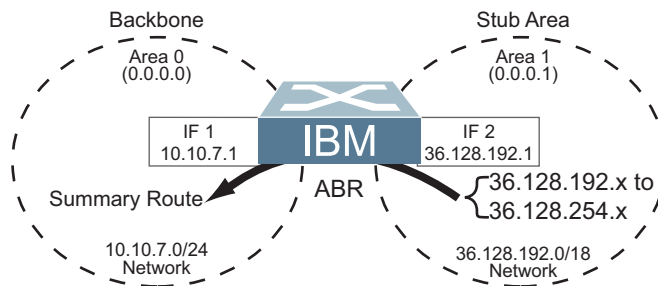
By default, ABRs advertise all the network addresses from one area into another area. Route summarization can be used for consolidating advertised addresses and reducing the perceived complexity of the network.

If the network IP addresses in an area are assigned to a contiguous subnet range, you can configure the ABR to advertise a single summary route that includes all the individual IP addresses within the area.

The following example shows one summary route from area 1 (stub area) injected into area 0 (the backbone). The summary route consists of all IP addresses from 36.128.192.0 through 36.128.254.255 except for the routes in the range 36.128.200.0 through 36.128.200.255.

**Note:** OSPFv2 supports IPv4 only. IPv6 is supported in OSPFv3 (see [“OSPFv3 Implementation in IBM N/OS” on page 435](#)).

Figure 48. Summarizing Routes



**Note:** You can specify a range of addresses to prevent advertising by using the hide option. In this example, routes in the range 36.128.200.0 through 36.128.200.255 are kept private.

Follow this procedure to configure OSPF support as shown in [Figure 48](#):

1. Configure IP interfaces for each network which will be attached to OSPF areas.

```
VFSM(config)# interface ip 1
VFSM(config-ip-if)# ip address 10.10.7.1 255.255.255.0 enable
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip address 36.128.192.1 255.255.255.0 enable
VFSM(config-ip-if)# exit
```

2. Enable OSPF.

```
VFSM(config)# router ospf
VFSM(config-router-ospf)# enable
```

3. Define the backbone.

```
VFSM(config-router-ospf)# area 0 area-id 0.0.0.0
VFSM(config-router-ospf)# area 0 type transit
VFSM(config-router-ospf)# area 0 enable
```

4. Define the stub area.

```
VFSM(config-router-ospf)# area 1 area-id 0.0.0.1
VFSM(config-router-ospf)# area 1 type stub
VFSM(config-router-ospf)# area 1 enable
VFSM(config-router-ospf)# exit
```

5. Attach the network interface to the backbone.

```
VFSM(config)# interface ip 1
VFSM(config-ip-if)# ip ospf area 0
VFSM(config-ip-if)# ip ospf enable
VFSM(config-ip-if)# exit
```

6. Attach the network interface to the stub area.

```
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip ospf area 1
VFSM(config-ip-if)# ip ospf enable
VFSM(config-ip-if)# exit
```

7. Configure route summarization by specifying the starting address and mask of the range of addresses to be summarized.

```
VFSM(config)# router ospf
VFSM(config-router-ospf)# area-range 1 address 36.128.192.0 255.255.192.0
VFSM(config-router-ospf)# area-range 1 area 1
VFSM(config-router-ospf)# area-range 1 enable
VFSM(config-router-ospf)# exit
```

8. Use the hide command to prevent a range of addresses from advertising to the backbone.

```
VFSM(config)# router ospf
VFSM(config-router-ospf)# area-range 2 address 36.128.200.0 255.255.255.0
VFSM(config-router-ospf)# area-range 2 area 1
VFSM(config-router-ospf)# area-range 2 hide
VFSM(config-router-ospf)# exit
```

## Verifying the OSPF Configuration

Use the following commands to verify the OSPF configuration on your switch:

- show ip ospf
- show ip ospf neighbor
- show ip ospf database database-summary
- show ip ospf routes

Refer to the *IBM Networking OS Command Reference* for information on the above commands.

---

## OSPFv3 Implementation in IBM N/OS

OSPF version 3 is based on OSPF version 2, but has been modified to support IPv6 addressing. In most other ways, OSPFv3 is similar to OSPFv2: They both have the same packet types and interfaces, and both use the same mechanisms for neighbor discovery, adjacency formation, LSA flooding, aging, and so on. The administrator should be familiar with the OSPFv2 concepts covered in the preceding sections of this chapter before implementing the OSPFv3 differences as described in the following sections.

Although OSPFv2 and OSPFv3 are very similar, they represent independent features on the VFSM. They are configured separately, and both can run in parallel on the switch with no relation to one another, serving different IPv6 and IPv4 traffic, respectively.

The IBM N/OS implementation conforms to the OSPFv3 interface-based authentication/confidentiality specifications in RFC 4552.

### OSPFv3 Differences from OSPFv2

**Note:** When OSPFv3 is enabled, the OSPF backbone area (0.0.0.0) is created by default and is always active.

### OSPFv3 Requires IPv6 Interfaces

OSPFv3 is designed to support IPv6 addresses. This requires IPv6 interfaces to be configured on the switch and assigned to OSPF areas, in much the same way IPv4 interfaces are assigned to areas in OSPFv2. This is the primary configuration difference between OSPFv3 and OSPFv2.

See [“Internet Protocol Version 6” on page 329](#) for configuring IPv6 interfaces.

### OSPFv3 Uses Independent Command Paths

Though OSPFv3 and OSPFv2 are very similar, they are configured independently. OSPFv3 command paths are located as follows:

VFSM(config)# ipv6 router ospf	(OSPFv3 router config mode)
VFSM(config-router-ospf3)# ?	
VFSM(config)# interface ip <Interface number>	(Configure OSPFv3)
VFSM(config-ip-if)# ipv6 ospf ?	(OSPFv3 interface config)
VFSM# show ipv6 ospf ?	(Show OSPFv3 information)

### OSPFv3 Identifies Neighbors by Router ID

Where OSPFv2 uses a mix of IPv4 interface addresses and Router IDs to identify neighbors, depending on their type, OSPFv3 configuration consistently uses a Router ID to identify all neighbors.

Although Router IDs are written in dotted decimal notation, and may even be based on IPv4 addresses from an original OSPFv2 network configuration, it is important to realize that Router IDs are not IP addresses in OSPFv3, and can be assigned

independently of IP address space. However, maintaining Router IDs consistent with any legacy OSPFv2 IPv4 addressing allows for easier implementation of both protocols.

## Other Internal Improvements

OSPFv3 has numerous improvements that increase the protocol efficiency in addition to supporting IPv6 addressing. These improvements change some of the behaviors in the OSPFv3 network and may affect topology consideration, but have little direct impact on configuration. For example:

- Addressing fields have been removed from Router and Network LSAs.
- Flexible treatment of unknown LSA types to make integration of OSPFv3 easier.
- Interface network type can be specified using the command:  
`VFSM(config-ip-if)# ipv6 ospf network`  
`{broadcast|non-broadcast|point-to-multipoint|point-to-point}`
- For an interface network type that is not broadcast or NBMA, link LSA suppression can be enabled so link LSA is not originated for the interface. Use the command: `VFSM(config-ip-if)# ipv6 ospf linklsasuppress`

## OSPFv3 Limitations

N/OS 7.8 does not currently support the following OSPFv3 features:

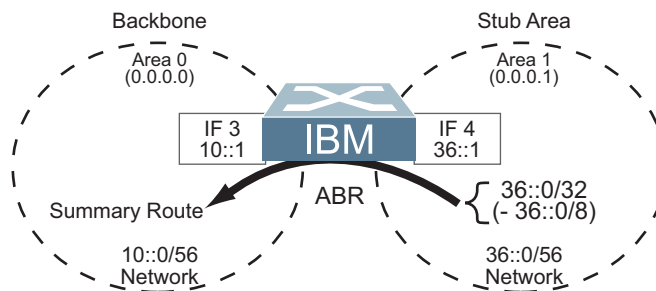
- Multiple instances of OSPFv3 on one IPv6 link.
- Authentication of OSPFv3 packets via IPv6 Security (IPsec) for virtual links.

## OSPFv3 Configuration Example

The following example depicts the OSPFv3 equivalent configuration of [“Example 3: Summarizing Routes” on page 433](#) for OSPFv2.

In this example, one summary route from area 1 (stub area) is injected into area 0 (the backbone). The summary route consists of all IP addresses for the 36::0/32 portion of the 36::0/56 network except for the routes in the 36::0/8 range.

Figure 49. Summarizing Routes



**Note:** You can specify a range of addresses to prevent advertising by using the `hide` option. In this example, routes in the 36::0/8 range are kept private.

Use the following procedure to configure OSPFv3 support as shown in [Figure 49](#):

1. Configure IPv6 interfaces for each link which will be attached to OSPFv3 areas.

```
VFSM(config)# interface ip 3
VFSM(config-ip-if)# ipv6 address 10:0:0:0:0:0:1
VFSM(config-ip-if)# ipv6 prefixlen 56
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 4
VFSM(config-ip-if)# ip address 36:0:0:0:0:0:1
VFSM(config-ip-if)# ipv6 prefixlen 56
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# exit
```

This is equivalent to configuring the IP address and netmask for IPv4 interfaces.

2. Enable OSPFv3.

```
VFSM(config)# ipv6 router ospf
VFSM(config-router-ospf3)# enable
```

This is equivalent to the OSPFv2 enable option in the `router ospf` command path.

3. Define the backbone.

```
VFSM(config-router-ospf3)# area 0 area-id 0.0.0.0
VFSM(config-router-ospf3)# area 0 type transit
VFSM(config-router-ospf3)# area 0 enable
```

This is identical to OSPFv2 configuration.

4. Define the stub area.

```
VFSM(config-router-ospf3)# area 1 area-id 0.0.0.1
VFSM(config-router-ospf3)# area 1 type stub
VFSM(config-router-ospf3)# area 1 enable
VFSM(config-router-ospf3)# exit
```

This is identical to OSPFv2 configuration.

5. Attach the network interface to the backbone.

```
VFSM(config)# interface ip 3
VFSM(config-ip-if)# ipv6 ospf area 0
VFSM(config-ip-if)# ipv6 ospf enable
VFSM(config-ip-if)# exit
```

The `ipv6` command path is used instead of the OSPFv2 `ip` command path

6. Attach the network interface to the stub area.

```
VFSM(config)# interface ip 4
VFSM(config-ip-if)# ipv6 ospf area 1
VFSM(config-ip-if)# ipv6 ospf enable
VFSM(config-ip-if)# exit
```

The `ipv6` command path is used instead of the OSPFv2 `ip` command path

7. Configure route summarization by specifying the starting address and prefix length of the range of addresses to be summarized.

```
VFSM(config)# ipv6 router ospf
VFSM(config-router-ospf3)# area-range 1 address 36:0:0:0:0:0:0:0 32
VFSM(config-router-ospf3)# area-range 1 area 0
VFSM(config-router-ospf3)# area-range 1 enable
```

This differs from OSPFv2 only in that the OSPFv3 command path is used, and the address and prefix are specified in IPv6 format.

8. Use the hide command to prevent a range of addresses from advertising to the backbone.

```
VFSM(config-router-ospf)# area-range 2 address 36:0:0:0:0:0:0:0 8
VFSM(config-router-ospf)# area-range 2 area 0
VFSM(config-router-ospf)# area-range 2 hide
VFSM(config-router-ospf)# exit
```

This differs from OSPFv2 only in that the OSPFv3 command path is used, and the address and prefix are specified in IPv6 format.

## Neighbor Configuration Example

When using NBMA or point to multipoint interfaces, you must manually configure neighbors. Following example includes the steps for neighbor configuration.

1. Configure IPv6 interface parameters:

```
VFSM(config# interface ip 10
VFSM(config-ip-if)# ipv6 address 10:0:0:0:0:0:12 64
VFSM(config-ip-if)# vlan 10
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# ipv6 ospf area 0
VFSM(config-ip-if)# ipv6 ospf retransmit-interval 5
VFSM(config-ip-if)# ipv6 ospf transmit-delay 1
VFSM(config-ip-if)# ipv6 ospf priority 1
VFSM(config-ip-if)# ipv6 ospf hello-interval 10
VFSM(config-ip-if)# ipv6 ospf dead-interval 40
VFSM(config-ip-if)# ipv6 ospf network point-to-multipoint
VFSM(config-ip-if)# ipv6 ospf poll-interval 120
VFSM(config-ip-if)# ipv6 ospf enable
VFSM(config-ip-if)# exit
```

2. Enable OSPFv3:

```
VFSM(config# ipv6 router ospf
VFSM(config-router-ospf3)# router-id 12.12.12.12
VFSM(config-router-ospf3)# enable
```

3. Define the backbone.

```
VFSM(config-router-ospf3)# area 0 area-id 0.0.0.0
VFSM(config-router-ospf3)# area 0 stability-interval 40
VFSM(config-router-ospf3)# area 0 default-metric 1
VFSM(config-router-ospf3)# area 0 default-metric type 1
VFSM(config-router-ospf3)# area 0 translation-role candidate
VFSM(config-router-ospf3)# area 0 type transit
VFSM(config-router-ospf3)# area 0 enable
```

#### 4. Configure neighbor entry:

```
VFSM(config-router-ospf3)# neighbor 1 address fe80:0:0:0:dceb:ff:fe00:9  
VFSM(config-router-ospf3)# neighbor 1 interface 10  
VFSM(config-router-ospf3)# neighbor 1 priority 1  
VFSM(config-router-ospf3)# neighbor 1 enable
```



# Part 6: High Availability Fundamentals

Internet traffic consists of myriad services and applications which use the Internet Protocol (IP) for data delivery. However, IP is not optimized for all the various applications. High Availability goes beyond IP and makes intelligent switching decisions to provide redundant network configurations.



---

## Chapter 27. Basic Redundancy

IBM Networking OS 7.8 includes various features for providing basic link or device redundancy:

- [“Trunking for Link Redundancy” on page 444](#)
- [“Hot Links” on page 445](#)
- [“Stacking for High Availability Topologies” on page 447](#)

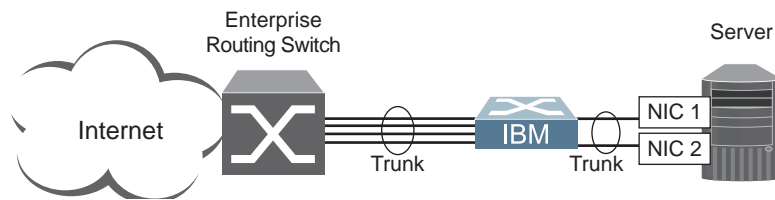
---

## Trunking for Link Redundancy

Multiple switch ports can be combined together to form robust, high-bandwidth trunks to other devices. Since trunks are comprised of multiple physical links, the trunk group is inherently fault tolerant. As long as one connection between the switches is available, the trunk remains active.

In [Figure 50](#), four ports are trunked together between the switch and the enterprise routing device. Connectivity is maintained as long as one of the links remain active. The links to the server are also trunked, allowing the secondary NIC to take over in the event that the primary NIC link fails.

Figure 50. Trunking Ports for Link Redundancy



For more information on trunking, see [“Ports and Trunking” on page 143](#).

---

## Hot Links

For network topologies that require Spanning Tree to be turned off, Hot Links provides basic link redundancy with fast recovery.

Hot Links consists of up to 200 triggers. A trigger consists of a pair of layer 2 interfaces, each containing an individual port, trunk, or LACP adminkey. One interface is the Master, and the other is a Backup. While the Master interface is set to the active state and forwards traffic, the Backup interface is set to the standby state and blocks traffic until the Master interface fails. If the Master interface fails, the Backup interface is set to active and forwards traffic. Once the Master interface is restored, it transitions to the standby state and blocks traffic until the Backup interface fails.

You may select a physical port, static trunk, or an LACP adminkey as a Hot Link interface. Only external uplink ports (EXT.x) can be members of a Hot Links trigger interface.

## Forward Delay

The Forward Delay timer allows Hot Links to monitor the Master and Backup interfaces for link stability before selecting one interface to transition to the active state. Before the transition occurs, the interface must maintain a stable link for the duration of the Forward Delay interval.

For example, if you set the Forward delay timer to 10 seconds, the switch will select an interface to become active only if a link remained stable for the duration of the Forward Delay period. If the link is unstable, the Forward Delay period starts again.

## Preemption

You can configure the Master interface to resume the active state whenever it becomes available. With Hot Links preemption enabled, the Master interface transitions to the active state immediately upon recovery. The Backup interface immediately transitions to the standby state. If Forward Delay is enabled, the transition occurs when an interface has maintained link stability for the duration of the Forward Delay period.

## FDB Update

Use the FDB update option to notify other devices on the network about updates to the Forwarding Database (FDB). When you enable FDB update, the switch sends multicasts of addresses in the forwarding database (FDB) over the active interface, so that other devices on the network can learn the new path. The Hot Links FBD update option uses the station update rate to determine the rate at which to send FDB packets.

## Configuration Guidelines

The following configuration guidelines apply to Hot links:

- Only external ports and inter-switch links can be configured as Hot Links.
- When Hot Links is turned on, MSTP, RSTP, and PVRST must be turned off.
- A port that is a member of the Master interface cannot be a member of the Backup interface. A port that is a member of one Hot Links trigger cannot be a member of another Hot Links trigger.

- An individual port that is configured as a Hot Link interface cannot be a member of a trunk.

## Configuring Hot Links

Use the following commands to configure Hot Links.

VFSM(config)# hotlinks trigger 1 enable	<i>(Enable Hot Links Trigger 1)</i>
VFSM(config)# hotlinks trigger 1 master port 38	<i>(Add port to Master interface)</i>
VFSM(config)# hotlinks trigger 1 backup port 39	<i>(Add port to Backup interface)</i>
VFSM(config)# hotlinks enable	<i>(Turn on Hot Links)</i>

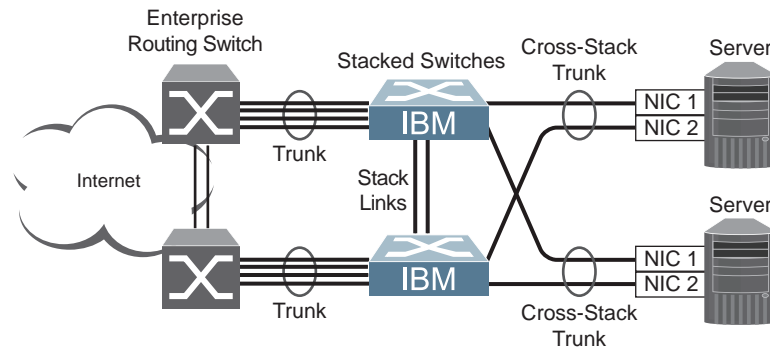
---

## Stacking for High Availability Topologies

A *stack* is a group of up to eight Virtual Fabric Switch Module devices that work together as a unified system. Because the multiple members of a stack acts as a single switch entity with distributed resources, high-availability topologies can be more easily achieved.

In [Figure 51](#), a simple stack using two switches provides full redundancy in the event that either switch were to fail. As shown with the servers in the example, stacking permits ports within different physical switches to be trunked together, further enhancing switch redundancy.

Figure 51. High Availability Topology Using Stacking



For more information on stacking, see [“Stacking” on page 195](#).



---

## Chapter 28. Layer 2 Failover

The primary application for Layer 2 Failover is to support Network Adapter Teaming. With Network Adapter Teaming, all the NICs on each server share the same IP address, and are configured into a team. One NIC is the primary link, and the other is a standby link. For more details, refer to the documentation for your Ethernet adapter.

**Note:** Only two links per server blade can be used for Layer 2 Trunk Failover (one primary and one backup). Network Adapter Teaming allows only one backup NIC for each server blade.

## Auto Monitoring Trunk Links

Layer 2 Failover can be enabled on any trunk group in the VFSM, including LACP trunks. Trunks can be added to failover trigger groups. Then, if some specified number of trigger links fail, the switch disables all the internal ports in the switch (unless VLAN Monitor is turned on). When the internal ports are disabled, it causes the NIC team on the affected server blades to failover from the primary to the backup NIC. This process is called a failover event.

When the appropriate number of links in a trigger group return to service, the switch enables the internal ports. This causes the NIC team on the affected server blades to fail back to the primary switch (unless Auto-Fallback is disabled on the NIC team). The backup switch processes traffic until the primary switch's internal links come up, which can take up to five seconds.

## VLAN Monitor

The VLAN Monitor allows Layer 2 Failover to discern different VLANs. With VLAN Monitor turned on:

- If enough links in a trigger fail (see [“Setting the Failover Limit” on page 452](#)), the switch disables all internal ports that reside in the same VLAN membership as the trunk(s) in the trigger.
- When enough links in the trigger return to service, the switch enables the internal ports that reside in the same VLAN membership as the trunk(s) in the trigger.

If you turn off the VLAN Monitor, only one failover trigger is allowed. When a link failure occurs on the trigger, the switch disables all internal server-blade ports.

## Auto Monitor Configurations

[Figure 52](#) is a simple example of Layer 2 Failover. One VFSM is the primary, and the other is used as a backup. In this example, all external ports on the primary switch belong to a single trunk group, with Layer 2 Failover enabled, and Failover Limit set to 2. If two or fewer links in trigger 1 remain active, the switch temporarily disables all internal server-blade ports that reside in VLAN 1. This action causes a failover event on Server 1 and Server 2.

Figure 52. Basic Layer 2 Failover

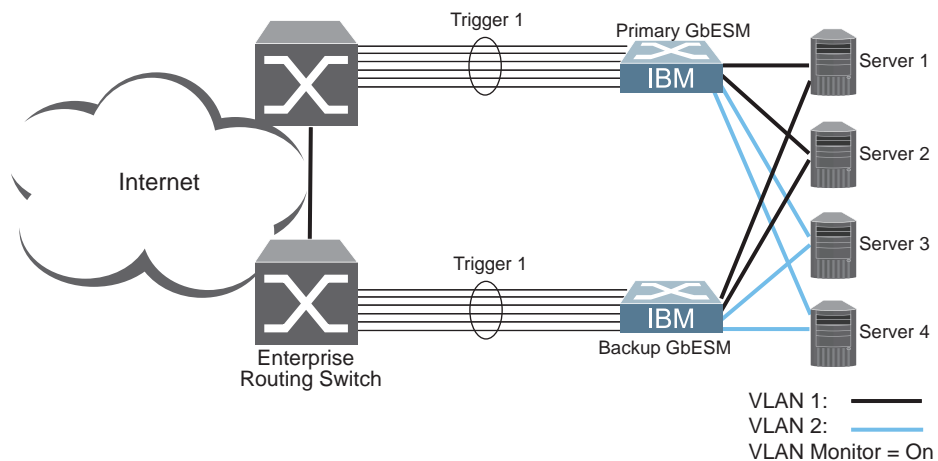


Figure 53 shows a configuration with two trunks, each in a different Failover Trigger. VFSM 1 is the primary switch for Server 1 and Server 2. VFSM 2 is the primary switch for Server 3 and Server 4. VLAN Monitor is turned on. STP is turned off.

If all links go down in trigger 1, VFSM 1 disables all internal ports that reside in VLAN 1. If all links in trigger 2 go down, VFSM 1 disables all internal ports that reside in VLAN 2.

Figure 53. Two trunks, each in a different Failover Trigger

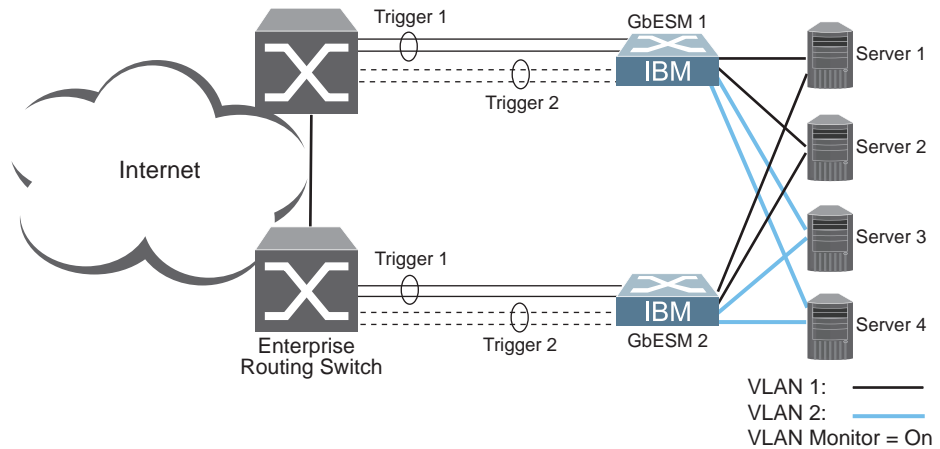
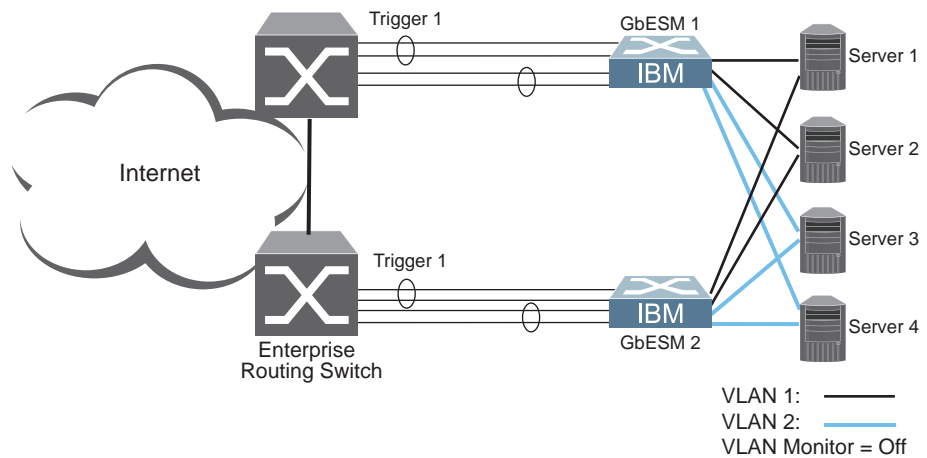


Figure 54 shows a configuration with two trunks. VLAN Monitor is turned off, so only one Failover Trigger is configured on each switch. VFSM 1 is the primary switch for Server 1 and Server 2. VFSM 2 is the primary switch for Server 3 and Server 4. STP is turned off.

If all links in trigger 1 go down, VFSM 1 disables all internal links to server blades.

Figure 54. Two trunks, one Failover Trigger



---

## Setting the Failover Limit

The failover limit lets you specify the minimum number of operational links required within each trigger before the trigger initiates a failover event. For example, if the limit is two, a failover event occurs when the number of operational links in the trigger is two or fewer. When you set the limit to zero, the switch triggers a failover event only when no links in the trigger are operational.

---

## Manually Monitoring Port Links

The Manual Monitor allows you to configure a set of ports and/or trunks to monitor for link failures (a monitor list), and another set of ports and/or trunks to disable when the trigger limit is reached (a control list). When the switch detects a link failure on the monitor list, it automatically disables the items in control list. When server ports are disabled, the corresponding server's network adapter can detect the disabled link, and trigger a network-adapter failover to another port or trunk on the switch, or another switch in the chassis.

The switch automatically enables the control list items when the monitor list items return to service.

### Monitor Port State

A monitor port is considered operational as long as the following conditions are true:

- The port must be in the `Link Up` state.
- If STP is enabled, the port must be in the `Forwarding` state.
- If the port is part of an LACP trunk, the port must be in the `Aggregated` state.

If any of the above conditions is false, the monitor port is considered to have failed.

### Control Port State

A control port is considered Operational if the monitor trigger is up. As long as the trigger is up, the port is considered operational from a teaming perspective, even if the port itself is actually in the `Down` state, `Blocking` state (if STP is enabled on the port), or `Not Aggregated` state (if part of an LACP trunk).

A control port is considered to have failed only if the monitor trigger is in the `Down` state.

To view the state of any port, use one of the following commands:

VFSM# show interface link	<i>(View port link status)</i>
VFSM# show interface port <x> spanning-tree stp <x>	<i>(View port STP status)</i>
VFSM# show lacp information	<i>(View port LACP status)</i>

---

## L2 Failover with Other Features

L2 Failover works together with Link Aggregation Control Protocol (LACP) and with Spanning Tree Protocol (STP), as described below.

### LACP

Link Aggregation Control Protocol allows the switch to form dynamic trunks. You can use the *admin key* to add up to two LACP trunks to a failover trigger using automatic monitoring. When you add an *admin key* to a trigger, any LACP trunk with that *admin key* becomes a member of the trigger.

### Spanning Tree Protocol

If Spanning Tree Protocol (STP) is enabled on the ports in a failover trigger, the switch monitors the port STP state rather than the link state. A port failure results when STP is not in a Forwarding state (such as Listening, Learning, Blocking, or No Link) in all the Spanning Tree Groups (STGs) to which the port belongs. The switch automatically disables the appropriate control ports.

When the switch determines that ports in the trigger are in STP Forwarding state in any one of the STGs it belongs to, then it automatically enables the appropriate control ports. The switch *fails back* to normal operation.

For example, if a monitor port is a member of STG1, STG2, and STG3, a failover will be triggered only if the port is not in a forwarding state in all the three STGs. When the port state in any of the three STGs changes to forwarding, then the control port is enabled and normal switch operation is resumed.

---

## Configuration Guidelines

This section provides important information about configuring Layer 2 Failover.

**Note:** Auto Monitor and Manual Monitor are mutually exclusive. They cannot both be configured on the switch.

### Auto Monitor Guidelines

- Any specific failover trigger may monitor static trunks only or LACP trunks only, but not both.
- All external ports in all static or LACP trunks added to any specific failover trigger must belong to the same VLAN.
- A maximum of two LACP keys can be added per trigger.
- When VLAN Monitor is on, the following additional guidelines apply:
  - All external ports in all static or LACP trunks added to a specific failover trigger must belong to the same VLAN and have the same PVID/Native VLAN.
  - Different triggers are not permitted to operate on the same VLAN.
  - Different triggers are not permitted to operate on the same internal port.
  - For each port in each trunk in a specific failover trigger, the trigger will monitor the STP state on only the default PVID/Native VLAN.

### Manual Monitor Guidelines

- A Manual Monitor can monitor only external ports.
- Any specific failover trigger can monitor external ports only, static trunks only, or LACP trunks only. The different types cannot be combined in the same trigger.
- A maximum of two LACP keys can be added per trigger.
- Management ports and stacking ports cannot be monitored.
- Port membership for different triggers should not overlap. Any specific port should be a member of only one trigger.

---

## Configuring Layer 2 Failover

### Auto Monitor Example

The following procedure pertains to the configuration shown in [Figure 52](#).

1. Configure Network Adapter Teaming on the servers.
2. Define a trunk group on the VFSM.

```
VFSM(config)# portchannel 1 port EXT1,EXT2,EXT3 enable
```

3. Configure Failover parameters.

```
VFSM(config)# failover trigger 1 enable  
VFSM(config)# failover trigger 1 limit <0-1024>  
VFSM(config)# failover trigger 1 amon portchannel 1
```

4. Verify the configuration.

```
VFSM(config)# show failover trigger 1 information
```

### Manual Monitor Example

Use the following procedure to configure a Layer 2 Failover Manual Monitor.

1. Configure Network Adapter Teaming on the servers.
2. Specify the links to monitor.

```
VFSM(config)# failover trigger 1 mmon monitor member EXT4,EXT5,EXT6
```

3. Specify the links to disable when the failover limit is reached.

```
VFSM(config)# failover trigger 1 mmon control member INT13,INT14
```

4. Configure general Layer 2 Failover parameters.

```
VFSM(config)# failover trigger 1 enable  
VFSM(config)# failover trigger 1 limit <0-1024>
```

5. Enable failover globally.

```
VFSM(config)# failover enable
```

6. Verify the configuration.

```
VFSM(config)# show failover trigger 1 information
```

---

## Chapter 29. Virtual Router Redundancy Protocol

The IBM Virtual Fabric Switch Module (VFSM) supports IPv4 high-availability network topologies through an enhanced implementation of the Virtual Router Redundancy Protocol (VRRP).

**Note:** IBM Networking OS 7.8 does not support IPv6 for VRRP.

The following topics are discussed in this chapter:

- [“VRRP Overview” on page 458](#). This section discusses VRRP operation and IBM N/OS redundancy configurations.
- [“Failover Methods” on page 461](#). This section describes the three modes of high availability.
- [“IBM N/OS Extensions to VRRP” on page 464](#). This section describes VRRP enhancements implemented in N/OS.
- [“Virtual Router Deployment Considerations” on page 465](#). This section describes issues to consider when deploying virtual routers.
- [“High Availability Configurations” on page 466](#). This section discusses the more useful and easily deployed redundant configurations.
  - [“Active-Active Configuration” on page 466](#)
  - [“Hot-Standby Configuration” on page 470](#)

---

## VRRP Overview

In a high-availability network topology, no device can create a single point-of-failure for the network or force a single point-of-failure to any other part of the network. This means that your network will remain in service despite the failure of any single device. To achieve this usually requires redundancy for all vital network components.

VRRP enables redundant router configurations within a LAN, providing alternate router paths for a host to eliminate single points-of-failure within a network. Each participating VRRP-capable routing device is configured with the same virtual router IPv4 address and ID number. One of the virtual routers is elected as the master, based on a number of priority criteria, and assumes control of the shared virtual router IPv4 address. If the master fails, one of the backup virtual routers will take control of the virtual router IPv4 address and actively process traffic addressed to it.

With VRRP, Virtual Interface Routers (VIR) allow two VRRP routers to share an IP interface across the routers. VIRs provide a single Destination IPv4 (DIP) address for upstream routers to reach various servers, and provide a virtual default Gateway for the server blades.

## VRRP Components

Each physical router running VRRP is known as a *VRRP router*.

### Virtual Router

Two or more VRRP routers can be configured to form a *virtual router* (RFC 2338). Each VRRP router may participate in one or more virtual routers. Each virtual router consists of a user-configured *virtual router identifier* (VRID) and an IPv4 address.

### Virtual Router MAC Address

The VRID is used to build the *virtual router MAC Address*. The five highest-order octets of the virtual router MAC Address are the standard MAC prefix (00-00-5E-00-01) defined in RFC 2338. The VRID is used to form the lowest-order octet.

### Owners and Renters

Only one of the VRRP routers in a virtual router may be configured as the IPv4 address owner. This router has the virtual router's IPv4 address as its real interface address. This router responds to packets addressed to the virtual router's IPv4 address for ICMP pings, TCP connections, and so on.

There is no requirement for any VRRP router to be the IPv4 address owner. Most VRRP installations choose not to implement an IPv4 address owner. For the purposes of this chapter, VRRP routers that are not the IPv4 address owner are called *renters*.

### Master and Backup Virtual Router

Within each virtual router, one VRRP router is selected to be the virtual router master. See [“Selecting the Master VRRP Router” on page 460](#) for an explanation of the selection process.

**Note:** If the IPv4 address owner is available, it will always become the virtual router master.

The virtual router master forwards packets sent to the virtual router. It also responds to Address Resolution Protocol (ARP) requests sent to the virtual router's IPv4 address. Finally, the virtual router master sends out periodic advertisements to let other VRRP routers know it is alive and its priority.

Within a virtual router, the VRRP routers not selected to be the master are known as virtual router backups. Should the virtual router master fail, one of the virtual router backups becomes the master and assumes its responsibilities.

### **Virtual Interface Router**

At Layer 3, a Virtual Interface Router (VIR) allows two VRRP routers to share an IP interface across the routers. VIRs provide a single Destination IPv4 (DIP) address for upstream routers to reach various destination networks, and provide a virtual default Gateway.

**Note:** Every VIR must be assigned to an IP interface, and every IP interface must be assigned to a VLAN. If no port in a VLAN has link up, the IP interface of that VLAN is down, and if the IP interface of a VIR is down, that VIR goes into INIT state.

## **VRRP Operation**

Only the virtual router master responds to ARP requests. Therefore, the upstream routers only forward packets destined to the master. The master also responds to ICMP ping requests. The backup does not forward any traffic, nor does it respond to ARP requests.

If the master is not available, the backup becomes the master and takes over responsibility for packet forwarding and responding to ARP requests.

## Selecting the Master VRRP Router

Each VRRP router is configured with a priority between 1–254. A bidding process determines which VRRP router is or becomes the master—the VRRP router with the highest priority.

The master periodically sends advertisements to an IPv4 multicast address. As long as the backups receive these advertisements, they remain in the backup state. If a backup does not receive an advertisement for three advertisement intervals, it initiates a bidding process to determine which VRRP router has the highest priority and takes over as master. In addition to the three advertisement intervals, a manually set holdoff time can further delay the backups from assuming the master status.

If, at any time, a backup determines that it has higher priority than the current master does, it can preempt the master and become the master itself, unless configured not to do so. In preemption, the backup assumes the role of master and begins to send its own advertisements. The current master sees that the backup has higher priority and will stop functioning as the master.

A backup router can stop receiving advertisements for one of two reasons—the master can be down, or all communications links between the master and the backup can be down. If the master has failed, it is clearly desirable for the backup (or one of the backups, if there is more than one) to become the master.

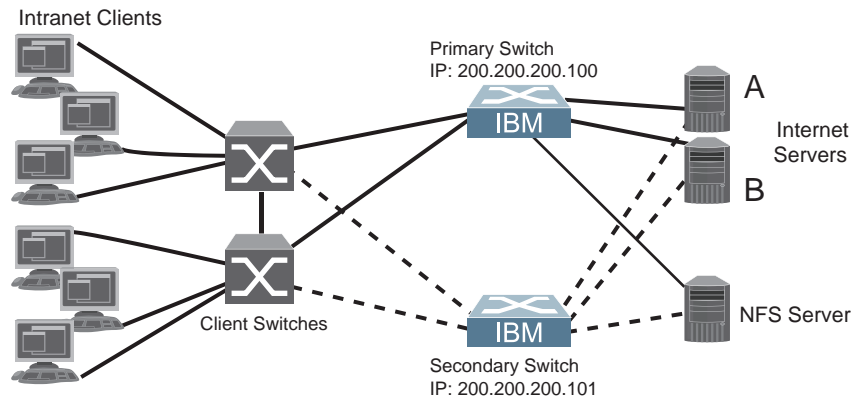
**Note:** If the master is healthy but communication between the master and the backup has failed, there will then be two masters within the virtual router. To prevent this from happening, configure redundant links to be used between the switches that form a virtual router.

---

## Failover Methods

With service availability becoming a major concern on the Internet, service providers are increasingly deploying Internet traffic control devices, such as application switches, in redundant configurations. Traditionally, these configurations have been *hot-standby* configurations, where one switch is active and the other is in a standby mode. A non-VRRP hot-standby configuration is shown [Figure 55](#):

**Figure 55** A Non-VRRP, Hot-Standby Configuration



While hot-standby configurations increase site availability by removing single points-of-failure, service providers increasingly view them as an inefficient use of network resources because one functional application switch sits by idly until a failure calls it into action. Service providers now demand that vendors' equipment support redundant configurations where all devices can process traffic when they are healthy, increasing site throughput and decreasing user response times when no device has failed.

N/OS high availability configurations are based on VRRP. The N/OS implementation of VRRP includes proprietary extensions.

The N/OS implementation of VRRP supports the following modes of high availability:

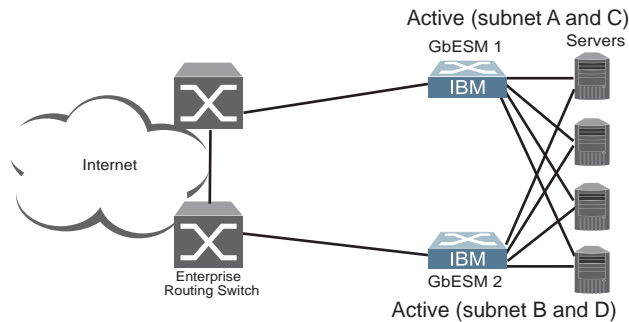
- **Active-Active**—based on proprietary N/OS extensions to VRRP
- **Hot-Standby**—supports Network Adapter Teaming on your server blades

## Active-Active Redundancy

In an active-active configuration, shown in [Figure 56](#), two switches provide redundancy for each other, with both active at the same time. Each switch processes traffic on a different subnet. When a failure occurs, the remaining switch can process traffic on all subnets.

For a configuration example, see [“High Availability Configurations” on page 466](#).

Figure 56. Active-Active Redundancy

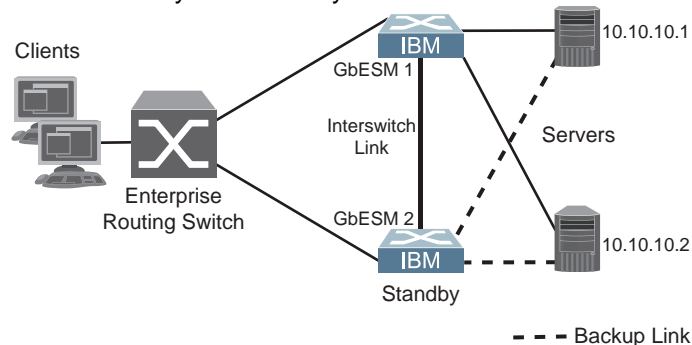


## Hot-Standby Redundancy

The primary application for VRRP-based hot-standby is to support Server Load Balancing when you have configured Network Adapter Teaming on your server blades. With Network Adapter Teaming, the NICs on each server share the same IPv4 address, and are configured into a team. One NIC is the primary link, and the others are backup links. For more details, refer to the NetXen 10 Gb Ethernet Adapter documentation.

The hot-standby model is shown in [Figure 57](#).

Figure 57. Hot-Standby Redundancy



## Virtual Router Group

The virtual router group ties all virtual routers on the switch together as a single entity. By definition, hot-standby requires that all virtual routers failover as a group, and not individually. As members of a group, all virtual routers on the switch (and therefore the switch itself), are in either a master or standby state.

The virtual router group cannot be used for active-active configurations or any other configuration that require shared interfaces.

A VRRP group has the following characteristics:

- When enabled, all virtual routers behave as one entity, and all group settings override any individual virtual router settings.
- All individual virtual routers, once the VRRP group is enabled, assume the group's tracking and priority.
- When one member of a VRRP group fails, the priority of the group decreases, and the state of the entire switch changes from Master to Standby.

Each VRRP advertisement can include up to 128 addresses. All virtual routers are advertised within the same packet, conserving processing and buffering resources.

---

## IBM N/OS Extensions to VRRP

This section describes VRRP enhancements that are implemented in N/OS.

N/OS supports a tracking function that dynamically modifies the priority of a VRRP router, based on its current state. The objective of tracking is to have, whenever possible, the master bidding processes for various virtual routers in a LAN converge on the same switch. Tracking ensures that the selected switch is the one that offers optimal network performance. For tracking to have any effect on virtual router operation, preemption must be enabled.

N/OS can track the attributes listed in [Table 35](#):

*Table 35. VRRP Tracking Parameters*

Parameter	Description
Number of IP interfaces on the switch that are active ("up") <code>tracking-priority-increment interfaces</code>	Helps elect the virtual routers with the most available routes as the master. (An IP interface is considered active when there is at least one active port on the same VLAN.) This parameter influences the VRRP router's priority in virtual interface routers.
Number of active ports on the same VLAN <code>tracking-priority-increment ports</code>	Helps elect the virtual routers with the most available ports as the master. This parameter influences the VRRP router's priority in virtual interface routers.  <b>Note:</b> In a hot-standby configuration, only external ports are tracked.
Number of virtual routers in master mode on the switch <code>tracking-priority-increment virtual-routers</code>	Useful for ensuring that traffic for any particular client/server pair is handled by the same switch, increasing routing efficiency. This parameter influences the VRRP router's priority in virtual interface routers.

Each tracked parameter has a user-configurable weight associated with it. As the count associated with each tracked item increases (or decreases), so does the VRRP router's priority, subject to the weighting associated with each tracked item. If the priority level of a standby is greater than that of the current master, then the standby can assume the role of the master.

See ["Configuring the Switch for Tracking" on page 465](#) for an example on how to configure the switch for tracking VRRP priority.

---

## Virtual Router Deployment Considerations

### Assigning VRRP Virtual Router ID

During the software upgrade process, VRRP virtual router IDs will be automatically assigned if failover is enabled on the switch. When configuring virtual routers at any point after upgrade, virtual router ID numbers must be assigned. The virtual router ID may be configured as any number between 1 and 255. Use the following commands to configure the virtual router ID:

```
VFSM(config)# router vrrp
VFSM(config-vrrp)# virtual-router 1 virtual-router-id <1-255>
```

### Configuring the Switch for Tracking

Tracking configuration largely depends on user preferences and network environment. Consider the configuration shown in [Figure 56 on page 462](#). Assume the following behavior on the network:

- Switch 1 is the master router upon initialization.
- If switch 1 is the master and it has one fewer active servers than switch 2, then switch 1 remains the master.

This behavior is preferred because running one server down is less disruptive than bringing a new master online and severing all active connections in the process.

- If switch 1 is the master and it has two or more active servers fewer than switch 2, then switch 2 becomes the master.
- If switch 2 is the master, it remains the master even if servers are restored on switch 1 such that it has one fewer or an equal number of servers.
- If switch 2 is the master and it has one active server fewer than switch 1, then switch 1 becomes the master.

The user can implement this behavior by configuring the switch for tracking as follows:

1. Set the priority for switch 1 to 101.
2. Leave the priority for switch 2 at the default value of 100.
3. On both switches, enable tracking based on ports (`ports`), interfaces (`ifs`), or virtual routers (`vr`). You can choose any combination of tracking parameters, based on your network configuration.

**Note:** There is no shortcut to setting tracking parameters. The goals must first be set and the outcomes of various configurations and scenarios analyzed to find settings that meet the goals.

## High Availability Configurations

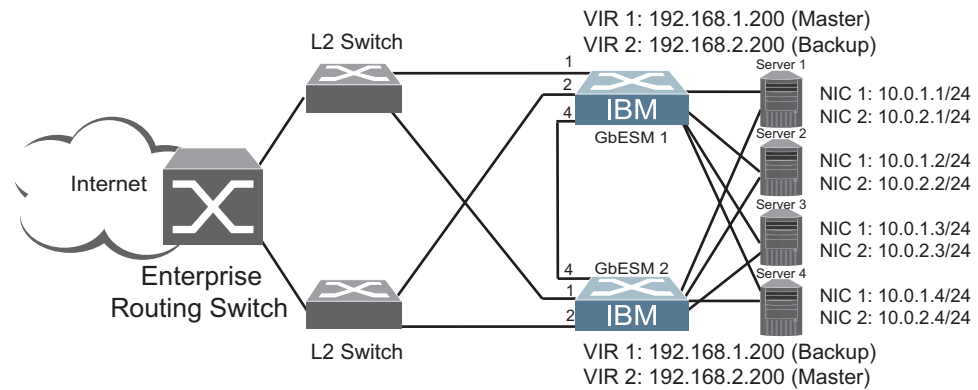
VFSMs offer flexibility in implementing redundant configurations. This section discusses the more useful and easily deployed configurations:

- [“Active-Active Configuration” on page 466](#)
- [“Hot-Standby Configuration” on page 470](#)

### Active-Active Configuration

[Figure 58](#) shows an example configuration where two VFSMs are used as VRRP routers in an active-active configuration. In this configuration, both switches respond to packets.

Figure 58. Active-Active High-Availability Configuration



Although this example shows only two switches, there is no limit on the number of switches used in a redundant configuration. It is possible to implement an active-active configuration across all the VRRP-capable switches in a LAN.

Each VRRP-capable switch in an active-active configuration is autonomous. Switches in a virtual router need not be identically configured.

In the scenario illustrated in [Figure 58](#), traffic destined for IPv4 address 10.0.1.1 is forwarded through the Layer 2 switch at the top of the drawing, and ingresses VFSM 1 on port EXT1. Return traffic uses default gateway 1 (192.168.1.1).

If the link between VFSM 1 and the Layer 2 switch fails, VFSM 2 becomes the Master because it has a higher priority. Traffic is forwarded to VFSM 2, which forwards it to VFSM 1 through port EXT4. Return traffic uses default gateway 2 (192.168.2.1), and is forwarded through the Layer 2 switch at the bottom of the drawing.

To implement the active-active example, perform the following switch configuration.

## Task 1: Configure VFSM 1

### 1. Configure client and server interfaces.

```
VFSM(config)# interface ip 1
VFSM(config-ip-if)# ip address 192.168.1.100 255.255.255.0
VFSM(config-ip-if)# vlan 10
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip address 192.168.2.101 255.255.255.0
VFSM(config-ip-if)# vlan 20
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 3
VFSM(config-ip-if)# ip address 10.0.1.100 255.255.255.0
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 4
VFSM(config-ip-if)# ip address 10.0.2.101 255.255.255.0
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# exit
```

### 2. Configure the default gateways. Each default gateway points to a Layer 3 router.

```
VFSM(config)# ip gateway 1 address 192.168.1.1
VFSM(config)# ip gateway 1 enable
VFSM(config)# ip gateway 2 address 192.168.2.1
VFSM(config)# ip gateway 2 enable
```

### 3. Turn on VRRP, set up a 20 second holdoff time and configure two Virtual Interface Routers.

```
VFSM(config)# router vrrp
VFSM(config-vrrp)# enable
VFSM(config-vrrp)# holdoff 20
VFSM(config-vrrp)# virtual-router 1 virtual-router-id 1
VFSM(config-vrrp)# virtual-router 1 interface 1
VFSM(config-vrrp)# virtual-router 1 address 192.168.1.200
VFSM(config-vrrp)# virtual-router 1 enable
VFSM(config-vrrp)# virtual-router 2 virtual-router-id 2
VFSM(config-vrrp)# virtual-router 2 interface 2
VFSM(config-vrrp)# virtual-router 2 address 192.168.2.200
VFSM(config-vrrp)# virtual-router 2 enable
```

### 4. Enable tracking on ports. Set the priority of Virtual Router 1 to 101, so that it becomes the Master.

```
VFSM(config-vrrp)# virtual-router 1 track ports
VFSM(config-vrrp)# virtual-router 1 priority 101
VFSM(config-vrrp)# virtual-router 2 track ports
VFSM(config-vrrp)# exit
```

## 5. Configure ports.

```
VFSM(config)# vlan 10
VFSM(config-vlan)# exit
VFSM(config)# interface port EXT1
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk allowed vlan add 10
VFSM(config-if)# exit

VFSM(config)# vlan 20
VFSM(config-vlan)# exit
VFSM(config)# interface port EXT2
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk allowed vlan add 20
VFSM(config-if)# exit
```

## 6. Turn off Spanning Tree Protocol globally..

```
VFSM(config)# no spanning-tree stp 1
```

## Task 2: Configure VFSM 2

### 1. Configure client and server interfaces.

```
VFSM(config)# interface ip 1
VFSM(config-ip-if)# ip address 192.168.1.101 255.255.255.0
VFSM(config-ip-if)# vlan 10
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip address 192.168.2.100 255.255.255.0
VFSM(config-ip-if)# vlan 20
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 3
VFSM(config-ip-if)# ip address 10.0.1.101 255.255.255.0
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 4
VFSM(config-ip-if)# ip address 10.0.2.100 255.255.255.0
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# exit
```

### 2. Configure the default gateways. Each default gateway points to a Layer 3 router.

```
VFSM(config)# ip gateway 1 address 192.168.2.1
VFSM(config)# ip gateway 1 enable
VFSM(config)# ip gateway 2 address 192.168.1.1
VFSM(config)# ip gateway 2 enable
```

3. Turn on VRRP and configure two Virtual Interface Routers.

```
VFSM(config)# router vrrp
VFSM(config-vrrp)# enable
VFSM(config-vrrp)# virtual-router 1 virtual-router-id 1
VFSM(config-vrrp)# virtual-router 1 interface 1
VFSM(config-vrrp)# virtual-router 1 address 192.168.1.200
VFSM(config-vrrp)# virtual-router 1 enable
VFSM(config-vrrp)# virtual-router 2 virtual-router-id 2
VFSM(config-vrrp)# virtual-router 2 interface 2
VFSM(config-vrrp)# virtual-router 2 address 192.168.2.200
VFSM(config-vrrp)# virtual-router 2 enable
```

4. Enable tracking on ports. Set the priority of Virtual Router 2 to 101, so that it becomes the Master.

```
VFSM(config-vrrp)# virtual-router 1 track ports
VFSM(config-vrrp)# virtual-router 2 track ports
VFSM(config-vrrp)# virtual-router 2 priority 101
VFSM(config-vrrp)# exit
```

5. Configure ports.

```
VFSM(config)# vlan 10
VFSM(config-vlan)# exit
VFSM(config)# interface port EXT1
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk allowed vlan add 10
VFSM(config-if)# exit

VFSM(config)# vlan 20
VFSM(config-vlan)# exit
VFSM(config)# interface port EXT2
VFSM(config-if)# switchport mode trunk
VFSM(config-if)# switchport trunk allowed vlan add 20
VFSM(config-if)# exit
```

6. Turn off Spanning Tree Protocol globally.

```
VFSM(config)# no spanning-tree stp 1
```

## Hot-Standby Configuration

The primary application for VRRP-based hot-standby is to support Network Adapter Teaming on your server blades. With Network Adapter Teaming, the NICs on each server share the same IPv4 address, and are configured into a team. One NIC is the primary link, and the others are backup links. For more details, refer to the NetXen 10 Gb Ethernet Adapter documentation.

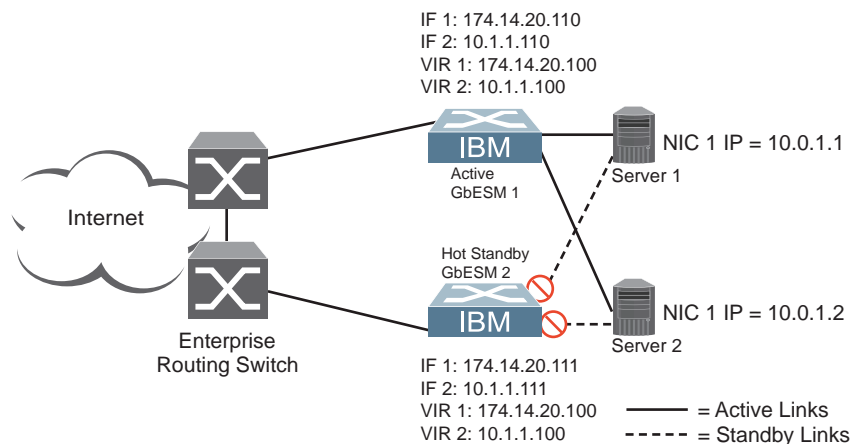
A hot-standby configuration allows all processes to failover to a standby switch if any type of failure should occur. All Virtual Interface Routers (VIRs) are bundled into one Virtual Router group, and then they failover together. When there is a failure that causes the VRRP Master to failover to the Standby, then the original primary switch temporarily disables the internal server links, which, in turn, causes the NIC teams to failover as well.

**Note:** When using hot-standby redundancy, peer switches should have an equal number of connected ports.

If hot-standby is implemented in a looped environment, the hot-standby feature automatically disables the hot-standby ports on the VRRP Standby. If the Master switch should failover to the Standby switch, it would change the hot-standby ports from *disabled* to *forwarding*, without relying on Spanning Tree or manual intervention. Therefore, Spanning Tree must be disabled.

Figure 59 illustrates a common hot-standby implementation on a single blade server. Notice that the BladeCenter server NICs are configured into a team that shares the same IPv4 address across both NICs. Because only one link can be active at a time, the hot-standby feature controls the NIC failover by having the Standby switch disable its internal ports (holding down the server links).

Figure 59. Hot-Standby Configuration



## Task 1: Configure VFSM 1

1. On VFSM 1, configure the interfaces for clients (174.14.20.110) and servers (10.1.1.110).

```
VFSM(config)# interface ip 1
VFSM(config-ip-if)# ip address 174.14.20.110      (Define IPv4 address for interface 1)
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip address 10.1.1.110         (Define IPv4 address for interface 2)
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# exit
```

2. Configure Virtual Interface Routers.

```
VFSM(config)# router vrrp
VFSM(config-vrrp)# enable
VFSM(config-vrrp)# virtual-router 1 virtual-router-id 1
VFSM(config-vrrp)# virtual-router 1 interface 1
VFSM(config-vrrp)# virtual-router 1 address 174.14.20.100
VFSM(config-vrrp)# virtual-router 1 enable
VFSM(config-vrrp)# virtual-router 2 virtual-router-id 2
VFSM(config-vrrp)# virtual-router 2 interface 2
VFSM(config-vrrp)# virtual-router 2 address 10.1.1.100
VFSM(config-vrrp)# virtual-router 2 enable
```

3. Enable VRRP Hot Standby.

```
VFSM(config-vrrp)# hot-standby      (Enable Hot Standby)
```

4. Configure VRRP Group parameters. Set the VRRP priority to 101, so that this switch is the Master.

```
VFSM(config-vrrp)# group enable      (Enable Virtual Router Group)
VFSM(config-vrrp)# group virtual-router-id 1 (Set Virtual Router ID for Group)
VFSM(config-vrrp)# group interface 1 (Set interface for Group)
VFSM(config-vrrp)# group priority 101 (Set VRRP priority to 101)
VFSM(config-vrrp)# group track ports (Enable tracking on ports)
```

5. Turn off Spanning Tree Protocol globally..

```
VFSM(config)# no spanning-tree stp 1
```

## Task 2: Configure VFSM 2

1. On VFSM 2, configure the interfaces for clients (174.14.20.111) and servers (10.1.1.111).

```
VFSM(config)# interface ip 1
VFSM(config-ip-if)# ip address 174.14.20.111      (Define IPv4 address for interface 1)
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# exit
VFSM(config)# interface ip 2
VFSM(config-ip-if)# ip address 10.1.1.111        (Define IPv4 address for interface 2)
VFSM(config-ip-if)# enable
VFSM(config-ip-if)# exit
```

2. Configure Virtual Interface Routers.

```
VFSM(config)# router vrrp
VFSM(config-vrrp)# enable
VFSM(config-vrrp)# virtual-router 1 virtual-router-id 1
VFSM(config-vrrp)# virtual-router 1 interface 1
VFSM(config-vrrp)# virtual-router 1 address 174.14.20.100
VFSM(config-vrrp)# virtual-router 1 enable
VFSM(config-vrrp)# virtual-router 2 virtual-router-id 2
VFSM(config-vrrp)# virtual-router 2 interface 2
VFSM(config-vrrp)# virtual-router 2 address 10.1.1.100
VFSM(config-vrrp)# virtual-router 2 enable
```

3. Enable VRRP Hot Standby.

```
VFSM(config-vrrp)# hot-standby
```

4. Configure VRRP Group parameters. Use the default VRRP priority of 100, so that this switch is the Standby.

```
VFSM(config-vrrp)# group enable      (Enable Virtual Router Group)
VFSM(config-vrrp)# group virtual-router-id 1 (Set Virtual Router ID for Group)
VFSM(config-vrrp)# group interface 1   (Set interface for Group)
VFSM(config-vrrp)# group track ports   (Enable tracking on ports)
```

5. Turn off Spanning Tree Protocol globally.

```
VFSM(config)# spanning-tree mode disable
```

# **Part 7: Network Management**



---

## Chapter 30. Link Layer Discovery Protocol

The IBM Networking OS software support Link Layer Discovery Protocol (LLDP). This chapter discusses the use and configuration of LLDP on the switch:

- [“LLDP Overview” on page 476](#)
- [“Enabling or Disabling LLDP” on page 477](#)
- [“LLDP Transmit Features” on page 478](#)
- [“LLDP Receive Features” on page 482](#)
- [“LLDP Example Configuration” on page 486](#)

---

## LLDP Overview

Link Layer Discovery Protocol (LLDP) is an IEEE 802.1AB-2005 standard for discovering and managing network devices. LLDP uses Layer 2 (the data link layer), and allows network management applications to extend their awareness of the network by discovering devices that are direct neighbors of already known devices.

With LLDP, the VFSM can advertise the presence of its ports, their major capabilities, and their current status to other LLDP stations in the same LAN. LLDP transmissions occur on ports at regular intervals or whenever there is a relevant change to their status. The switch can also receive LLDP information advertised from adjacent LLDP-capable network devices.

In addition to discovery of network resources, and notification of network changes, LLDP can help administrators quickly recognize a variety of common network configuration problems, such as unintended VLAN exclusions or mis-matched port aggregation membership.

The LLDP transmit function and receive function can be independently configured on a per-port basis. The administrator can allow any given port to transmit only, receive only, or both transmit and receive LLDP information.

The LLDP information to be distributed by the VFSM ports, and that which has been collected from other LLDP stations, is stored in the switch's Management Information Base (MIB). Network Management Systems (NMS) can use Simple Network Management Protocol (SNMP) to access this MIB information. LLDP-related MIB information is read-only.

Changes, either to the local switch LLDP information or to the remotely received LLDP information, are flagged within the MIB for convenient tracking by SNMP-based management systems.

For LLDP to provide expected benefits, all network devices that support LLDP should be consistent in their LLDP configuration.

### LLDP - Stacking Mode

In stacking mode, LLDP can be configured only on the ports that are not used to create the stack. The LLDP configuration menus on the stacking ports are disabled.

When configuring LLDP on a port, use the correct port syntax. See [“Rebooting Stacked Switches using the ISCLI” on page 207](#).

---

## Enabling or Disabling LLDP

### Global LLDP Setting

By default, LLDP is enabled on the VFSM. To turn LLDP off or on, use the following commands:

VFSM(config)# [no] lldp enable	<i>(Turn LLDP on or off globally)</i>
--------------------------------	---------------------------------------

### Transmit and Receive Control

The VFSM can also be configured to transmit or receive LLDP information on a port-by-port basis. By default, when LLDP is globally enabled on the switch, VFSM ports transmit and receive LLDP information (see the `tx_rx` option below). To change the LLDP transmit and receive state, the following commands are available:

VFSM(config)# interface port <x>	<i>(Select a switch port)</i>
VFSM(config-if)# lldp admin-status tx_rx	<i>(Transmit and receive LLDP)</i>
VFSM(config-if)# lldp admin-status tx_only	<i>(Only transmit LLDP)</i>
VFSM(config-if)# lldp admin-status rx_only	<i>(Only receive LLDP)</i>
VFSM(config-if)# no lldp admin-status	<i>(Do not participate in LLDP)</i>
VFSM(config-if)# exit	<i>(Exit port mode)</i>

To view the LLDP transmit and receive status, use the following commands:

VFSM(config)# show lldp port	<i>(status of all ports)</i>
VFSM(config)# show interface port <n> lldp	<i>(status of selected port)</i>

---

## LLDP Transmit Features

Numerous LLDP transmit options are available, including scheduled and minimum transmit interval, expiration on remote systems, SNMP trap notification, and the types of information permitted to be shared.

**Note:** In stacking mode, only the stack Master transmits LLDP information for all the ports in a stack. The egress port's MAC address is used as the source address in the LLDP packets.

### Scheduled Interval

The VFSM can be configured to transmit LLDP information to neighboring devices once each 5 to 32768 seconds. The scheduled interval is global; the same interval value applies to all LLDP transmit-enabled ports. However, to help balance LLDP transmissions and keep them from being sent simultaneously on all ports, each port maintains its own interval clock, based on its own initialization or reset time. This allows switch-wide LLDP transmissions to be spread out over time, though individual ports comply with the configured interval.

The global transmit interval can be configured using the following command:

```
VFSM(config)# lldp refresh-interval <interval>
```

where *interval* is the number of seconds between LLDP transmissions. The range is 5 to 32768. The default is 30 seconds.

### Minimum Interval

In addition to sending LLDP information at scheduled intervals, LLDP information is also sent when the VFSM detects relevant changes to its configuration or status (such as when ports are enabled or disabled). To prevent the VFSM from sending multiple LLDP packets in rapid succession when port status is in flux, a transmit delay timer can be configured.

The transmit delay timer represents the minimum time permitted between successive LLDP transmissions on a port. Any interval-driven or change-driven updates will be consolidated until the configured transmit delay expires.

The minimum transmit interval can be configured using the following command:

```
VFSM(config)# lldp transmission-delay <interval>
```

where *interval* is the minimum number of seconds permitted between successive LLDP transmissions on any port. The range is 1 to one-quarter of the scheduled transmit interval (`lldp refresh-interval <value>`), up to 8192. The default is 2 seconds.

## Time-to-Live for Transmitted Information

The transmitted LLDP information is held by remote systems for a limited time. A time-to-live parameter allows the switch to determine how long the transmitted data should be held before it expires. The hold time is configured as a multiple of the configured transmission interval.

```
VFSM(config)# lldp holdtime-multiplier <multiplier>
```

where *multiplier* is a value between 2 and 10. The default value is 4, meaning that remote systems will hold the port's LLDP information for 4 x the 30-second msgtxint value, or 120 seconds, before removing it from their MIB.

## Trap Notifications

If SNMP is enabled on the VFSM (see [“Using Simple Network Management Protocol” on page 39](#)), each port can be configured to send SNMP trap notifications whenever LLDP transmissions are sent. By default, trap notification is disabled for each port. The trap notification state can be changed using the following commands:

```
VFSM(config)# interface port <rt>
VFSM(config-if)# [no] lldp trap-notification
VFSM(config-if)# exit
```

In addition to sending LLDP information at scheduled intervals, LLDP information is also sent when the VFSM detects relevant changes to its configuration or status (such as when ports are enabled or disabled). To prevent the VFSM from sending multiple trap notifications in rapid succession when port status is in flux, a global trap delay timer can be configured.

The trap delay timer represents the minimum time permitted between successive trap notifications on any port. Any interval-driven or change-driven trap notices from the port will be consolidated until the configured trap delay expires.

The minimum trap notification interval can be configured using the following command:

```
VFSM(config)# lldp trap-notification-interval <interval>
```

where *interval* is the minimum number of seconds permitted between successive LLDP transmissions on any port. The range is 1 to 3600. The default is 5 seconds.

If SNMP trap notification is enabled, the notification messages can also appear in the system log. This is enabled by default. To change whether the SNMP trap notifications for LLDP events appear in the system log, use the following commands:

```
VFSM(config)# [no] logging log lldp
```

## Changing the LLDP Transmit State

When the port is disabled, or when LLDP transmit is turned off for the port using the `admstat` command's `rx_only` or `disabled` options (see [“Transmit and Receive Control” on page 477](#)), a final LLDP packet is transmitted with a time-to-live value of 0. Neighbors that receive this packet will remove the LLDP information associated with the VFSM port from their MIB.

In addition, if LLDP is fully disabled on a port (using `admstat disabled`) and later re-enabled, the VFSM will temporarily delay resuming LLDP transmissions on the port in order to allow the port LLDP information to stabilize. The reinitialization delay interval can be globally configured for all ports using the following command:

```
VFSM(config)# lldp reinit-delay <interval>
```

where *interval* is the number of seconds to wait before resuming LLDP transmissions. The range is between 1 and 10. The default is 2 seconds.

## Types of Information Transmitted

When LLDP transmission is permitted on the port (see [“Enabling or Disabling LLDP” on page 477](#)), the port advertises the following required information in type/length/value (TLV) format:

- Chassis ID
- Port ID
- LLDP Time-to-Live

LLDP transmissions can also be configured to enable or disable inclusion of optional information, using the following commands:

```
VFSM(config)# interface port <x>
VFSM(config-if)# [no] lldp tlv <type>
VFSM(config-if)# exit
```

where *type* is an LLDP information option from [Table 36](#):

Table 36. LLDP Optional Information Types

Type	Description	Default
portdesc	Port Description	Enabled
sysname	System Name	Enabled
sysdescr	System Description	Enabled
syscap	System Capabilities	Enabled
mgmtaddr	Management Address	Enabled
portvid	IEEE 802.1 Port VLAN ID	Disabled
portprot	IEEE 802.1 Port and Protocol VLAN ID	Disabled
vlanname	IEEE 802.1 VLAN Name	Disabled
protid	IEEE 802.1 Protocol Identity	Disabled

Table 36. LLDP Optional Information Types (continued)

Type	Description	Default
macphy	IEEE 802.3 MAC/PHY Configuration/Status, including the auto-negotiation, duplex, and speed status of the port.	Disabled
powermdi	IEEE 802.3 Power via MDI, indicating the capabilities and status of devices that require or provide power over twisted-pair copper links.	Disabled
linkaggr	IEEE 802.3 Link Aggregation status for the port.	Disabled
framesz	IEEE 802.3 Maximum Frame Size for the port.	Disabled
dcbx	Data Center Bridging Capability Exchange Protocol (DCBX) for the port.	Enabled
all	Select all optional LLDP information for inclusion or exclusion.	Disabled

---

## LLDP Receive Features

### Types of Information Received

When the LLDP receive option is enabled on a port (see [“Enabling or Disabling LLDP” on page 477](#)), the port may receive the following information from LLDP-capable remote systems:

- Chassis Information
- Port Information
- LLDP Time-to-Live
- Port Description
- System Name
- System Description
- System Capabilities Supported/Enabled
- Remote Management Address

The VFSM stores the collected LLDP information in the MIB. Each remote LLDP-capable device is responsible for transmitting regular LLDP updates. If the received updates contain LLDP information changes (to port state, configuration, LLDP MIB structures, deletion), the switch will set a change flag within the MIB for convenient notification to SNMP-based management systems.

**Note:** In stacking mode, both the Master and the Backup receive LLDP information for all the ports in a stack and update the LLDP table. The Master and Backup switches synchronize the LLDP tables.

### Viewing Remote Device Information

LLDP information collected from neighboring systems can be viewed in numerous ways:

- Using a centrally-connected LLDP analysis server
- Using an SNMP agent to examine the VFSM MIB
- Using the VFSM Browser-Based Interface (BBI)
- Using CLI or isCLI commands on the VFSM

Using the ISCLI the following command displays remote LLDP information:

```
VFSM(config)# show lldp remote-device [<index number>]
```

To view a summary of remote information, omit the *Index number* parameter. For example:

```
VFSM(config)# show lldp remote-device
LLDP Remote Devices Information
Legend(possible values in DMAC column) :
NB - Nearest Bridge - 01-80-C2-00-00-0E
NnTB - Nearest non-TPMR Bridge - 01-80-C2-00-00-03
NCB - Nearest Customer Bridge - 01-80-C2-00-00-00
Total number of current entries: 2
```

LocalPort	Index	Remote Chassis ID	Remote Port	Remote System Name	DMAC
MGT	1	00 00 00 00 11 00	5	C12	NB
20	2	74 99 75 1c 75 00	44	C3+	NB

To view detailed information for a remote device, specify the *Index number* as found in the summary. For example, in keeping with the sample summary, to list details for the first remote device (with an Index value of 1), use the following command:

```
VFSM(config)# show lldp remote-device 1
Local Port Alias: EXT3
    Remote Device Index      : 1
    Remote Device TTL        : 99
    Remote Device RxChanges  : false
    Chassis Type             : Mac Address
    Chassis Id               : 00-18-b1-33-1d-00
    Port Type                : Locally Assigned
    Port Id                  : 23
    Port Description         : EXT7

    System Name              :
    System Description       : IBM Networking Operating System IBM Virtual Fabric
10Gb Switch Module for IBM BladeCenter, IBM Networking OS: version 7.8, Boot
Image: version 6.9.1.14

    System Capabilities Supported : bridge, router
    System Capabilities Enabled   : bridge, router

    Remote Management Address:
        Subtype                 : IPv4
        Address                  : 10.100.120.181
        Interface Subtype       : ifIndex
        Interface Number        : 128
        Object Identifier       :
```

**Note:** Received LLDP information can change very quickly. When using show commands, it is possible that flags for some expected events may be too short-lived to be observed in the output.

To view detailed information of all remote devices, use the following command:

```
VFSM(config)# show lldp remote-device detail
Local Port Alias: MGTA
    Remote Device Index          : 1
    Remote Device TTL            : 4678
    Remote Device RxChanges      : false
    Chassis Type                 : Mac Address
    Chassis Id                   : 08-17-f4-a1-db-00
    Port Type                    : Locally Assigned
    Port Id                      : 25
    Port Description              : MGTA

    System Name                  :
    System Description           : IBM Networking Operating System IBM
Virtual Fabric 10Gb Switch Module for IBM BladeCenter, IBM Networking OS: version
7.8, Boot Image: version 6.9.1.14
    System Capabilities Supported : bridge, router
    System Capabilities Enabled   : bridge, router

    Remote Management Address:
        Subtype                  : IPv4
        Address                   : 10.38.22.23
        Interface Subtype        : ifIndex
        Interface Number         : 127
        Object Identifier        :

Local Port Alias: 2
    Remote Device Index          : 2
    Remote Device TTL            : 4651
    Remote Device RxChanges      : false
    Chassis Type                 : Mac Address
    Chassis Id                   : 08-17-f4-a1-db-00
    Port Type                    : Locally Assigned
    Port Id                      : 2
    Port Description              : 2

    System Name                  :
    System Description           : IBM Networking Operating System IBM
Virtual Fabric 10Gb Switch Module for IBM BladeCenter, IBM Networking OS: version
7.8, Boot Image: version 6.9.1.14
    System Capabilities Supported : bridge, router
    System Capabilities Enabled   : bridge, router

    Remote Management Address:
        Subtype                  : IPv4
        Address                   : 10.38.22.23
        Interface Subtype        : ifIndex
        Interface Number         : 127
        Object Identifier        :

Total entries displayed:  2
```

## Time-to-Live for Received Information

Each remote device LLDP packet includes an expiration time. If the switch port does not receive an LLDP update from the remote device before the time-to-live clock expires, the switch will consider the remote information to be invalid, and will remove all associated information from the MIB.

Remote devices can also intentionally set their LLDP time-to-live to 0, indicating to the switch that the LLDP information is invalid and should be immediately removed.

---

## LLDP Example Configuration

1. Turn LLDP on globally.

```
VFSM(config)# lldp enable
```

2. Set the global LLDP timer features.

```
VFSM(config)# lldp refresh-interval 30      (Transmit each 30 seconds)
VFSM(config)# lldp transmission-delay 2     (No more often than 2 sec.)
VFSM(config)# lldp holdtime-multiplier 4    (Remote hold 4 intervals)
VFSM(config)# lldp reinit-delay 2           (Wait 2 sec. after reinit.)
VFSM(config)# lldp trap-notification-interval 5 (Minimum 5 sec. between)
```

3. Set LLDP options for each port.

```
VFSM(config)# interface port <n>           (Select a switch port)
VFSM(config-if)# lldp admin-status tx_rx    (Transmit and receive LLDP)
VFSM(config-if)# lldp trap-notification     (Enable SNMP trap notifications)
VFSM(config-if)# lldp tlv all               (Transmit all optional information)
VFSM(config-if)# exit
```

4. Enable syslog reporting.

```
VFSM(config)# logging log lldp
```

5. Verify the configuration settings:

```
VFSM(config)# show lldp
```

6. View remote device information as needed.

```
VFSM(config)# show lldp remote-device
or
VFSM(config)# show lldp remote-device <index number>
or
VFSM(config)# show lldp remote-devices detail
```

---

## Chapter 31. Simple Network Management Protocol

IBM Networking OS provides Simple Network Management Protocol (SNMP) version 1, version 2, and version 3 support for access through any network management software, such as IBM Director or HP-OpenView.

---

## SNMP Version 1 & Version 2

To access the SNMP agent on the VFSM, the read and write community strings on the SNMP manager should be configured to match those on the switch.

The read and write community strings on the switch can be changed using the following commands on the ISCLI:

```
VFSM(config)# snmp-server read-community <1-32 characters>  
  
-and-  
  
VFSM(config)# snmp-server write-community <1-32 characters>
```

The SNMP manager should be able to reach the management interface or any one of the IP interfaces on the switch.

For the SNMP manager to receive the SNMPv1 traps sent out by the SNMP agent on the switch, configure the trap host on the switch with the following command:

```
VFSM(config)# snmp-server trap-source <trap source IP interface>  
VFSM(config)# snmp-server host <IPv4 address> <trap host community string>
```

**Note:** You can use a loopback interface to set the source IP address for SNMP traps. Use the following command to apply a configured loopback interface:

```
VFSM(config)# snmp-server trap-source loopback <1-5>
```

---

## SNMP Version 3

SNMP version 3 (SNMPv3) is an enhanced version of the Simple Network Management Protocol, approved by the Internet Engineering Steering Group in March, 2002. SNMPv3 contains additional security and authentication features that provide data origin authentication, data integrity checks, timeliness indicators and encryption to protect against threats such as masquerade, modification of information, message stream modification and disclosure.

SNMPv3 allows clients to query the MIBs securely.

SNMPv3 configuration is managed using the following menu:

```
VFSM(config)# snmp-server ?
```

For more information on SNMP MIBs and the commands used to configure SNMP on the switch, see the *IBM Networking OS 7.8 Command Reference*.

### Default Configuration

IBM N/OS has three SNMPv3 users by default. All the three users have access to all the MIBs supported by the switch:

- User 1 name is `adminmd5` (password `adminmd5`). Authentication used is MD5. Privacy protocol used is DES.
- User 2 name is `adminsha` (password `adminsha`). Authentication used is SHA. Privacy protocol used is DES.
- User 3 name is `adminshaaes` (password `Edpq132x!#9Zpx432w`). Authentication used is SHA. Privacy protocol used is AES-128.

In boot strict mode (See [“Boot Strict Mode” on page 46](#)), IBM N/OS has only one SNMPv3 user:

- User name is `adminshaaes` (password `Edpq132x!#9Zpx432w`). Authentication used is SHA. Privacy protocol used is AES-128.

Up to 16 SNMP users can be configured on the switch. To modify an SNMP user, enter the following command:

```
VFSM(config)# snmp-server user <1-16> name <1-32 characters>
```

Users can be configured to use the authentication/privacy options. The VFSM support two authentication algorithms: MD5 and SHA, as specified in the following command:

```
VFSM(config)# snmp-server user <1-16> authentication-protocol {md5|sha}
authentication-password

-Or-

VFSM(config)# snmp-server user <1-16> authentication-protocol none
```

## User Configuration Example

1. To configure a user with name “admin,” authentication type MD5, and authentication password of “admin,” privacy option DES with privacy password of “admin,” use the following ISCLI commands.

```
VFSM(config)# snmp-server user 5 name admin
VFSM(config)# snmp-server user 5 authentication-protocol md5
authentication-password
Changing authentication password; validation required:
Enter current admin password:          <admin. password>
Enter new authentication password:     <auth. password>
Re-enter new authentication password:  <auth. password>
New authentication password accepted.

VFSM(config)# snmp-server user 5 privacy-protocol des privacy-password
Changing privacy password; validation required:
Enter current admin password:          <admin. password>
Enter new privacy password:            <privacy password>
Re-enter new privacy password:         <privacy password>
New privacy password accepted.
```

2. Configure a user access group, along with the views the group may access. Use the access table to configure the group’s access level.

```
VFSM(config)# snmp-server access 5 name admingrp
VFSM(config)# snmp-server access 5 level authpriv
VFSM(config)# snmp-server access 5 read-view iso
VFSM(config)# snmp-server access 5 write-view iso
VFSM(config)# snmp-server access 5 notify-view iso
```

Because the read view, write view, and notify view are all set to “iso,” the user type has access to all private and public MIBs.

3. Assign the user to the user group. Use the group table to link the user to a particular access group.

```
VFSM(config)# snmp-server group 5 user-name admin
VFSM(config)# snmp-server group 5 group-name admingrp
```

If you want to allow user access only to certain MIBs, see “View-Based Configuration,” next.

## View-Based Configurations

- Switch User equivalent

To configure an SNMP user equivalent to the switch “user” login, use the following configuration:

```
(Configure the user)
VFSM(config)# snmp-server user 4 name usr

(Configure access group 3)
VFSM(config)# snmp-server access 3 name usrgrp
VFSM(config)# snmp-server access 3 read-view usr
VFSM(config)# snmp-server access 3 write-view usr
VFSM(config)# snmp-server access 3 notify-view usr

(Assign user to access group 3)
VFSM(config)# snmp-server group 3 user-name usr
VFSM(config)# snmp-server group 3 group-name usrgrp

(Create views for user)
VFSM(config)# snmp-server view 6 name usr
VFSM(config)# snmp-server view 6 tree 1.3.6.1.4.1.1872.2.5.1.2
(Agent information)
VFSM(config)# snmp-server view 7 name usr
VFSM(config)# snmp-server view 7 tree 1.3.6.1.4.1.1872.2.5.1.3
(L2 statistics)
VFSM(config)# snmp-server view 8 name usr
VFSM(config)# snmp-server view 8 tree 1.3.6.1.4.1.1872.2.5.2.2
(L2 information)
VFSM(config)# snmp-server view 9 name usr
VFSM(config)# snmp-server view 9 tree 1.3.6.1.4.1.1872.2.5.2.3
(L3 statistics)
VFSM(config)# snmp-server view 10 name usr
VFSM(config)# snmp-server view 10 tree 1.3.6.1.4.1.1872.2.5.2.3
(L3 information)
VFSM(config)# snmp-server view 11 name usr
VFSM(config)# snmp-server view 11 tree 1.3.6.1.4.1.1872.2.5.3.3
```

- Switch Oper equivalent

```

VFSM(config)# snmp-server user 5 name usr                                (Configure the user)
VFSM(config)# snmp-server access 4 name opergrp                        (Configure access group 3)
VFSM(config)# snmp-server access 4 read-view oper
VFSM(config)# snmp-server access 4 write-view oper
VFSM(config)# snmp-server access 4 notify-view oper
VFSM(config)# snmp-server group 4 user-name oper                      (Assign oper to access group 4)
VFSM(config)# snmp-server group 4 group-name opergrp
VFSM(config)# snmp-server view 20 name oper                          (Create views for oper)
VFSM(config)# snmp-server view 20 tree 1.3.6.1.4.1.1872.2.5.1.2
VFSM(config)# snmp-server view 21 name oper                          (Agent information)
VFSM(config)# snmp-server view 21 tree 1.3.6.1.4.1.1872.2.5.1.3
VFSM(config)# snmp-server view 22 name oper                          (L2 statistics)
VFSM(config)# snmp-server view 22 tree 1.3.6.1.4.1.1872.2.5.2.2
VFSM(config)# snmp-server view 23 name oper                          (L2 information)
VFSM(config)# snmp-server view 23 tree 1.3.6.1.4.1.1872.2.5.2.3
VFSM(config)# snmp-server view 24 name oper                          (L3 statistics)
VFSM(config)# snmp-server view 24 tree 1.3.6.1.4.1.1872.2.5.2.3
VFSM(config)# snmp-server view 25 name oper                          (L3 information)
VFSM(config)# snmp-server view 25 tree 1.3.6.1.4.1.1872.2.5.3.3

```

---

## Configuring SNMP Trap Hosts

### SNMPv1 Trap Host

1. Configure a user with no authentication and password.

```
VFSM(config)# snmp-server user 10 name vltrap
```

2. Configure an access group and group table entries for the user. Use the following menu to specify which traps can be received by the user:

```
VFSM(config)# snmp-server access <user number>
```

In the following example the user will receive the traps sent by the switch.

```
VFSM(config)# snmp-server access 10          (Access group to view SNMPv1 traps)
name vltrap
security snmpv1
notify-view iso
VFSM(config)# snmp-server group 10          (Assign user to the access group)
security snmpv1
user-name vltrap
group-name vltrap
```

3. Configure an entry in the notify table.

```
VFSM(config)# snmp-server notify 10 name vltrap
VFSM(config)# snmp-server notify 10 tag vltrap
```

4. Specify the IPv4 address and other trap parameters in the `targetAddr` and `targetParam` tables. Use the following menus to specify the user name associated with the `targetParam` table:

```
VFSM(config)# snmp-server target-address 10 name vltrap address 10.70.70.190
VFSM(config)# snmp-server target-address 10 parameters-name vlparam
VFSM(config)# snmp-server target-address 10 taglist vlparam
VFSM(config)# snmp-server target-parameters 10 name vlparam
VFSM(config)# snmp-server target-parameters 10 user-name vlonly
VFSM(config)# snmp-server target-parameters 10 message snmpv1
```

5. Use the community table to specify which community string is used in the trap.

```
VFSM(config)# snmp-server community 10      ( Define the community string)
index vltrap
name public
user-name vltrap
```

## SNMPv2 Trap Host Configuration

The SNMPv2 trap host configuration is similar to the SNMPv1 trap host configuration. Wherever you specify the model, use `snmpv2` instead of `snmpv1`.

```
VFSM(config)# snmp-server user 10 name v2trap

VFSM(config)# snmp-server group 10 security snmpv2
VFSM(config)# snmp-server group 10 user-name v2trap
VFSM(config)# snmp-server group 10 group-name v2trap
VFSM(config)# snmp-server access 10 name v2trap
VFSM(config)# snmp-server access 10 security snmpv2
VFSM(config)# snmp-server access 10 notify-view iso

VFSM(config)# snmp-server notify 10 name v2trap
VFSM(config)# snmp-server notify 10 tag v2trap

VFSM(config)# snmp-server target-address 10 name v2trap
                address 100.10.2.1
VFSM(config)# snmp-server target-address 10 taglist v2trap
VFSM(config)# snmp-server target-address 10 parameters-name
                v2param
VFSM(config)# snmp-server target-parameters 10 name v2param
VFSM(config)# snmp-server target-parameters 10 message snmpv2c
VFSM(config)# snmp-server target-parameters 10 user-name v2trap
VFSM(config)# snmp-server target-parameters 10 security snmpv2

VFSM(config)# snmp-server community 10 index v2trap
VFSM(config)# snmp-server community 10 user-name v2trap
```

## SNMPv3 Trap Host Configuration

To configure a user for SNMPv3 traps, you can choose to send the traps with both privacy and authentication, with authentication only, or without privacy or authentication.

This is configured in the access table using the following commands:

```
VFSM(config)# snmp-server access <1-32> level [authNoPriv | authPriv | noAuthNoPriv]
VFSM(config)# snmp-server target-parameters <1-16>
```

Configure the user in the user table accordingly.

It is not necessary to configure the community table for SNMPv3 traps because the community string is not used by SNMPv3.

The following example shows how to configure a SNMPv3 user `v3trap` with authentication only:

```
VFSM(config)# snmp-server user 11 name v3trap
VFSM(config)# snmp-server user 11 authentication-protocol md5
authentication-password
Changing authentication password; validation required:
Enter current admin password:          <admin.password>
Enter new authentication password:     <auth.password>
Re-enter new authentication password:  <auth.password>
New authentication password accepted.
VFSM(config)# snmp-server access 11 notify-view iso
VFSM(config)# snmp-server access 11 level authnopriv
VFSM(config)# snmp-server group 11 user-name v3trap
VFSM(config)# snmp-server group 11 tag v3trap
VFSM(config)# snmp-server notify 11 name v3trap
VFSM(config)# snmp-server notify 11 tag v3trap
VFSM(config)# snmp-server target-address 11 name v3trap address 47.81.25.66
VFSM(config)# snmp-server target-address 11 taglist v3trap
VFSM(config)# snmp-server target-address 11 parameters-name v3param
VFSM(config)# snmp-server target-parameters 11 name v3param
VFSM(config)# snmp-server target-parameters 11 user-name v3trap
VFSM(config)# snmp-server target-parameters 11 level authNoPriv
```

---

## SNMP MIBs

The N/OS SNMP agent supports SNMP version 3. Security is provided through SNMP community strings. The default community strings are “public” for SNMP GET operation and “private” for SNMP SET operation. The community string can be modified only through the Command Line Interface (CLI). Detailed SNMP MIBs and trap definitions of the N/OS SNMP agent are contained in the following N/OS enterprise MIB document:

GbESM-24-10G-L2L3.mib

The N/OS SNMP agent supports the following standard MIBs:

- dot1x.mib
- ieee8021ab.mib
- ieee8023ad.mib
- lldpxdcbx.mib
- rfc1213.mib
- rfc1215.mib
- rfc1493.mib
- rfc1573.mib
- rfc1643.mib
- rfc1657.mib
- rfc1757.mib
- rfc1850.mib
- rfc1907.mib
- rfc2037.mib
- rfc2233.mib
- rfc2465.mib
- rfc2571.mib
- rfc2572.mib
- rfc2573.mib
- rfc2574.mib
- rfc2575.mib
- rfc2576.mib
- rfc2790.mib
- rfc3176.mib
- rfc4133.mib
- rfc4363.mib

The N/OS SNMP agent supports the following generic traps as defined in RFC 1215:

- ColdStart
- WarmStart
- LinkDown
- LinkUp
- AuthenticationFailure

The SNMP agent also supports two Spanning Tree traps as defined in RFC 1493:

- NewRoot
- TopologyChange

The following are the enterprise SNMP traps supported in N/OS:

*Table 37. IBM N/OS-Supported Enterprise SNMP Traps*

Trap Name	Description
altSwDefGwUp	Signifies that the default gateway is alive.
altSwDefGwDown	Signifies that the default gateway is down.
altSwDefGwInService	Signifies that the default gateway is up and in service
altSwDefGwNotInService	Signifies that the default gateway is alive but not in service
altSwVrrpNewMaster	Indicates that the sending agent has transitioned to "Master" state.
altSwVrrpNewBackup	Indicates that the sending agent has transitioned to "Backup" state.
altSwVrrpAuthFailure	Signifies that a packet has been received from a router whose authentication key or authentication type conflicts with this router's authentication key or authentication type. Implementation of this trap is optional.
altSwNtpNotServer	Signifies that the primary or secondary NTP server cannot be contacted.
altSwNtpUpdateClock	Signifies that an NTP update was received.
altSwECMPGatewayUp	Signifies that the ECMP gateway is up.
altSwECMPGatewayDown	Signifies that the ECMP gateway is down.
altSwOspfRouteUpdated	Signifies that an OSPF route message was received.
altSwLoginFailure	Signifies that someone failed to enter a valid username/password combination.
altSwTempExceedThreshold	Signifies that the switch temperature has exceeded maximum safety limits.
altSwTempReturnThreshold	Signifies that the switch temperature has returned below maximum safety limits.
altSwAcloPNotification	Signifies that traffic exceeds the committed rate for an ACL.

Table 37. IBM N/OS-Supported Enterprise SNMP Traps (continued)

Trap Name	Description
altSwStgNewRoot	Signifies that the bridge has become the new root of the STG.
altSwStgTopologyChanged	Signifies that there was a STG topology change.
altSwStgBlockingState	An altSwStgBlockingState trap is sent when port state is changed in blocking state.
altSwCistNewRoot	Signifies that the bridge has become the new root of the CIST.
altSwCistTopologyChanged	Signifies that there was a CIST topology change.
altSwHotlinksMasterUp	Signifies that the Master interface is active.
altSwHotlinksMasterDn	Signifies that the Master interface is not active.
altSwHotlinksBackupUp	Signifies that the Backup interface is active.
altSwHotlinksBackupDn	Signifies that the Backup interface is not active.
altSwHotlinksNone	Signifies that there are no active interfaces.
altSwTeamingCtrlUp	Signifies that the teaming control is up.
altSwTeamingCtrlDown	Signifies that the teaming control is down.
altSwTeamingCtrlDownTearDownBlocked	Signifies that the teaming control is down, but teardown is blocked.
altSwTeamingCtrlError	Signifies error. Action is undefined.
altSwLACPPortBlocked	Signifies that LACP is operationally down on a port, and traffic is blocked on the port.
altSwLACPPortUnblocked	Signifies that LACP is operationally up on a port, and traffic is no longer blocked on the port.
altSwValidLogin	Signifies that a user login has occurred.
altSwValidLogout	Signifies that a user logout has occurred.
altVMGroupVMotion	Signifies that a virtual machine has moved from a port to another.

Table 37. IBM N/OS-Supported Enterprise SNMP Traps (continued)

Trap Name	Description
altVMGroupVMOnline	Signifies that a advance provisioned virtual machine has came online.
altVMGroupVMVlanChange	Signifies that a virtual machine has entered into a VLAN, or changed the VLAN.
altSwApplyComplete	Signifies that a new configuration has been applied.
altSwSaveComplete	Signifies that a new configuration has been saved.
altSwFwDownloadSuccess	Signifies that firmware has been downloaded to image1/image2/boot image.
altSwFwDownloadFailure	Signifies that a firmware download to image1/image2/boot image has failed.
altSwDefAdminDisable	Signifies that the default admin account has been disabled.
altSwAcntStrngPswdNotMet	Signifies that the configured password does not match strong password complexity.
altSwAcntLocked	Signifies that the account has been locked.
altSwAcntUnlocked	Signifies that the account has been unlocked.
altSwStackSwitchAttached	Signifies that a new switch has attached to the stack.
altSwStackSwitchDettached	Signifies that a new switch has detached from the stack.
altSwStackBackupPresent	Signifies that a new backup has been set.
altSwStackBackupGone	Signifies that the backup switch has been made unavailable.
altSwStackMasterAfterInit	Signifies that the switch has become master after initialization.
altSwStackMasterFromBackup	Signifies that the switch has become master from backup.
altSwStackDuplicateJoinAttempt	Signifies that a new switch with duplicate UUID/bay has tried to join the stack.
altSwStackLinkUp	Signifies that a stack link went up.

Table 37. IBM N/OS-Supported Enterprise SNMP Traps (continued)

Trap Name	Description
altSwStackLinkDown	Signifies that a stack link went down.
altSwStackXferError	Signifies that a transfer between the master and a member has terminated with error.
altSwStackXferSuccess	Signifies that a transfer between the master and a member has terminated with no errors.
altSwStackSwitchTypeMismatch	Signifies that a new switch of different type has attempted to join the stack.
altSwStackImageSlotMismatch	Signifies that the slot of the boot image of a newly attached switch does not match that of the master.
altSwStackImageVersMismatch	Signifies that the version of the boot image of a newly attached switch does not match that of the master.
altSwStackChassisTypeMismatch	Signifies that a new switch in a chassis of different and incompatible type has attempted to join the stack.
altSwStackBcsBayMismatch	Signifies that a new switch in a BCS chassis in a bay with different and incompatible port mapping has attempted to join the stack.
altSwStackBootCfgMismatch	Signifies that the booted config of a newly attached switch does not match that of the master.
altSwStackNvramMasterJoin	Signifies that a switch which was configured as master in NVRAM was attached to the stack.
altSwStackForceDetach	Signifies that the master has sent a FORCE DETACH message to a member.
vmCheckSpoofedvm	Signifies that a spoofed VM MAC was found.

---

## Switch Images and Configuration Files

This section describes how to use MIB calls to work with switch images and configuration files. You can use a standard SNMP tool to perform the actions, using the MIBs listed in [Table 38](#).

[Table 38](#) lists the MIBs used to perform operations associated with the Switch Image and Configuration files.

*Table 38. MIBs for Switch Image and Configuration Files*

MIB Name	MIB OID
agTransferServer	1.3.6.1.4.26543.2.5.1.1.7.1.0
agTransferImage	1.3.6.1.4.26543.2.5.1.1.7.2.0
agTransferImageFileName	1.3.6.1.4.26543.2.5.1.1.7.3.0
agTransferCfgFileName	1.3.6.1.4.26543.2.5.1.1.7.4.0
agTransferDumpFileName	1.3.6.1.4.26543.2.5.1.1.7.5.0
agTransferAction	1.3.6.1.4.26543.2.5.1.1.7.6.0
agTransferLastActionStatus	1.3.6.1.4.26543.2.5.1.1.7.7.0
agTransferUserName	1.3.6.1.4.26543.2.5.1.1.7.9.0
agTransferPassword	1.3.6.1.4.1.26543.2.5.1.1.7.10.0
agTransferTSDumpFileName	1.3.6.1.4.1.26543.2.5.1.1.7.11.0

The following SNMP actions can be performed using the MIBs listed in [Table 38](#).

- Load a new Switch image (boot or running) from a FTP/TFTP server
- Load a previously saved switch configuration from a FTP/TFTP server
- Save the switch configuration to a FTP/TFTP server
- Save a switch dump to a FTP/TFTP server

## Loading a New Switch Image

To load a new switch image with the name "MyNewImage-1.img" into image2, follow the steps below. This example shows an FTP/TFTP server at IPv4 address 192.168.10.10, though IPv6 is also supported.

1. Set the FTP/TFTP server address where the switch image resides:

```
Set agTransferServer.0 "192.168.10.10"
```

2. Set the area where the new image will be loaded:

```
Set agTransferImage.0 "image2"
```

3. Set the name of the image:

```
Set agTransferImageFileName.0 "MyNewImage-1.img"
```

4. If you are using an FTP server, enter a username:

```
Set agTransferUserName.0 "MyName"
```

5. If you are using an FTP server, enter a password:

```
Set agTransferPassword.0 "MyPassword"
```

6. Initiate the transfer. To transfer a switch image, enter 2 (gting):

```
Set agTransferAction.0 "2"
```

## Loading a Saved Switch Configuration

To load a saved switch configuration with the name "MyRunningConfig.cfg" into the switch, follow the steps below. This example shows a TFTP server at IPv4 address 192.168.10.10, though IPv6 is also supported.

1. Set the FTP/TFTP server address where the switch Configuration File resides:

```
Set agTransferServer.0 "192.168.10.10"
```

2. Set the name of the configuration file:

```
Set agTransferCfgFileName.0 "MyRunningConfig.cfg"
```

3. If you are using an FTP server, enter a username:

```
Set agTransferUserName.0 "MyName"
```

4. If you are using an FTP server, enter a password:

```
Set agTransferPassword.0 "MyPassword"
```

5. Initiate the transfer. To restore a running configuration, enter 3:

```
Set agTransferAction.0 "3"
```

## Saving the Switch Configuration

To save the switch configuration to a FTP/TFTP server follow the steps below. This example shows a FTP/TFTP server at IPv4 address 192.168.10.10, though IPv6 is also supported.

1. Set the FTP/TFTP server address where the configuration file is saved:

```
Set agTransferServer.0 "192.168.10.10"
```

2. Set the name of the configuration file:

```
Set agTransferCfgFileName.0 "MyRunningConfig.cfg"
```

3. If you are using an FTP server, enter a username:  
`Set agTransferUserName.0 "MyName"`
4. If you are using an FTP server, enter a password:  
`Set agTransferPassword.0 "MyPassword"`
5. Initiate the transfer. To save a running configuration file, enter 4:  
`Set agTransferAction.0 "4"`

## **Saving a Switch Dump**

To save a switch dump to a FTP/TFTP server, follow the steps below. This example shows an FTP/TFTP server at 192.168.10.10, though IPv6 is also supported.

1. Set the FTP/TFTP server address where the configuration will be saved:  
`Set agTransferServer.0 "192.168.10.10"`
2. Set the name of dump file:  
`Set agTransferDumpFileName.0 "MyDumpFile.dmp"`
3. If you are using an FTP server, enter a username:  
`Set agTransferUserName.0 "MyName"`
4. If you are using an FTP server, enter a password:  
`Set agTransferPassword.0 "MyPassword"`
5. Initiate the transfer. To save a dump file, enter 5:  
`Set agTransferAction.0 "5"`



# Part 8: Monitoring

The ability to monitor traffic passing through the VFSM can be invaluable for troubleshooting some types of networking problems. This sections cover the following monitoring features:

- Remote Monitoring (RMON)
- sFLOW
- Port Mirroring



---

## Chapter 32. Remote Monitoring

Remote Monitoring (RMON) allows network devices to exchange network monitoring data.

RMON performs the following major functions:

- Gathers cumulative statistics for Ethernet interfaces
- Tracks a history of statistics for Ethernet interfaces
- Creates and triggers alarms for user-defined events

---

## RMON Overview

The RMON MIB provides an interface between the RMON agent on the switch and an RMON management application. The RMON MIB is described in RFC 1757.

The RMON standard defines objects that are suitable for the management of Ethernet networks. The RMON agent continuously collects statistics and proactively monitors switch performance. RMON allows you to monitor traffic flowing through the switch.

The switch supports the following RMON Groups, as described in RFC 1757:

- Group 1: Statistics
- Group 2: History
- Group 3: Alarms
- Group 9: Events

---

## RMON Group 1—Statistics

The switch supports collection of Ethernet statistics as outlined in the RMON statistics MIB, in reference to `etherStatsTable`. RMON statistics are sampled every second, and new data overwrites any old data on a given port.

**Note:** RMON port statistics must be enabled for the port before you can view RMON statistics.

To configure RMON Statistics:

1. Enable RMON on each port where you wish to collect RMON statistics.

```
VFSM(config)# interface port 23
VFSM(config-if)# rmon
```

2. View RMON statistics for the port.

```
VFSM(config-if)# show interface port 23 rmon-counters
-----
RMON statistics for port 23:
etherStatsDropEvents:                NA
etherStatsOctets:                    7305626
etherStatsPkts:                      48686
etherStatsBroadcastPkts:             4380
etherStatsMulticastPkts:             6612
etherStatsCRCAlignErrors:            22
etherStatsUndersizePkts:             0
etherStatsOversizePkts:              0
etherStatsFragments:                 2
etherStatsJabbers:                   0
etherStatsCollisions:                0
etherStatsPkts64Octets:              27445
etherStatsPkts65to127Octets:         12253
etherStatsPkts128to255Octets:         1046
etherStatsPkts256to511Octets:         619
etherStatsPkts512to1023Octets:        7283
etherStatsPkts1024to1518Octets:       38
```

---

## RMON Group 2—History

The RMON History Group allows you to sample and archive Ethernet statistics for a specific interface during a specific time interval.

**Note:** RMON port statistics must be enabled for the port before an RMON history group can monitor the port.

Data is stored in buckets, which store data gathered during discreet sampling intervals. At each configured interval, the history instance takes a sample of the current Ethernet statistics, and places them into a bucket. History data buckets reside in dynamic memory. When the switch is re-booted, the buckets are emptied.

Requested buckets are the number of buckets, or data slots, requested by the user for each History Group. Granted buckets are the number of buckets granted by the system, based on the amount of system memory available. The system grants a maximum of 50 buckets.

Use an SNMP browser to view History samples.

## History MIB Objects

The type of data that can be sampled must be of an `ifIndex` object type, as described in RFC1213 and RFC1573. The most common data type for the history sample is as follows:

```
1.3.6.1.2.1.2.2.1.1.<x>
-mgmt.interfaces.ifTable.ifIndex.interface
```

The last digit (*x*) represents the interface on which to monitor, which corresponds to the switch port number. History sampling is done per port, by utilizing the interface number to specify the port number.

## Configuring RMON History

This example configuration creates an RMON History Group to monitor port 23. It takes a data sample every two minutes, and places the data into one of the 30 requested buckets. After 30 samples are gathered, the new samples overwrite the previous samples, beginning with the first bucket.

1. Enable RMON on each port where you wish to collect RMON History.

```
VFSM(config)# interface port 1
VFSM(config-if)# rmon
VFSM(config-if)# exit
```

2. Configure the RMON History parameters.

```
VFSM(config)# rmon history 1 interface-oid 1.3.6.1.2.1.2.2.1.1.<x>
VFSM(config)# rmon history 1 requested-buckets 30
VFSM(config)# rmon history 1 polling-interval 120
VFSM(config)# rmon history 1 owner "rmon port 1 history"
```

where *<x>* is the number of the port to monitor. For example, the full OID for port 1 would be: 1.3.6.1.2.1.2.2.1.1.1

### 3. View RMON history for the port.

```
VFSM(config)# show rmon history
RMON History group configuration:
```

Index	IFOID	Interval	Rbnum	Gbnum
1	1.3.6.1.2.1.2.2.1.1.1	120	30	30

Index	Owner
1	rmon port 1 history

---

## RMON Group 3—Alarms

The RMON Alarm Group allows you to define a set of thresholds used to determine network performance. When a configured threshold is crossed, an alarm is generated. For example, you can configure the switch to issue an alarm if more than 1,000 CRC errors occur during a 10-minute time interval.

Each Alarm index consists of a variable to monitor, a sampling time interval, and parameters for rising and falling thresholds. The Alarm group can be used to track rising or falling values for a MIB object. The object must be a counter, gauge, integer, or time interval.

Use one of the following commands to correlate an Alarm index to an Event index:

```
VFSM(config)# rmon alarm <alarm number> rising-crossing-index <event number>
VFSM(config)# rmon alarm <alarm number> falling-crossing-index <event number>
```

### Alarm MIB Objects

The most common data types used for alarm monitoring are `ifStats`: errors, drops, bad CRCs, and so on. These MIB Object Identifiers (OIDs) correlate to the ones tracked by the History group. An example of an ICMP stat is as follows:

```
1.3.6.1.2.1.5.1.<x> - mgmt.icmp.icmpInMsgs
```

where *x* represents the interface on which to monitor, which corresponds to the switch interface number or port number, as follows:

- 1 through 128 = Switch interface number
- 129 = Switch port 1
- 130 = Switch port 2
- 131 = Switch port 3, and so on.

This value represents the alarm's MIB OID, as a string. Note that for non-tables, you must supply a `.0` to specify an end node.

## Configuring RMON Alarms

### Alarm Example 1

This example configuration creates an RMON alarm that checks `ifInOctets` on port 20 once every hour. If the statistic exceeds two billion, an alarm is generated that triggers event index 6.

Configure the RMON Alarm parameters to track the number of packets received on a port.

```
VFSM(config)# rmon alarm 1 oid 1.3.6.1.2.1.2.2.1.10.129
VFSM(config)# rmon alarm 1 alarm-type rising
VFSM(config)# rmon alarm 1 rising-crossing-index 100
VFSM(config)# rmon alarm 1 interval 3600
VFSM(config)# rmon alarm 1 rising-limit 2000000000
VFSM(config)# rmon alarm 1 owner "Alarm for ifInOctets"
```

## Alarm Example 2

This example configuration creates an RMON alarm that checks `icmpInEchos` on the switch once every minute. If the statistic exceeds 200 within a 60 second interval, an alarm is generated that triggers event index 5. Configure the RMON Alarm parameters to track ICMP messages.

```
VFSM(config)# rmon alarm 1 oid 1.3.6.1.2.1.5.8.0
VFSM(config)# rmon alarm 1 alarm-type rising
VFSM(config)# rmon alarm 1 rising-crossing-index 110
VFSM(config)# rmon alarm 1 interval-time 60
VFSM(config)# rmon alarm 1 rising-limit 200
VFSM(config)# rmon alarm 1 sample delta
VFSM(config)# rmon alarm 1 owner "Alarm for icmpInEchos"
```

---

## RMON Group 9—Events

The RMON Event Group allows you to define events that are triggered by alarms. An event can be a log message, an SNMP trap message, or both.

When an alarm is generated, it triggers a corresponding event notification. Use the following commands to correlate an Event index to an alarm:

```
VFSM(config)# rmon alarm <alarm number> rising-crossing-index <event number>
VFSM(config)# rmon alarm <alarm number> falling-crossing-index <event number>
```

RMON events use SNMP and system logs to send notifications. Therefore, an SNMP trap host must be configured for trap event notification to work properly.

RMON uses a syslog host to send syslog messages. Therefore, an existing syslog host must be configured for event log notification to work properly. Each log event generates a system log message of type `RMON` that corresponds to the event.

For example, to configure the RMON event parameters.

```
VFSM(config)# rmon event 110 type log
VFSM(config)# rmon event 110 description "SYSLOG_this_alarm"
VFSM(config)# rmon event 110 owner "log icmpInEchos alarm"
```

This configuration creates an RMON event that sends a syslog message each time it is triggered by an alarm.

---

## Chapter 33. sFLOW

The VFSM supports sFlow technology for monitoring traffic in data networks. The switch includes an embedded sFlow agent which can be configured to sample network traffic and provide continuous monitoring information of IPv4 traffic to a central sFlow analyzer.

The switch is responsible only for forwarding sFlow information. A separate sFlow analyzer is required elsewhere on the network in order to interpret sFlow data.

**Note:** IBM Networking OS 7.8 does not support IPv6 for sFLOW.

---

## sFlow Statistical Counters

The VFSM can be configured to send network statistics to an sFlow analyzer at regular intervals. For each port, a polling interval of 5 to 60 seconds can be configured, or 0 (the default) to disable this feature.

When polling is enabled, at the end of each configured polling interval, the VFSM reports general port statistics.

---

## sFlow Network Sampling

In addition to statistical counters, the VFSM can be configured to collect periodic samples of the traffic data received on each port. For each sample, 128 bytes are copied, UDP-encapsulated, and sent to the configured sFlow analyzer.

For each port, the sFlow sampling rate can be configured to occur once each 256 to 65536 packets, or 0 to disable (the default). A sampling rate of 256 means that one sample will be taken for approximately every 256 packets received on the port. The sampling rate is statistical, however. It is possible to have slightly more or fewer samples sent to the analyzer for any specific group of packets (especially under low traffic conditions). The actual sample rate becomes most accurate over time, and under higher traffic flow.

sFlow sampling has the following restrictions:

- **Sample Rate**—The fastest sFlow sample rate is 1 out of every 256 packets.
- **ACLs**—sFlow sampling is performed before ACLs are processed. For ports configured both with sFlow sampling and one or more ACLs, sampling will occur regardless of the action of the ACL.
- **Port Mirroring**—sFlow sampling will not occur on mirrored traffic. If sFlow sampling is enabled on a port that is configured as a port monitor, the mirrored traffic will not be sampled.

**Note:** Although sFlow sampling is not generally a CPU-intensive operation, configuring fast sampling rates (such as once every 256 packets) on ports under heavy traffic loads can cause switch CPU utilization to reach maximum. Use larger rate values for ports that experience heavy traffic.

---

## sFlow Example Configuration

1. Specify the location of the sFlow analyzer (the server and optional port to which the sFlow information will be sent):

VFSM(config)# sflow server <IPv4 address>	(sFlow server address)
VFSM(config)# sflow port <service port>	(Set the optional service port)
VFSM(config)# sflow enable	(Enable sFlow features)

By default, the switch uses established sFlow service port 6343.

To disable sFlow features across all ports, use the following command:

```
VFSM(config)# no sflow enable
```

2. On a per-port basis, define the statistics polling rate:

VFSM(config)# interface port <port>	
VFSM(config-if)# sflow polling <polling rate>	(Statistics polling rate)

Specify a polling rate between 5 and 60 seconds, or 0 to disable. By default, polling is 0 (disabled) for each port.

3. On a per-port basis, define the data sampling rate:

VFSM(config-if)# sflow sampling <sampling rate>	(Data sampling rate)
---	----------------------

Specify a sampling rate between 256 and 65536 packets, or 0 to disable. By default, the sampling rate is 0 (disabled) for each port.

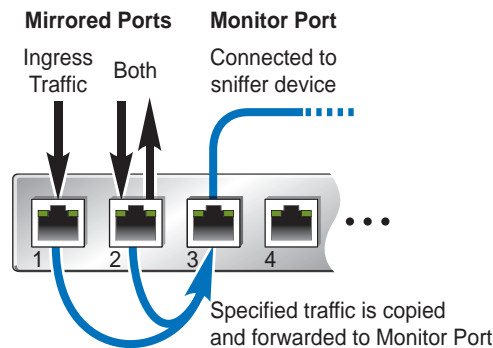
## Chapter 34. Port Mirroring

The IBM Networking OS port mirroring feature allows you to mirror (copy) the packets of a target port, and forward them to a monitoring port. Port mirroring functions for all layer 2 and layer 3 traffic on a port. This feature can be used as a troubleshooting tool or to enhance the security of your network. For example, an IDS server or other traffic sniffer device or analyzer can be connected to the monitoring port in order to detect intruders attacking the network.

**Note:** You can also use ACLs and VMaps with the port mirroring feature to monitor packets that match a filter. See [“ACL Port Mirroring” on page 112](#).

The VFSM supports a “many to one” mirroring model. As shown in [Figure 60](#), selected traffic for ports EXT1 and EXT2 is being monitored by port EXT3. In the example, both ingress traffic and egress traffic on port EXT2 are copied and forwarded to the monitor. However, port EXT1 mirroring is configured so that only ingress traffic is copied and forwarded to the monitor. A device attached to port EXT3 can analyze the resulting mirrored traffic.

Figure 60. Mirroring Ports



The VFSM supports one monitor port. The monitor port can receive mirrored traffic from any number of target ports.

IBM N/OS does not support “one to many” or “many to many” mirroring models where traffic from a specific port traffic is copied to multiple monitor ports. For example, port EXT1 traffic cannot be monitored by both port EXT3 and EXT4 at the same time, nor can port EXT2 ingress traffic be monitored by a different port than its egress traffic.

Ingress and egress traffic is duplicated and sent to the monitor port after processing.

**Note:** The Virtual Fabric Switch Module (VFSM) cannot mirror LACPDU packets. Also, traffic on management VLANs is not mirrored to the external ports.

---

## Port Mirroring Behavior

This section describes the composition of monitored packets in the VFSM, based on the configuration of the ports.

- Packets mirrored at port egress are mirrored prior to VLAN tag processing and may have a different PVID/Native VLAN than packets that egress the port toward their actual network destination.
- Packets mirrored at port ingress are not modified.

### Configuring Port Mirroring

The following procedure may be used to configure port mirroring for the example shown in [Figure 60 on page 519](#):

1. Specify the monitoring port, the mirroring port(s), and the port-mirror direction.

```
VFSM(config)# port-mirroring monitor-port EXT3 mirroring-port EXT1 in
VFSM(config)# port-mirroring monitor-port EXT3 mirroring-port EXT2 both
```

2. Enable port mirroring.

```
VFSM(config)# port-mirroring enable
```

3. View the current configuration.

```
VFSM# show port-mirroring                                     (Display the current settings)
Port mirroring is enabled
Monitoring Ports      Mirrored Ports
INTA1                 none
INTA2                 none
INTA3                 none
INTA4                 none
...
EXT1                  none
EXT2                  none
EXT3                  EXT1, in
                     EXT2, both
EXT4                  none
...
```

# Part 9: Appendices

This section describes the following topics:

- Glossary
- RADIUS Server Configuration Notes
- Getting help and technical assistance
- Notices



---

## Appendix A. Glossary

<b>CNA</b>	Converged Network Adapter. A device used for I/O consolidation such as that in Converged Enhanced Ethernet (CEE) environments implementing Fibre Channel over Ethernet (FCoE). The CNA performs the duties of both a Network Interface Card (NIC) for Local Area Networks (LANs) and a Host Bus Adapter (HBA) for Storage Area Networks (SANs).
<b>DIP</b>	The destination IP address of a frame.
<b>Dport</b>	The destination port (application socket: for example, http-80/https-443/DNS-53)
<b>HBA</b>	Host Bus Adapter. An adapter or card that interfaces with device drivers in the host operating system and the storage target in a Storage Area Network (SAN). It is equivalent to a Network Interface Controller (NIC) from a Local Area Network (LAN).
<b>NAT</b>	Network Address Translation. Any time an IP address is changed from one source IP or destination IP address to another address, network address translation can be said to have taken place. In general, half NAT is when the destination IP or source IP address is changed from one address to another. Full NAT is when both addresses are changed from one address to another. No NAT is when neither source nor destination IP addresses are translated.
<b>Preemption</b>	In VRRP, preemption will cause a Virtual Router that has a lower priority to go into backup should a peer Virtual Router start advertising with a higher priority.
<b>Priority</b>	In VRRP, the value given to a Virtual Router to determine its ranking with its peer(s). Minimum value is 1 and maximum value is 254. Default is 100. A higher number will win out for master designation.
<b>Proto (Protocol)</b>	The protocol of a frame. Can be any value represented by a 8-bit value in the IP header adherent to the IP specification (for example, TCP, UDP, OSPF, ICMP, and so on.)
<b>SIP</b>	The source IP address of a frame.
<b>SPort</b>	The source port (application socket: for example, HTTP-80/HTTPS-443/DNS-53).
<b>Tracking</b>	<p>In VRRP, a method to increase the priority of a virtual router and thus master designation (with preemption enabled). Tracking can be very valuable in an active/active configuration.</p> <p>You can track the following:</p> <ul style="list-style-type: none"><li>• Active IP interfaces on the Web switch (increments priority by 2 for each)</li><li>• Active ports on the same VLAN (increments priority by 2 for each)</li><li>• Number of virtual routers in master mode on the switch</li></ul>
<b>VIR</b>	Virtual Interface Router. A VRRP address is an IP interface address shared between two or more virtual routers.

<b>Virtual Router</b>	<p>A shared address between two devices utilizing VRRP, as defined in RFC 2338. One virtual router is associated with an IP interface. This is one of the IP interfaces that the switch is assigned. All IP interfaces on the VFSMs must be in a VLAN. If there is more than one VLAN defined on the Web switch, then the VRRP broadcasts will only be sent out on the VLAN of which the associated IP interface is a member.</p>
<b>VRID</b>	<p>Virtual Router Identifier. In VRRP, a numeric ID is used by each virtual router to create its MAC address and identify its peer for which it is sharing this VRRP address. The VRRP MAC address as defined in the RFC is 00-00-5E-00-01-&lt;VRID&gt;.</p> <p>If you have a VRRP address that two switches are sharing, then the VRID number needs to be identical on both switches so each virtual router on each switch knows with whom to share.</p>
<b>VRRP</b>	<p>Virtual Router Redundancy Protocol. A protocol that acts very similarly to Cisco's proprietary HSRP address sharing protocol. The reason for both of these protocols is so devices have a next hop or default gateway that is always available. Two or more devices sharing an IP interface are either advertising or listening for advertisements. These advertisements are sent via a broadcast message to an address such as 224.0.0.18.</p> <p>With VRRP, one switch is considered the master and the other the backup. The master is always advertising via the broadcasts. The backup switch is always listening for the broadcasts. Should the master stop advertising, the backup will take over ownership of the VRRP IP and MAC addresses as defined by the specification. The switch announces this change in ownership to the devices around it by way of a Gratuitous ARP, and advertisements. If the backup switch didn't do the Gratuitous ARP the Layer 2 devices attached to the switch would not know that the MAC address had moved in the network. For a more detailed description, refer to RFC 2338.</p>

## Appendix B. RADIUS Server Configuration Notes

Use the following information to modify your RADIUS configuration files for the Nortel Networks BaySecure Access Control RADIUS server, to provide authentication for users of the Virtual Fabric Switch Module (VFSM).

1. Create a dictionary file called `blade.dct`, with the following content:

```
#####
# blade.dct - RADLINX BLADE dictionary
#
# (See README.DCT for more details on the format of this file)
#####
#
# Use the Radius specification attributes in lieu of the
# RADLINX BLADE ones
#
@radius.dct

#
# Define additional RADLINX BLADE parameters
# (add RADLINX BLADE specific attributes below)

ATTRIBUTE   Radlinx-Vendor-Specific  26  [vid=648 data=string]  R

#####
# blade.dct - RADLINX BLADE dictionary
#####

#Define Virtual Fabric Switch Module dictionary
#@radius.dct

@blade.dct
        VALUE           Service-Type           user255
        VALUE           Service-Type           oper252
#####
```

2. Open the `dictiona.dcm` file, and add the following line (as in the example):  
`@blade.dct`

```
#####
# dictiona.dcm
#####
# Generic Radius

@radius.dct

#
# Specific Implementations (vendor specific)
#
@pprtl2l3.dct
@acc.dct
@accessbd.dct
@blade.dct
.
.
.
#####
# dictiona.dcm
#####
```

3. Open the vendor file (`vendor.ini`), and add the following data to the Vendor-Product identification list:

<code>vendor-product</code>	<code>= BLADE Blade-server module</code>
<code>dictionary</code>	<code>= blade</code>
<code>ignore-ports</code>	<code>= no</code>
<code>help-id</code>	<code>= 0</code>

---

## Appendix C. Getting help and technical assistance

If you need help, service, or technical assistance or just want more information about IBM products, you will find a wide variety of sources available from IBM to assist you. This section contains information about where to go for additional information about IBM and IBM products, what to do if you experience a problem with your system, and whom to call for service, if it is necessary.

---

### Before you call

Before you call, make sure that you have taken these steps to try to solve the problem yourself:

- Check all cables to make sure that they are connected.
- Check the power switches to make sure that the system and any optional devices are turned on.
- Use the troubleshooting information in your system documentation, and use the diagnostic tools that come with your system. Information about diagnostic tools is in the *Problem Determination and Service Guide* on the IBM Documentation CD that comes with your system.
- Go to the IBM support website at <http://www.ibm.com/systems/support/> to check for technical information, hints, tips, and new device drivers or to submit a request for information.

You can solve many problems without outside assistance by following the troubleshooting procedures that IBM provides in the online help or in the documentation that is provided with your IBM product. The documentation that comes with IBM systems also describes the diagnostic tests that you can perform. Most systems, operating systems, and programs come with documentation that contains troubleshooting procedures and explanations of error messages and error codes. If you suspect a software problem, see the documentation for the operating system or program.

---

### Using the documentation

Information about your IBM system and pre-installed software, if any, or optional device is available in the documentation that comes with the product. That documentation can include printed documents, online documents, ReadMe files, and Help files. See the troubleshooting information in your system documentation for instructions for using the diagnostic programs. The troubleshooting information or the diagnostic programs might tell you that you need additional or updated device drivers or other software. IBM maintains pages on the World Wide Web where you can get the latest technical information and download device drivers and updates. To access these pages, go to <http://www.ibm.com/systems/support/> and follow the instructions. Also, some documents are available through the IBM Publications Center at <http://www.ibm.com/shop/publications/order/>.

---

### Getting help and information on the World Wide Web

On the World Wide Web, the IBM website has up-to-date information about IBM systems, optional devices, services, and support. The address for IBM System x<sup>®</sup> and xSeries<sup>®</sup> information is <http://www.ibm.com/systems/x/>. The address for IBM BladeCenter information is <http://www.ibm.com/systems/bladecenter/>. The address for IBM IntelliStation<sup>®</sup> information is <http://www.ibm.com/intellistation/>.

You can find service information for IBM systems and optional devices at <http://www.ibm.com/systems/support/>.

---

## Software service and support

Through IBM Support Line, you can get telephone assistance, for a fee, with usage, configuration, and software problems with System x and x Series servers, BladeCenter products, IntelliStation workstations, and appliances. For information about which products are supported by Support Line in your country or region, see <http://www.ibm.com/services/sl/products/>.

For more information about Support Line and other IBM services, see <http://www.ibm.com/services/>, or see <http://www.ibm.com/planetwide/> for support telephone numbers. In the U.S. and Canada, call 1-800-IBM-SERV (1-800-426-7378).

---

## Hardware service and support

You can receive hardware service through your IBM reseller or IBM Services. To locate a reseller authorized by IBM to provide warranty service, go to <http://www.ibm.com/partnerworld/> and click **Find Business Partners** on the right side of the page. For IBM support telephone numbers, see <http://www.ibm.com/planetwide/>. In the U.S. and Canada, call 1-800-IBM-SERV (1-800-426-7378).

In the U.S. and Canada, hardware service and support is available 24 hours a day, 7 days a week. In the U.K., these services are available Monday through Friday, from 9 a.m. to 6 p.m.

---

## IBM Taiwan product service

台灣 IBM 產品服務聯絡方式：  
台灣國際商業機器股份有限公司  
台北市松仁路 7 號 3 樓  
電話：0800-016-888

IBM Taiwan product service contact information:

IBM Taiwan Corporation  
3F, No 7, Song Ren Rd.  
Taipei, Taiwan  
Telephone: 0800-016-888

---

## Appendix D. Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.*

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product, and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

---

## Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at <http://www.ibm.com/legal/copytrade.shtml>.

Adobe and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc., in the United States, other countries, or both and is used under license therefrom.

Intel, Intel Xeon, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc., in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

---

## Important Notes

Processor speed indicates the internal clock speed of the microprocessor; other factors also affect application performance.

CD or DVD drive speed is the variable read rate. Actual speeds vary and are often less than the possible maximum.

When referring to processor storage, real and virtual storage, or channel volume, KB stands for 1024 bytes, MB stands for 1 048 576 bytes, and GB stands for 1 073 741 824 bytes.

When referring to hard disk drive capacity or communications volume, MB stands for 1 000 000 bytes, and GB stands for 1 000 000 000 bytes. Total user-accessible capacity can vary depending on operating environments.

Maximum internal hard disk drive capacities assume the replacement of any standard hard disk drives and population of all hard disk drive bays with the largest currently supported drives that are available from IBM.

Maximum memory might require replacement of the standard memory with an optional memory module.

IBM makes no representation or warranties regarding non-IBM products and services that are ServerProven, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. These products are offered and warranted solely by third parties.

IBM makes no representations or warranties with respect to non-IBM products. Support (if any) for the non-IBM products is provided by the third party, not IBM.

Some software might differ from its retail version (if available) and might not include user manuals or all program functionality.

---

## Particulate contamination

**Attention:** Airborne particulates (including metal flakes or particles) and reactive gases acting alone or in combination with other environmental factors such as humidity or temperature might pose a risk to the device that is described in this document. Risks that are posed by the presence of excessive particulate levels or concentrations of harmful gases include damage that might cause the device to malfunction or cease functioning altogether. This specification sets forth limits for particulates and gases that are intended to avoid such damage. The limits must not be viewed or used as definitive limits, because numerous other factors, such as temperature or moisture content of the air, can influence the impact of particulates or environmental corrosives and gaseous contaminant transfer. In the absence of specific limits that are set forth in this document, you must implement practices that maintain particulate and gas levels that are consistent with the protection of human health and safety. If IBM determines that the levels of particulates or gases in your environment have caused damage to the device, IBM may condition provision of repair or replacement of devices or parts on implementation of appropriate remedial measures to mitigate such environmental contamination. Implementation of such remedial measures is a customer responsibility.

Contaminant	Limits
Particulate	<ul style="list-style-type: none"><li>• The room air must be continuously filtered with 40% atmospheric dust spot efficiency (MERV 9) according to ASHRAE Standard 52.2<sup>1</sup>.</li><li>• Air that enters a data center must be filtered to 99.97% efficiency or greater, using high-efficiency particulate air (HEPA) filters that meet MIL-STD-282.</li><li>• The deliquescent relative humidity of the particulate contamination must be more than 60%<sup>2</sup>.</li><li>• The room must be free of conductive contamination such as zinc whiskers.</li></ul>
Gaseous	<ul style="list-style-type: none"><li>• Copper: Class G1 as per ANSI/ISA 71.04-1985<sup>3</sup></li><li>• Silver: Corrosion rate of less than 300 Å in 30 days</li></ul>

<sup>1</sup> ASHRAE 52.2-2008 - *Method of Testing General Ventilation Air-Cleaning Devices for Removal Efficiency by Particle Size*. Atlanta: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.

<sup>2</sup> The deliquescent relative humidity of particulate contamination is the relative humidity at which the dust absorbs enough water to become wet and promote ionic conduction.

<sup>3</sup> ANSI/ISA-71.04-1985. *Environmental conditions for process measurement and control systems: Airborne contaminants*. Instrument Society of America, Research Triangle Park, North Carolina, U.S.A.

---

## Documentation format

The publications for this product are in Adobe Portable Document Format (PDF) and should be compliant with accessibility standards. If you experience difficulties when you use the PDF files and want to request a web-based format or accessible PDF document for a publication, direct your mail to the following address:

Information Development  
IBM Corporation  
205/A0153039 E. Cornwallis Road  
P.O. Box 12195  
Research Triangle Park, North Carolina 27709-2195  
U.S.A.

In the request, be sure to include the publication part number and title.

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

---

## Electronic emission notices

### Federal Communications Commission (FCC) statement

**Note:** This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user will be required to correct the interference at his own expense.

Properly shielded and grounded cables and connectors must be used in order to meet FCC emission limits. IBM is not responsible for any radio or television interference caused by using other than recommended cables and connectors or by unauthorized changes or modifications to this equipment. Unauthorized changes or modifications could void the user's authority to operate the equipment.

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

### Industry Canada Class A emission compliance statement

This Class A digital apparatus complies with Canadian ICES-003.

### Avis de conformité à la réglementation d'Industrie Canada

Cet appareil numérique de la classe A est conforme à la norme NMB-003 du Canada.

### Australia and New Zealand Class A statement

**Attention:** This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

### European Union EMC Directive conformance statement

This product is in conformity with the protection requirements of EU Council Directive 2004/108/EC on the approximation of the laws of the Member States relating to electromagnetic compatibility. IBM cannot accept responsibility for any failure to satisfy the protection requirements resulting from a nonrecommended modification of the product, including the fitting of non-IBM option cards.

**Attention:** This is an EN 55022 Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

Responsible manufacturer:

International Business Machines Corp.  
New Orchard Road  
Armonk, New York 10504  
914-499-1900

European Community contact:

IBM Technical Regulations, Department M456  
IBM-Allee 1, 71137 Ehningen, Germany  
Telephone: +49 7032 15-2937  
E-mail: tjahn@de.ibm.com

## **Germany Class A statement**

### **Deutschsprachiger EU Hinweis:**

#### **Hinweis für Geräte der Klasse A EU-Richtlinie zur Elektromagnetischen Verträglichkeit**

Dieses Produkt entspricht den Schutzanforderungen der EU-Richtlinie 2004/108/EG zur Angleichung der Rechtsvorschriften über die elektromagnetische Verträglichkeit in den EU-Mitgliedsstaaten und hält die Grenzwerte der EN 55022 Klasse A ein.

Um dieses sicherzustellen, sind die Geräte wie in den Handbüchern beschrieben zu installieren und zu betreiben. Des Weiteren dürfen auch nur von der IBM empfohlene Kabel angeschlossen werden. IBM übernimmt keine Verantwortung für die Einhaltung der Schutzanforderungen, wenn das Produkt ohne Zustimmung der IBM verändert bzw. wenn Erweiterungskomponenten von Fremdherstellern ohne Empfehlung der IBM gesteckt/eingebaut werden.

EN 55022 Klasse A Geräte müssen mit folgendem Warnhinweis versehen werden: "Warnung: Dieses ist eine Einrichtung der Klasse A. Diese Einrichtung kann im Wohnbereich Funk-Störungen verursachen; in diesem Fall kann vom Betreiber verlangt werden, angemessene Maßnahmen zu ergreifen und dafür aufzukommen."

#### **Deutschland: Einhaltung des Gesetzes über die elektromagnetische Verträglichkeit von Geräten**

Dieses Produkt entspricht dem "Gesetz über die elektromagnetische Verträglichkeit von Geräten (EMVG)". Dies ist die Umsetzung der EU-Richtlinie 2004/108/EG in der Bundesrepublik Deutschland.

#### **Zulassungsbescheinigung laut dem Deutschen Gesetz über die elektromagnetische Verträglichkeit von Geräten (EMVG) (bzw. der EMC EG Richtlinie 2004/108/EG) für Geräte der Klasse A**

Dieses Gerät ist berechtigt, in Übereinstimmung mit dem Deutschen EMVG das EG-Konformitätszeichen - CE - zu führen.

Verantwortlich für die Einhaltung der EMV Vorschriften ist der Hersteller:

International Business Machines Corp.  
New Orchard Road  
Armonk, New York 10504  
914-499-1900

Der verantwortliche Ansprechpartner des Herstellers in der EU ist:

IBM Deutschland  
Technical Regulations, Department M456  
IBM-Allee 1, 71137 Ehningen, Germany  
Telephone: +49 7032 15-2937  
E-mail: tjahn@de.ibm.com

**Generelle Informationen:**

**Das Gerät erfüllt die Schutzanforderungen nach EN 55024 und EN 55022 Klasse A.**

## Japan VCCI Class A statement

この装置は、クラス A 情報技術装置です。この装置を家庭環境で使用する  
と電波妨害を引き起こすことがあります。この場合には使用者が適切な対策  
を講ずるよう要求されることがあります。 VCCI-A

This is a Class A product based on the standard of the Voluntary Control Council for Interference (VCCI). If this equipment is used in a domestic environment, radio interference may occur, in which case the user may be required to take corrective actions.

## Korea Communications Commission (KCC) statement

이 기기는 업무용으로 전자파 적합등록을 받은 기기  
이오니, 판매자 또는 사용자는 이점을 주의하시기  
바라며, 만약 잘못 구입하셨을 때에는 구입한 곳에  
서 비업무용으로 교환하시기 바랍니다.

Please note that this equipment has obtained EMC registration for commercial use. In the event that it has been mistakenly sold or purchased, please exchange it for equipment certified for home use.

## Russia Electromagnetic Interference (EMI) Class A statement

ВНИМАНИЕ! Настоящее изделие относится к классу А.  
В жилых помещениях оно может создавать радиопомехи, для  
снижения которых необходимы дополнительные меры

## People's Republic of China Class A electronic emission statement

中华人民共和国“A类”警告声明

声 明

此为A级产品，在生活环境中，该产品可能会造成无线电干扰。在这种情况下，  
可能需要用户对其干扰采取切实可行的措施。

## Taiwan Class A compliance statement

警告使用者：  
這是甲類的資訊產品，在  
居住的環境中使用時，可  
能會造成射頻干擾，在這  
種情況下，使用者會被要  
求採取某些適當的對策。



---

# Index

## Symbols

[ ] 22

## Numerics

802.1p QoS 269  
802.1Q VLAN tagging 127, 281  
802.1Qaz ETS 281  
802.1Qbb PFC 277  
802.1Qbg. *See* EVB  
802.3x flow control 269, 277

## A

Access Control Lists. *See* ACLs.  
accessible documentation 531  
accessing the switch  
    Browser-based Interface 29, 34  
    LDAP authentication 90  
    RADIUS authentication 82  
    security 81  
ACLs 103, 182  
    FCoE 273  
    FIP snooping 263, 271  
active-active redundancy 462  
administrator account 43, 51, 84  
advertise flag (DCBX) 288  
aggregating routes 404  
    example 410  
AH 342  
anycast address, IPv6 332  
application ports 105  
assistance, getting 521, 527  
authenticating, in OSPF 423  
Authentication Header (AH) 342  
autoconfiguration  
    link 57  
autoconfiguration, IPv6 333  
auto-negotiation  
    setup 57  
autonomous systems (AS) 417

## B

bandwidth allocation 269, 283  
BBI 28  
BBI. *See* Browser-Based Interface  
Border Gateway Protocol (BGP) 397  
    attributes 406  
    failover configuration 408  
    route aggregation 404  
    route maps 401  
    selecting route paths 407  
bridge module 233, 262, 264  
Bridge Protocol Data Unit (BPDU) 161

broadcast domains 123, 322  
Browser-Based Interface 28, 418

## C

CEE 261, 268  
    802.1p QoS 269  
    bandwidth allocation 269  
    DCBX 261, 268, 287  
    ETS 261, 269, 281  
    FCoE 264, 268  
    LLDP 268  
    on/off 268  
    PFC 261, 270, 277  
    priority groups 282  
Cisco EtherChannel 147  
CIST 174  
Class A electronic emission notice 532  
Class of Service queueCOS queue 190  
CNA 263, 264  
command conventions 22  
Command Line Interface 418  
Command-Line Interface (CLI) 51  
Community VLAN 139  
configuration rules  
    CEE 268  
    FCoE 264  
    port mirroring 147  
    spanning tree 147  
    Trunking 147  
    VLANs 147  
configuring  
    BGP failover 408  
    DCBX 289  
    ETS 284  
    FIP snooping 275  
    IP routing 321  
    OSPF 427  
    PFC 279  
    port trunking 148  
    spanning tree groups 171, 176  
contamination, particulate and gaseous 531  
Converged Enhanced Ethernet. *See* CEE.  
Converged Network Adapter. *See* CNA.

## D

Data Center Bridging Capability Exchange. *See* DCBX.  
date  
    setup 55  
DCBX 261, 268, 287  
default gateway 320  
    configuration example 322  
default password 43, 84  
default route, OSPF 422  
Differentiated Services Code Point (DSCP)DSCP 184

- digital certificate 344
  - generating 345
  - importing 345
- documentation format 531

## E

- EAPoL 94
- ECP 291
- Edge Control Protocol. *See ECP*
- Edge Virtual Bridging. *See EVB*
- electronic emission Class A notice 532
- Encapsulating Security Payload (ESP) 342
- End user access control, configuring 76
- Enhanced Transmission Selection. *See ETS*.
- ENodes 263, 271
- ESP 342
- EtherChannel 144
  - as used with port trunking 147
- Ethernet Nodes (FCoE). *See ENodes*.
- ETS 261, 269, 281
  - bandwidth allocation 269, 283
  - configuring 284
  - DCBX 288
  - PGID 269, 282
  - priority groups 282
  - priority values 283
- EVB 291
- Extensible Authentication Protocol over LAN 94
- external routing 398, 417

## F

- factory default configuration 45, 51, 53
- failover 449
  - overview 461
- FC-BB-5 262
- FCC Class A notice 532
- FCF 233, 262, 264, 271
  - detection mode 272
- FCoE 261, 262
  - bridge module 233, 262, 264
  - CEE 264, 268
  - CNA 263, 264
  - ENodes 263
  - FCF 233, 262, 264
  - FIP snooping 261, 263, 271
  - FLOGI 273
  - point-to-point links 262
  - requirements 264
  - SAN 262, 268
  - topology 262
  - VLANs 273
- FCoE Forwarder. *See FCF*.
- FCoE Initialization Protocol snooping. *See FIP snooping*.
- Fibre Channel over Ethernet. *See FCoE*.
- Final Steps 64

- FIP snooping 261, 263, 271
  - ACL rules 273
  - ENode mode 272
  - FCF mode 272
  - timeout 272
- first-time configuration 45, 51 to ??
- FLOGI 273
- flow control 269, 277
  - setup 57
- frame size 124
- frame tagging. *See VLANs tagging*.

## G

- gaseous contamination 531
- gateway. *See default gateway*.
- getting help 527

## H

- hardware service and support 528
- help, getting 527
- high-availability 457
- host routes, OSPF 425
- Hot Links 445
- hot-standby redundancy 462
- HP-OpenView 39, 487
- hypervisor 217

## I

- IBM DirectorSNMP
  - IBM Director 39, 487
- IBM support line 528
- ICMP 104
- IEEE standards
  - 802.1D 158
  - 802.1Qaz 281
  - 802.1Qbb 277
  - 802.1Qbg 291
  - 802.1s 174
  - 802.1x 94
  - 802.3x 277
- IGMP 104, 363
  - Querier 393
- IGMP Relay 378
- IGMPv3 367
- IKEv2 342
  - digital certificate 344, 345
  - presared key 344, 346
- IKEv2 proposal 344
- INCITS T11.3 262
- incoming route maps 401
- internal routing 398, 417
- Internet Group Management Protocol (IGMP) 363
- Internet Key Exchange Version 2 (IKEv2) 342
- Internet Protocol Security
  - See also IPsec* 341

- Inter-Switch Link, ISL 145
- IP address 60
  - IP interface 60
- IP address routing example 321
- IP configuration via setup 60
- IP interfaces 60
- IP interfaces, example configuration 321, 323
- IP routing 60
  - cross-subnet example 319
  - default gateway configuration 322
  - IP interface configuration 321, 323
  - subnet configuration example 321
  - switch-based topology 320
- IP subnet mask 60
- IP subnets 320
  - routing 320
  - VLANs 123
- IPSec
  - maximum traffic load 343
- IPsec 341
  - key policy 346
- IPv6 addressing 329, 331
- ISL Trunking 144
- Isolated VLAN 139

## J

- jumbo frames 124

## L

- LACP 150
- Layer 2 Failover 449
- LDAP authentication 90
- Link Aggregation Control Protocol 150
- LLDP 268, 287
- logical segment. *See* IP subnets.
- lossless Ethernet 262, 268
- LSAs 416

## M

- management module 28, 30
- manual style conventions 22
- Maximum Transmission Unit 124
- meter 111, 183
- mirroring ports 519
- monitoring ports 519
- MSTP 174
- MTU 124
- multi-links between switches using port trunking 143
- multiple spanning tree groups 166
- Multiple Spanning Tree Protocol 174

## N

- Neighbor Discovery, IPv6 335
- Network Load Balancing, *See* *NLB*,

- network management 28, 39, 487
- NLB 299
- notes, important 530
- notices 529
- notices, electronic emission 532
- notices, FCC Class A 532

## O

- OSPF
  - area types 414
  - authentication 423
  - configuration examples 428 to ??
  - default route 422
  - external routes 426
  - filtering criteria 104
  - host routes 425
  - link state database 416
  - neighbors 416
  - overview 414
  - redistributing routes 405
  - route maps 401, 402
  - route summarization 421
  - router ID 423
  - virtual link 422
- outgoing route maps 401

## P

- packet size 124
- particulate contamination 531
- password
  - administrator account 43, 84
  - default 43, 84
  - user account 43, 84
- passwords 43
- payload size 124
- Per Hop Behavior (PHB)PHB 185
- PFC 261, 270, 277
  - DCBX 288
- PGID 269, 282
- port flow control. *See* flow control.
- port mirroring 519
  - configuration rules 147
- port trunking
  - configuration example 148
  - EtherChannel 144
- ports
  - configuration 57
  - for services 105
  - monitoring 519
  - physical. *See* switch ports.
- preshared key 344
  - enabling 346
- priority groups 282
- priority value (802.1p) 189, 270, 281
- Priority-based Flow Control. *See* PFC.
- Private VLANs 139

- promiscuous port 140
- protocol types 104
- PVID (port VLAN ID) 126
- PVLAN 135

## Q

- Querier (IGMP) 393

## R

### RADIUS

- authentication 82
- port 1812 and 1645 105
- port 1813 105
- SSH/SCP 75

- Rapid Spanning Tree Protocol (RSTP) 173

- receive flow control 57

- redistributing routes 405, 410

### redundancy

- active-active 462
- hot-standby 462

- re-mark 111, 183

- restarting switch setup 54

### RIP (Routing Information Protocol)

- advertisements 354
- distance vector protocol 352
- hop count 352
- TCP/IP route information 20, 351
- version 1 352

- route aggregation 404, 410

- route maps 401

- configuring 402
- incoming and outgoing 401

- route paths in BGP 407

- Router ID, OSPF 423

- routers 319, 322

- border 417
- peer 417
- port trunking 144
- switch-based routing topology 320

- routes, advertising 417

- routing 398

- internal and external 417

- Routing Information Protocol. *See* RIP

- RSA keys 75

- RSTP 173

- rx flow control 57

## S

- SA 342

- SAN 262, 268

### security

- LDAP authentication 90
- port mirroring 519
- RADIUS authentication 82
- VLANs 123

- security association (SA) 342

- segmentation. *See* IP subnets.

- segments. *See* IP subnets.

- service and support 528

- service ports 105

- setup facility 45, 51

- IP configuration 60

- IP subnet mask 60

- port auto-negotiation mode 57

- port configuration 57

- port flow control 57

- restarting 54

- Spanning-Tree Protocol 56

- starting 53

- stopping 54

- system date 55

- system time 55

- VLAN name 59

- VLAN tagging 58

- VLANs 59

- SNMP 28, 39, 418, 487

- HP-OpenView 39, 487

- SNMP Agent 487

- software service and support 528

- Source-Specific MulticastSSM 368

- Spanning Tree Protocol

- configuration rules 147

- Spanning-Tree Protocol

- multiple instances 166

- setup (on/off) 56

- SSH/SCP

- configuring 71

- RSA host and server keys 75

- stacking 196, 296, 447

- starting switch setup 53

- Static ARP 299

- stopping switch setup 54

- Storage Area Network. *See* SAN.

- subnet mask 60

- subnets 60

- summarizing routes 421

- support line 528

- support web site 528

- switch failover 461

- switch ports VLANs membership 126

## T

- TACACS+ 86

- tagging. *See* VLANs tagging.

- TCP 104

- technical assistance 527

- technical terms
  - port VLAN identifier (PVID) 127
  - tagged frame 127
  - tagged member 127
  - untagged frame 127
  - untagged member 127
  - VLAN identifier (VID) 127
- telephone assistance 528
- telephone numbers 528
- Telnet support
  - optional setup for Telnet support 65
- text conventions 22
- time
  - setup 55
- trademarks 529
- transmit flow control 57
- Trunking configuration rules 147
- tx flow control 57
- typographic conventions 22

## U

- UDP 104
- user account 43, 84

## V

- VDP 291
- vDS. *See virtual Distributed Switch*
- VEB 291
- VEPA 291
- virtual Distributed Switch 247
- Virtual Ethernet Bridging. *See VEB*
- Virtual Ethernet Port Aggregator. *See VEPA*
- virtual interface router (VIR) 458
- virtual link, OSPF 422
- Virtual Local Area Networks. *See VLANs.*
- virtual NICs 217
- virtual router group 462
- virtual router ID numbering 465
- Virtual Station Interface. *See VSI*
- VLAN tagging
  - setup 58

- VLANs 60
  - broadcast domains 123, 322
  - configuration rules 147
  - default PVID 126
  - example showing multiple VLANs 133
  - FCoE 273
  - ID numbers 125
  - interface 61
  - IP interface configuration 323
  - multiple spanning trees 160
  - multiple VLANs 127
  - name setup 59
  - port members 126
  - PVID 126
  - routing 322
  - security 123
  - setup 59
  - Spanning-Tree Protocol 160
  - tagging 58, 126 to 134
  - topologies 132
- vNICs 217
- VRRP (Virtual Router Redundancy Protocol)
  - active-active redundancy 462
  - hot-standby redundancy 462
  - overview 458
  - virtual interface router 458
  - virtual router ID numbering 465
  - vrid 458
- VSI 291
- VSI Discovery and Configuration Protocol. *See VDP*

## W

- website, publication ordering 527
- website, support 528
- website, telephone support numbers 528
- willing flag (DCBX) 288