EnGenius

ECS8830F/ECS8854F

Configuration Guide

Version 1.1

Contents

Chapter 1 Basic Management Configuration....1
1.1 Switch Management....1
1.1.1 Management Options....1
1.1.2 CLI Interface....8
1.2 Basic Switch Configuration....12
1.2.1 Basic Configuration....12
1.2.2 Telnet Management....13
1.2.3 Configure Switch IP Addresses....16
1.2.4 SNMP Configuration....17
1.2.5 Switch Upgrade....22
1.3 File System....30
1.3.1 Introduction to File Storage Devices....30
1.3.2 File System Operation Configuration Task List....30
1.3.3 Typical Applications....31
1.3.4 Troubleshooting....31
1.4 USB....31
1.4.1 Introduction....31
1.4.2 USB Function Configuration Task List....32
1.4.3 USB Function Examples....33
1.4.4 USB Function Troubleshooting....33
1.5 Device Management....33
1.5.1 Device Management Overview....33
1.5.2 Device Management Configuration....34

Chapter 2 Network Security Operations Manual....35
2.1 Introduction to Network Security Features....35
2.2 Network Security Feature Configuration....35
2.2.1 Port Number Modification Configuration Task Sequence....35
2.2.2 Encryption Algorithm Modification Configuration Task Sequence....35
2.2.3 Password Security Function Configuration Task Sequence....35
2.2.4 Sensitive Information Encryption Function Configuration Task Sequence....36
2.2.5 Important File Verification Function Configuration Task Sequence....36

Chapter 3 Layer 2 Services Configuration....38
3.1 Port Configuration....38
3.1.1 Introduction to Port....38
3.1.2 Network Port Configuration Task List....38
3.1.3 Port Configuration Example....40
3.1.4 Port Troubleshooting....40
3.2 Port Isolation....41
3.2.1 Introduction to Port Isolation Function....41
3.2.2 Port Isolation Task Sequence....41
3.2.3 Port Isolation Typical Example....42
3.3 Port Loopback Detection....42
3.3.1 Introduction to Port Loopback Detection....42
3.3.2 Port Loopback Detection Configuration Task List....43
3.3.3 Port Loopback Detection Example....44
3.3.4 Port Loopback Detection Troubleshooting....44
3.4 ULDP....45
3.4.1 Introduction to ULDP....45
3.4.2 ULDP Configuration Task Sequence....46
3.4.3 ULDP Typical Example....48
3.4.4 ULDP Troubleshooting....49
3.5 LLDP....49
3.5.1 Introduction to LLDP....49
3.5.2 LLDP Configuration Task Sequence....50

3.5.3 LLDP Typical Example....52
3.5.4 LLDP Function Troubleshooting....53
3.6 LLDP-MED....53
3.6.1 Introduction to LLDP-MED....53
3.6.2 LLDP-MED Configuration Task Sequence....53
3.6.3 LLDP-MED Example....55
3.6.4 LLDP-MED Troubleshooting....56
3.7 Port Channel....57
3.7.1 Introduction to Port Channel....57
3.7.2 Brief Introduction to LACP....58
3.7.3 Introduction to Load Balance....59
3.7.4 Port Channel Configuration Task List....59
3.7.5 Port Channel Examples....60
3.7.6 Troubleshooting....62
3.8 MTU....62
3.8.1 Introduction to MTU....62
3.8.2 MTU Configuration Task Sequence....62
3.9 BPDU Tunnel....62
3.9.1 Introduction to BPDU Tunnel....62
3.9.2 BPDU Tunnel Configuration Task List....63
3.9.3 BPDU Tunnel Example....63
3.9.4 bpdu-tunnel Troubleshooting....64
3.10 DDM....65
3.10.1 Introduction to DDM....65
3.10.2 DDM Configuration Task List....66
3.10.3 Examples of DDM....67
3.10.4 DDM Troubleshooting....70
3.11 EFM OAM....70
3.11.1 Introduction to EFM OAM....70
3.11.2 EFM OAM Configuration....73
3.11.3 EFM OAM Example....75
3.11.4 EFM OAM Troubleshooting....75
3.12 Port Security....76
3.12.1 Introduction to Port Security....76
3.12.2 Port Security Configuration Task List....76
3.12.3 Port Security Example....77
3.12.4 Port Security Troubleshooting....77
3.13 VLAN....77
3.13.1 Introduction to VLAN....77
3.13.2 VLAN Configuration Task List....78
3.13.3 Typical VLAN Application....80
3.13.4 Typical Hybrid Port Application....82
3.14 GVRP....83
3.14.1 Introduction to GVRP....83
3.14.2 GVRP Configuration Task List....84
3.14.3 GVRP Example....84
3.14.4 GVRP Troubleshooting....86
3.15 Dot1q-tunnel....86
3.15.1 Introduction to Dot1q-tunnel....86
3.15.2 Dot1q-tunnel Configuration....87
3.15.3 Typical Dot1q-tunnel Application....87
3.15.4 Dot1q-tunnel Troubleshooting....88
3.16 VLAN-translation....88
3.16.1 Introduction to VLAN-translation....88
3.16.2 VLAN-translation Configuration....88
3.16.3 Typical VLAN-translation Application....89
3.16.4 VLAN-translation Troubleshooting....90
3.17 Dynamic VLAN....90
3.17.1 Introduction to Dynamic VLAN....90
3.17.2 Dynamic VLAN Configuration....90
3.17.3 Typical Application of the Dynamic VLAN....92

3.17.4 Dynamic VLAN Troubleshooting....92
3.18 Voice VLAN....93
3.18.1 Introduction to Voice VLAN....93
3.18.2 Voice VLAN Configuration....93
3.18.3 Typical Applications of the Voice VLAN....94
3.18.4 Voice VLAN Troubleshooting....95
3.19 Multi-to-One VLAN Translation....95
3.19.1 Introduction to Multi-to-One VLAN Translation....95
3.19.2 Multi-to-One VLAN Translation Configuration....95
3.19.3 Typical Application of Multi-to-One VLAN Translation....95
3.19.4 Multi-to-One VLAN Translation Troubleshooting....96
3.20 Super VLAN....97
3.20.1 Introduction to Super VLAN....97
3.20.2 Super VLAN Configuration....98
3.20.3 Typical Application of Super VLAN....99
3.20.4 Super VLAN Troubleshooting....100
3.21 MAC Address Table....100
3.21.1 Introduction to the MAC Table....100
3.21.2 MAC Address Table Configuration Task List....102
3.21.3 Typical Configuration Examples....104
3.21.4 MAC Table Troubleshooting....104
3.22 CFM-OAM Configuration....104
3.22.1 Overview....104
3.22.2 CFM OAM Basic Concepts....105
3.22.3 Introduction to CFM OAM Functions....107
3.22.4 CFM OAM Basic Function Configuration....110
3.22.5 CFM OAM Fault Verification....113
3.22.6 CFM OAM Fault Location....114
3.22.7 ULPP Linkage (Optional)....115
3.22.8 Configuration Application Example....117
3.22.9 CFM Troubleshooting....120

Chapter 4 IP Services Configuration....122

4.1 Layer 3 Interface....122
4.1.1 Introduction to Layer 3 Interfaces....122
4.1.2 Layer 3 Interface Configuration Task List....122
4.2 Network Management Port....123
4.2.1 Introduction to the IP Network Management Port....123
4.2.2 Network Management Port Configuration....123
4.3 IP Configuration....124
4.3.1 Introduction to IPv4 and IPv6....124
4.3.2 IP Configuration....125
4.3.3 IP Configuration Examples....130
4.3.4 IPv6 Troubleshooting....133
4.4 IP Forwarding....133
4.4.1 Introduction to IP Forwarding....133
4.4.2 IP Route Aggregation Configuration Task List....133
4.5 URPF....134
4.5.1 Introduction to URPF....134
4.5.2 URPF Configuration Task List....134
4.5.3 URPF Typical Example....135
4.5.4 URPF Troubleshooting....135
4.6 ARP....135
4.6.1 Introduction to ARP....135
4.6.2 ARP Configuration Task List....135
4.6.3 ARP Troubleshooting....136
4.7 Hardware Tunnel Capacity....136
4.7.1 Introduction to Hardware Tunnel Capacity....136
4.7.2 Hardware Tunnel Capacity Configuration....136
4.7.3 Hardware Tunnel Capacity Troubleshooting....137

4.8 ARP Scanning Prevention....137
4.8.1 Introduction to ARP Scanning Prevention Function....137
4.8.2 ARP Scanning Prevention Configuration Task List....137
4.8.3 ARP Scanning Prevention Typical Examples....139
4.8.4 ARP Scanning Prevention Troubleshooting....140
4.9 Prevent ARP Spoofing....140
4.9.1 Overview....140
4.9.2 Prevent ARP Spoofing Configuration....141
4.9.3 Prevent ARP Spoofing Example....141
4.10 ARP Guard....142
4.10.1 Introduction to ARP Guard....142
4.10.2 ARP Guard Configuration Task List....143
4.11 ARP Local Proxy....143
4.11.1 Introduction to ARP Local Proxy....143
4.11.2 ARP Local Proxy Configuration Task List....144
4.11.3 ARP Local Proxy Typical Example....144
4.11.4 ARP Local Proxy Troubleshooting....145
4.12 Gratuitous ARP....145
4.12.1 Introduction to Gratuitous ARP....145
4.12.2 Gratuitous ARP Configuration Task List....145
4.12.3 Gratuitous ARP Configuration Example....146
4.12.4 Gratuitous ARP Troubleshooting....146
4.13 Keepalive Gateway....146
4.13.1 Introduction to Keepalive Gateway....146
4.13.2 Keepalive Gateway Configuration Task List....147
4.13.3 Keepalive Gateway Example....147
4.13.4 Keepalive Gateway Troubleshooting....148
4.14 Dynamic ARP Inspection....148
4.14.1 Introduction to Dynamic ARP Inspection....148
4.14.2 Dynamic ARP Inspection Configuration Task List....148
4.14.3 Dynamic ARP Inspection Configuration Example....149
4.15 DHCP....150
4.15.1 Introduction to DHCP....150
4.15.2 DHCP Server Configuration....151
4.15.3 DHCP Relay Configuration....153
4.15.4 DHCP Configuration Examples....153
4.15.5 DHCP Troubleshooting....155
4.16 DHCP Option 82....156
4.16.1 Introduction to DHCP Option 82....156
4.16.2 DHCP Option 82 Configuration Task List....157
4.16.3 DHCP Option 82 Application Examples....159
4.16.4 DHCP Option 82 Troubleshooting....160
4.17 DHCP Snooping....161
4.17.1 Introduction to DHCP Snooping....161
4.17.2 DHCP Snooping Configuration Task Sequence....162
4.17.3 DHCP Snooping Typical Application....165
4.17.4 DHCP Snooping Troubleshooting Help....165
4.18 DHCP Option 60 and Option 43....166
4.18.1 Introduction to DHCP Option 60 and Option 43....166
4.18.2 DHCP Option 60 and Option 43 Configuration Task List....166
4.18.3 DHCP Option 60 and Option 43 Example....167
4.18.4 DHCP Option 60 and Option 43 Troubleshooting....167

Chapter 5 Routing Protocol Related Configuration....168

5.1 Routing Protocol Overview....168
5.1.1 Routing Table....168
5.1.2 IP Routing Policy....169
5.2 Static Route....174
5.2.1 Introduction to Static Route....174
5.2.2 Introduction to Default Route....174

5.2.3 Static Route Configuration Task List....174
5.2.4 Static Route Configuration Examples....175
5.3 RIP....176
5.3.1 Introduction to RIP....176
5.3.2 RIP Configuration Task List....177
5.3.3 RIP Examples....182
5.3.4 RIP Troubleshooting....184
5.4 OSPF....185
5.4.1 Introduction to OSPF....185
5.4.2 OSPF Configuration Task List....187
5.4.3 OSPF Examples....191
5.4.4 OSPF Troubleshooting....198
5.5 BGP....198
5.5.1 Introduction to BGP....198
5.5.2 BGP Configuration Task List....201
5.5.3 BGP Configuration Examples....209
5.5.4 BGP Troubleshooting....218
5.6 IPv4 Black Hole Routing....218
5.6.1 Introduction to Black Hole Routing....218
5.6.2 IPv4 Black Hole Routing Configuration Task....218
5.6.3 Black Hole Routing Configuration Examples....219
5.6.4 Black Hole Routing Troubleshooting....219
5.7 GRE....220
5.7.1 Introduction to GRE Tunnels....220
5.7.2 Basic GRE Tunnel Configuration....220
5.7.3 GRE Tunnel Example....221
5.7.4 GRE Tunnel Example Using a Loopback Group....224
5.7.5 GRE Tunnel Troubleshooting....227
5.8 ECMP....228
5.8.1 Introduction to ECMP....228
5.8.2 ECMP Configuration Task List....228
5.8.3 ECMP Typical Example....228
5.8.4 ECMP Troubleshooting....230
5.9 BFD....230
5.9.1 Introduction to BFD....230
5.9.2 BFD Configuration Task List....230
5.9.3 BFD Examples....232
5.9.4 BFD Troubleshooting....234
5.10 BGP GR....235
5.10.1 Introduction to GR....235
5.10.2 GR Configuration Task List....236
5.10.3 Typical GR Example....237
5.11 OSPF GR....238
5.11.1 Introduction to OSPF GR....238
5.11.2 OSPF GR Configuration....239
5.11.3 OSPF GR Example....240
5.11.4 OSPF GR Troubleshooting....240

Chapter 6 Multicast Protocol Configuration....242

6.1 Multicast....242
6.1.1 Introduction to Multicast....242
6.1.2 Multicast Address....242
6.1.3 IP Multicast Packet Transmission....243
6.1.4 IP Multicast Application....244
6.2 PIM-DM....244
6.2.1 Introduction to PIM-DM....244
6.2.2 PIM-DM Configuration Task List....245
6.2.3 PIM-DM Configuration Examples....246
6.2.4 PIM-DM Troubleshooting....247
6.3 PIM-SM....248

6.3.1 Introduction to PIM-SM....248
6.3.2 PIM-SM Configuration Task List....249
6.3.3 PIM-SM Configuration Examples....251
6.3.4 PIM-SM Troubleshooting....252
6.4 MSDP....253
6.4.1 Introduction to MSDP....253
6.4.2 MSDP Configuration Task List....253
6.4.3 Basic MSDP Configuration....254
6.4.4 MSDP Peer Configuration....255
6.4.5 MSDP Packet Delivery Configuration....255
6.4.6 SA Cache Parameter Configuration....256
6.4.7 MSDP Configuration Examples....256
6.4.8 MSDP Troubleshooting....260
6.5 Anycast RP....261
6.5.1 Introduction to Anycast RP....261
6.5.2 Anycast RP Configuration Task List....261
6.5.3 Anycast RP Configuration Examples....263
6.5.4 ANYCAST RP Troubleshooting....264
6.6 PIM-SSM....264
6.6.1 Introduction to PIM-SSM....264
6.6.2 PIM-SSM Configuration Task List....264
6.6.3 PIM-SSM Configuration Examples....265
6.6.4 PIM-SSM Troubleshooting....266
6.7 DVMRP....266
6.7.1 Introduction to DVMRP....266
6.7.2 DVMRP Configuration Task List....267
6.7.3 DVMRP Configuration Examples....269
6.7.4 DVMRP Troubleshooting....269
6.8 DCSCM....270
6.8.1 Introduction to DCSCM....270
6.8.2 DCSCM Configuration Task List....270
6.8.3 DCSCM Configuration Examples....272
6.8.4 DCSCM Troubleshooting....273
6.9 IGMP....273
6.9.1 Introduction to IGMP....273
6.9.2 IGMP Configuration Task List....275
6.9.3 IGMP Configuration Examples....276
6.9.4 IGMP Troubleshooting....277
6.10 IGMP Snooping....277
6.10.1 Introduction to IGMP Snooping....277
6.10.2 IGMP Snooping Configuration Task List....277
6.10.3 IGMP Snooping Examples....279
6.10.4 IGMP Snooping Troubleshooting....282
6.11 IGMP Proxy....282
6.11.1 Introduction to IGMP Proxy....282
6.11.2 IGMP Proxy Configuration Task List....282
6.11.3 IGMP Proxy Examples....283
6.11.4 IGMP Proxy Troubleshooting....286
6.12 Multicast VLAN....286
6.12.1 Introduction to Multicast VLAN....286
6.12.2 Multicast VLAN Configuration Task List....286
6.12.3 Multicast VLAN Examples....287

Chapter 7 Security Function Configuration....289

7.1 ACL....289
7.1.1 Introduction to ACL....289
7.1.2 ACL Configuration Task List....289
7.1.3 ACL Example....300
7.1.4 ACL Troubleshooting....303
7.2 Self-Defined ACL....304

7.2.1 Introduction to Self-Defined ACL....304
7.2.2 Self-Defined ACL Configuration....304
7.2.3 Self-Defined ACL Example....305
7.2.4 Self-Defined ACL Troubleshooting....306
7.3 802.1x....306
7.3.1 Introduction to 802.1x....306
7.3.2 802.1x Configuration Task List....316
7.3.3 802.1x Application Example....317
7.3.4 802.1x Troubleshooting....321
7.4 MAC and IP Number Limitation on Ports and VLANs....321
7.4.1 Introduction to MAC and IP Number Limitation on Ports and VLANs....321
7.4.2 MAC and IP Number Limitation Configuration Task Sequence....322
7.4.3 MAC and IP Number Limitation Typical Example....323
7.4.4 MAC and IP Number Limitation Troubleshooting....324
7.5 AM....324
7.5.1 Introduction to AM Function....324
7.5.2 AM Function Configuration Task List....324
7.5.3 AM Function Example....326
7.5.4 AM Function Troubleshooting....326
7.6 Security Feature....326
7.6.1 Introduction to Security Feature....326
7.6.2 Security Feature Configuration....327
7.6.3 Security Feature Example....328
7.7 TACACS+....328
7.7.1 Introduction to TACACS+....328
7.7.2 TACACS+ Configuration Task List....328
7.7.3 TACACS+ Typical Example....329
7.7.4 TACACS+ Troubleshooting....330
7.8 RADIUS....330
7.8.1 Introduction to RADIUS....330
7.8.2 RADIUS Configuration Task List....331
7.8.3 RADIUS Typical Examples....333
7.8.4 RADIUS Troubleshooting....334
7.9 SSL....334
7.9.1 Introduction to SSL....334
7.9.2 SSL Configuration Task List....335
7.9.3 SSL Typical Example....336
7.9.4 SSL Troubleshooting....336
7.10 VLAN-ACL....337
7.10.1 Introduction to VLAN-ACL....337
7.10.2 VLAN-ACL Configuration Task List....337
7.10.3 VLAN-ACL Configuration Example....338
7.10.4 VLAN-ACL Troubleshooting....339
7.11 Captive Portal Authentication....340
7.11.1 Captive Portal Authentication Configuration....340
7.11.2 Accounting Function Configuration....343
7.11.3 Free-Resource Configuration....345
7.11.4 Authentication White-List Configuration....346
7.11.5 http-redirect-filter....347
7.11.6 Portal Non-Perception....349
7.11.7 Portal Escaping....351
7.12 MAB....355
7.12.1 Introduction to MAB....355
7.12.2 MAB Configuration Task List....356
7.12.3 MAB Example....357
7.12.4 MAB Troubleshooting....359
7.13 PPPoE Intermediate Agent....359
7.13.1 Introduction to PPPoE Intermediate Agent....359
7.13.2 PPPoE Intermediate Agent Configuration Task List....363
7.13.3 PPPoE Intermediate Agent Typical Application....364
7.13.4 PPPoE Intermediate Agent Troubleshooting....365

7.14 QoS....365
7.14.1 QoS....365
7.14.2 PBR....376
7.14.3 IPv6 PBR....378
7.15 Flow-Based Redirection....380
7.15.1 Introduction to Flow-Based Redirection....380
7.15.2 Flow-Based Redirection Configuration Task Sequence....380
7.15.3 Flow-Based Redirection Examples....381
7.15.4 Flow-Based Redirection Troubleshooting Help....381
7.16 Egress QoS....381
7.16.1 Introduction to Egress QoS....381
7.16.2 Egress QoS Configuration....383
7.16.3 Egress QoS Examples....386
7.16.4 Egress QoS Troubleshooting Help....387
7.17 Flexible QinQ....387
7.17.1 Introduction to Flexible QinQ....387
7.17.2 Flexible QinQ Configuration Task List....388
7.17.3 Flexible QinQ Example....389
7.17.4 Flexible QinQ Troubleshooting....390

Chapter 8 Reliability Configuration....391
8.1 MSTP....391
8.1.1 Introduction to MSTP....391
8.1.2 MSTP Configuration Task List....392
8.1.3 MSTP Example....395
8.1.4 MSTP Troubleshooting....399
8.2 VRRP....399
8.2.1 Introduction to VRRP....399
8.2.2 VRRP Configuration Task List....400
8.2.3 VRRP Typical Examples....401
8.2.4 VRRP Troubleshooting....402
8.3 MRPP....402
8.3.1 Introduction to MRPP....402
8.3.2 MRPP Configuration Task List....405
8.3.3 MRPP Typical Scenario....406
8.3.4 MRPP Troubleshooting....407
8.4 ULPP....408
8.4.1 Introduction to ULPP....408
8.4.2 ULPP Configuration Task List....409
8.4.3 ULPP Typical Examples....411
8.4.4 ULPP Troubleshooting....413
8.5 ULSM....413
8.5.1 Introduction to ULSM....413
8.5.2 ULSM Configuration Task List....414
8.5.3 ULSM Typical Example....415
8.5.4 ULSM Troubleshooting....416
8.6 ERPS....416
8.6.1 Introduction to ERPS....416
8.6.2 ERPS Configuration....421
8.6.3 ERPS Examples....422
8.6.4 ERPS Troubleshooting....423

Chapter 9 Debugging and Diagnosis Configuration....425
9.1 Monitor and Debug....425
9.1.1 Ping....425
9.1.2 Ping6....425
9.1.3 Traceroute....425
9.1.4 Traceroute6....425
9.1.5 Show....426
9.1.6 Debug....426

9.2 Reload Switch After Specified Time....427
9.2.1 Introduction to Reloading the Switch After a Specified Time....427
9.2.2 Reload Switch After Specified Time Task List....427
9.3 Debugging and Diagnosis for Packets Received and Sent by CPU....427
9.3.1 Introduction to Debugging and Diagnosis for Packets Received and Sent by CPU....427
9.3.2 Debugging and Diagnosis for Packets Received and Sent by CPU Task List....427
9.4 Debugging for CPU-Protect....428
9.4.1 Introduction to CPU-Protect....428
9.4.2 CPU-Protect Configuration....428
9.5 Info-center....429
9.5.1 Overview....429
9.5.2 Configuration Sequence....431
9.5.3 Configuration Examples....433
9.5.4 Notes....435
9.6 Mirror....435
9.6.1 Introduction to Mirror....435
9.6.2 Mirror Configuration Task List....436
9.6.3 Mirror Examples....436
9.6.4 Mirror Troubleshooting....437
9.7 RSPAN....437
9.7.1 Introduction to RSPAN....437
9.7.2 RSPAN Configuration Task List....438
9.7.3 Typical Examples of RSPAN....439
9.7.4 RSPAN Troubleshooting....441
9.8 ERSPAN....441
9.8.1 Introduction to ERSPAN....441
9.8.2 ERSPAN Configuration Task List....442
9.8.3 Typical Examples of ERSPAN....442
9.8.4 ERSPAN Troubleshooting....444
9.9 sFlow....444
9.9.1 Introduction to sFlow....444
9.9.2 sFlow Configuration Task List....445
9.9.3 sFlow Examples....446
9.9.4 sFlow Troubleshooting....446
9.10 NQA....446
9.10.1 Introduction to NQA....446
9.10.2 NQA Configuration Task List....448

Chapter 10 Network Time Management Configuration....453

10.1 NTP....453
10.1.1 Introduction to NTP Function....453
10.1.2 NTP Function Configuration Task List....453
10.1.3 Typical Examples of NTP Function....455
10.1.4 NTP Function Troubleshooting....456
10.2 SNTP....456
10.2.1 Introduction to SNTP....456
10.2.2 Typical Examples of SNTP Configuration....457
10.3 DNSv4/v6....458
10.3.1 Introduction to DNS....458
10.3.2 DNSv4/v6 Configuration Task List....458
10.3.3 Typical Examples of DNS....460
10.3.4 DNS Troubleshooting....461
10.4 Summer Time....462
10.4.1 Introduction to Summer Time....462
10.4.2 Summer Time Configuration Task Sequence....462
10.4.3 Examples of Summer Time....462
10.4.4 Summer Time Troubleshooting....462

Chapter 11 Virtualization Configuration....463

11.1 VSF....463

11.1.1 Overview....463
11.1.2 VSF Configuration....469
11.1.3 Typical VSF Example....472
11.1.4 VSF Troubleshooting....474

Chapter 12 IPv6 Configuration....476
12.1 DHCPv6....476
12.1.1 Introduction to DHCPv6....476
12.1.2 DHCPv6 Server Configuration....477
12.1.3 DHCPv6 Relay Configuration....478
12.1.4 DHCPv6 Prefix Delegation Server Configuration....478
12.1.5 DHCPv6 Prefix Delegation Client Configuration....480
12.1.6 DHCPv6 Configuration Examples....480
12.1.7 DHCPv6 Troubleshooting....483
12.2 DHCPv6 Option 37, 38....483
12.2.1 Introduction to DHCPv6 Option 37, 38....483
12.2.2 DHCPv6 Option 37, 38 Configuration Task List....484
12.2.3 DHCPv6 Option 37, 38 Examples....488
12.2.4 DHCPv6 Option 37, 38 Troubleshooting....490
12.3 Prevent ND Spoofing....491
12.3.1 Overview....491
12.3.2 Prevent ND Spoofing Configuration....491
12.3.3 Prevent ND Spoofing Example....491
12.4 RIPng....491
12.4.1 Introduction to RIPng....491
12.4.2 RIPng Configuration Task List....493
12.4.3 RIPng Configuration Examples....496
12.4.4 RIPng Troubleshooting....498
12.5 OSPFv3....498
12.5.1 Introduction to OSPFv3....498
12.5.2 OSPFv3 Configuration Task List....500
12.5.3 OSPFv3 Examples....504
12.5.4 OSPFv3 Troubleshooting....505
12.6 BGP4+....506
12.6.1 Introduction to MBGP4+....506
12.6.2 MBGP4+ Configuration Task List....506
12.6.3 MBGP4+ Examples....507
12.6.4 MBGP4+ Troubleshooting....508
12.7 IPv6 Black Hole Routing....508
12.7.1 Introduction to Black Hole Routing....508
12.7.2 IPv6 Black Hole Routing Configuration Tasks....508
12.7.3 Black Hole Routing Configuration Examples....509
12.7.4 Black Hole Routing Troubleshooting....509
12.8 IPv6 Multicast Protocol....510
12.8.1 PIM-DM6....510
12.8.2 PIM-SM6....513
12.8.3 Anycast RP v6....519
12.8.4 PIM-SSM6....522
12.8.5 IPv6 DCSCM....524
12.8.6 MLD....527
12.8.7 MLD Snooping....530
12.9 IPv6 Security RA....533
12.9.1 Introduction to IPv6 Security RA....533
12.9.2 IPv6 Security RA Configuration Task Sequence....534
12.9.3 IPv6 Security RA Typical Examples....534
12.9.4 IPv6 Security RA Troubleshooting....535
12.10 SAVI Configuration....535
12.10.1 Introduction to SAVI....535
12.10.2 SAVI Configuration....535
12.10.3 SAVI Typical Application....538
12.10.4 SAVI Troubleshooting....539

12.11 IPv6 VRRPv3.....539
12.11.1 Introduction to VRRPv3.....539
12.11.2 VRRPv3 Configuration.....541
12.11.3 VRRPv3 Typical Examples.....543
12.11.4 VRRPv3 Troubleshooting.....543

Chapter 13 VPN Configuration.....545
13.1 MPLS Overview.....545
13.1.1 MPLS Overview.....545
13.2 LDP.....551
13.2.1 LDP Introduction.....551
13.2.2 LDP Configuration.....558
13.2.3 LDP Typical Examples.....562
13.2.4 LDP Troubleshooting.....564
13.3 MPLS VPN.....565
13.3.1 BGP/MPLS VPN Introduction.....565
13.3.2 BGP MPLS VPN Configuration.....572
13.3.3 BGP MPLS VPN Typical Examples.....578
13.3.4 MPLS BGP VPN Troubleshooting.....587
13.4 Public Network Access of MPLS VPN.....588
13.4.1 Public Network Access Introduction.....588
13.4.2 Public Network Access Configuration.....590
13.4.3 Public Network Access Typical Examples.....591
13.4.4 Public Network Access Troubleshooting.....597
13.5 VPLS and VPWS.....598
13.5.1 VPLS Overview.....598
13.5.2 VPLS Configuration.....604
13.5.3 Typical Examples of VPLS.....606
13.5.4 VPLS Troubleshooting.....615

Chapter 14 Data Center Configuration.....617
14.1 MC-LAG.....617
14.1.1 Introduction to MC-LAG.....617
14.1.2 Typical Applications.....617
14.1.3 MC-LAG Configuration.....619
14.1.4 Typical MC-LAG Applications.....621
14.1.5 MC-LAG Troubleshooting.....627
14.2 NETCONF.....628
14.2.1 Introduction to NETCONF.....628
14.2.2 NETCONF Configuration.....629
14.2.3 NETCONF Example.....630
14.2.4 NETCONF Troubleshooting.....633
14.3 Telemetry.....633
14.3.1 Telemetry Overview.....633
14.3.2 Telemetry Configuration.....635
14.3.3 Typical Telemetry Application.....637
14.4 VXLAN and EVPN.....638
14.4.1 VXLAN and EVPN Overview.....638
14.4.2 VXLAN and EVPN Configuration.....641
14.4.3 VXLAN Examples.....646
14.4.4 VXLAN Troubleshooting.....662

Chapter 1 Basic Management Configuration

1.1 Switch Management

1.1.1 Management Options

After installing the switch, configure it for network management. The switch supports two management options: in-band management and out-of-band management.

1.1.1.1 Out-Of-Band Management

Out-of-band management is management through the Console port. Out-of-band management is generally used for the initial configuration of the switch, or when in-band management is not available. For example, an IP address must be assigned to the switch through the Console port before the switch can be accessed through Telnet.

To manage the switch through the Console port, perform the following steps:

Step 1: Set up the environment.

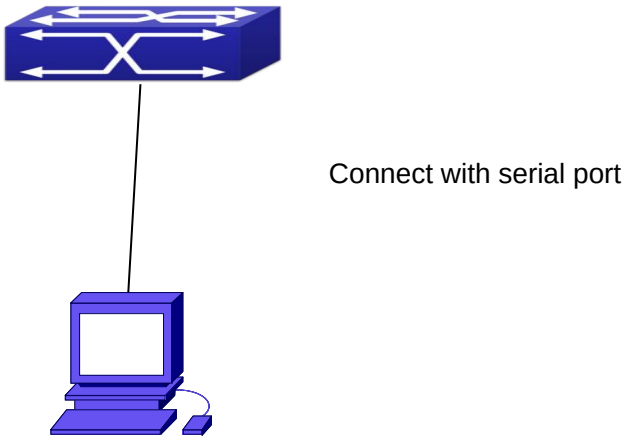


Figure 1-1 Out-of-band Management Configuration Environment

As shown in the figure above, the serial port (RS-232) of the PC is connected to the switch with the serial cable provided. The following table lists the devices used in the connection.

Device Name	Description
PC machine	A PC with a keyboard and an RS-232 serial port, with a terminal emulator installed, such as HyperTerminal included in Windows 9x/NT/2000/XP.
Serial cable	One end connects to the RS-232 serial port of the PC; the other end connects to the Console port of the switch.
Switch	A working Console port.

Step 2: Start HyperTerminal.

After the connection is established, open the HyperTerminal program included in Windows. The following example uses the HyperTerminal program included in Windows XP.

1) Click Start > All Programs > Accessories > Communications > HyperTerminal.

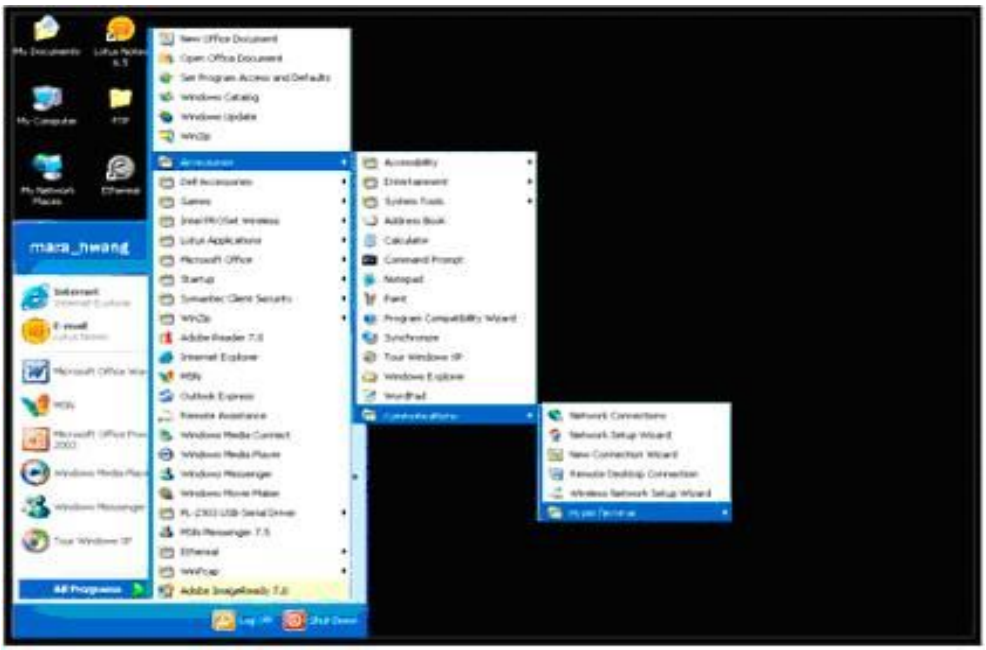


Figure 1-2 Opening HyperTerminal

2) Enter a name for the HyperTerminal connection, such as "Switch".

Figure 1-3 Opening HyperTerminal

3) In the "Connect using" drop-down list, select the RS-232 serial port used by the PC, such as COM1, and click "OK".

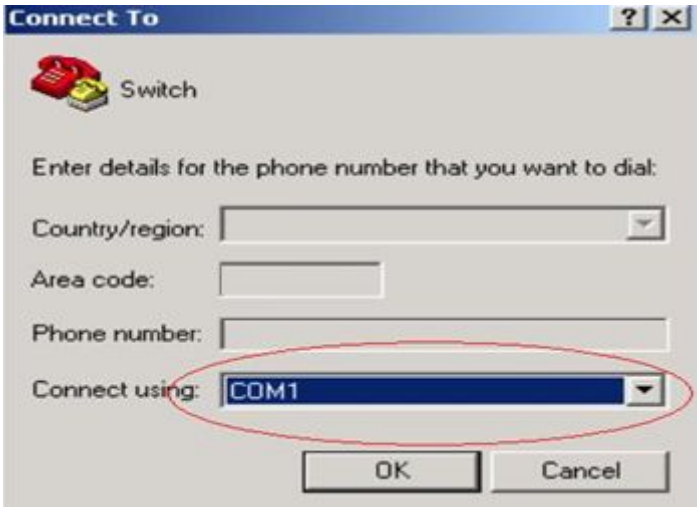


Figure 1-4 Opening HyperTerminal

4) In the COM1 Properties dialog box, set "Bits per second" to "115200", "Data bits" to "8", "Parity" to "None", "Stop bits" to "1", and "Flow control" to "None"; or click "Restore Defaults". Then click "OK".

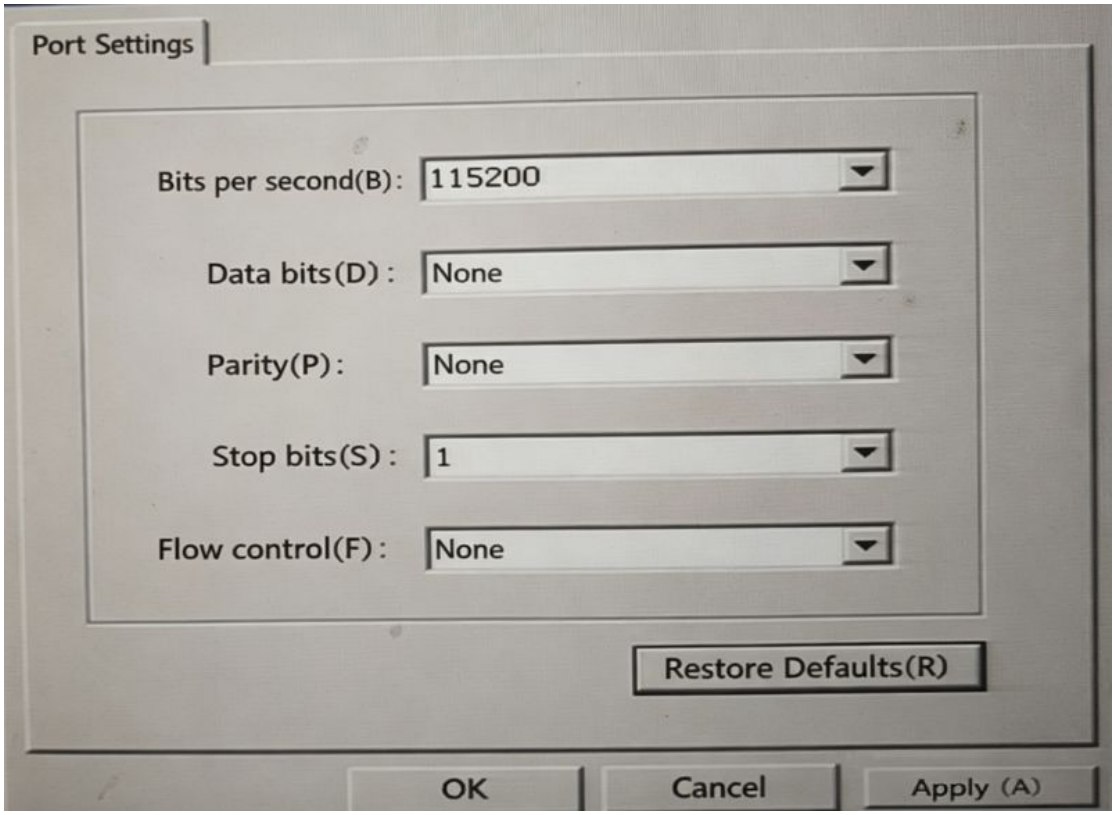


Figure 1-5 Opening HyperTerminal

Step 3: Enter the switch CLI.

Power on the switch. The following information appears in the HyperTerminal window, and then the CLI is available.

```
Testing RAM...
0x077C0000 RAM OK
Loading MiniBootROM...
Attaching to file system ...
Loading nos.img ...  done.
Booting......
Starting at 0x10000...
Attaching to file system ...
```

```
……
--- Performing Power-On Self Tests (POST) ---
DRAM Test.....................PASS!
PCI Device 1 Test.............PASS!
FLASH Test....................PASS!
FAN Test......................PASS!
Done All Pass.
------------------ DONE ---------------------
Current time is SUN JAN 01 00:00:00 2006
……
Switch>
```

You can now enter commands to manage the switch. For detailed descriptions of the commands, see the following chapters.

1.1.1.2 In-band Management

In-band management refers to managing the switch by logging in to it through Telnet or HTTP, or by using SNMP network management software. In-band management allows devices connected to the switch to manage it. If in-band management fails because of configuration changes on the switch, out-of-band management can be used to configure and manage the switch.

1.1.1.2.1 Management via Telnet

To manage the switch through Telnet, the following conditions must be met:

1) The switch has an IPv4/IPv6 address configured.
2) The IP address of the host (Telnet client) and the IPv4/IPv6 address of the switch VLAN interface are in the same network segment.
3) If condition 2) is not met, the Telnet client can reach an IPv4/IPv6 address of the switch through other devices, such as a router.

The switch is a Layer 3 switch and can be configured with multiple IPv4/IPv6 addresses; for the configuration method, see the related chapter. The following example assumes the factory default state of the switch, in which only VLAN 1 exists.

The following steps describe how a Telnet client connects to the VLAN 1 interface of the switch through Telnet (IPv4 example):

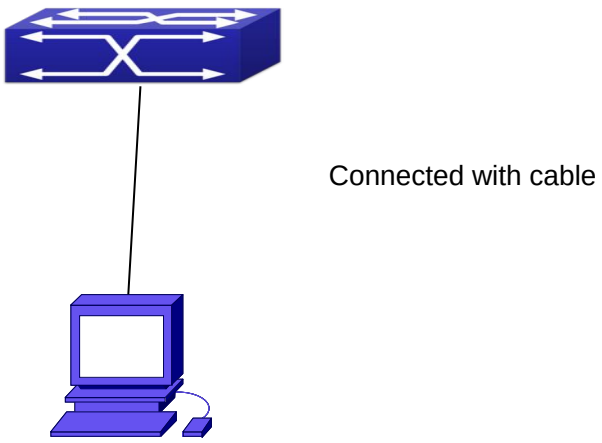


Figure 1-6 Managing the Switch through Telnet

Step 1: Configure the IP addresses and enable the Telnet server on the switch.

First, configure the IP address of the host. It must be in the same network segment as the IP address of the switch VLAN 1 interface. For example, if the IP address of the switch VLAN 1 interface is

10.1.128.251/24, the host IP address can be 10.1.128.252/24. Run "ping 10.1.128.251" on the host to verify connectivity; if the ping fails, find the cause.

The commands for configuring the IP address of the VLAN 1 interface are listed below. Before in-band management can be used, an IP address must be configured on the switch through out-of-band management (that is, through the Console port). The configuration commands are as follows (unless otherwise specified, the switch prompt is assumed to be "Switch" throughout this guide):

```
Switch>
Switch>enable
Switch#config
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ip address 10.1.128.251 255.255.255.0
Switch(Config-if-Vlan1)#no shutdown
```

To enable the Telnet server, run the **telnet-server enable** command in Global Mode:

```
Switch>enable
Switch#config
Switch(config)# telnet-server enable
```

Step 2: Run the Telnet client program.

Run the Telnet client program included in Windows and specify the Telnet target.

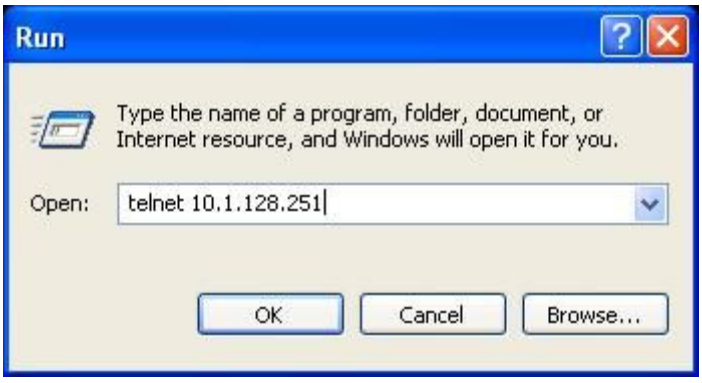


Figure 1-7 Running the Telnet Client Program Included in Windows

Step 3: Log in to the switch.

Log in to the Telnet configuration interface. A valid username and password are required; otherwise, the switch rejects the Telnet access. This protects the switch from unauthorized access. Therefore, when Telnet is used to configure and manage the switch, configure the username and password of authorized Telnet users with the command **username <username> privilege <privilege> [password (0|7) <password>]**, and enable local authentication with the command **authentication line vty login local**. The privilege option is required and must be 15. For example, if an authorized user has the username "test" and the password "test", the configuration is as follows:

```
Switch>enable
Switch#config
Switch(config)#username test privilege 15 password 0 test
Switch(config)#authentication line vty login local
```

After a valid username and password are entered in the Telnet configuration interface, the Telnet user enters the CLI of the switch. The commands available in the Telnet CLI are the same as those in the Console interface.

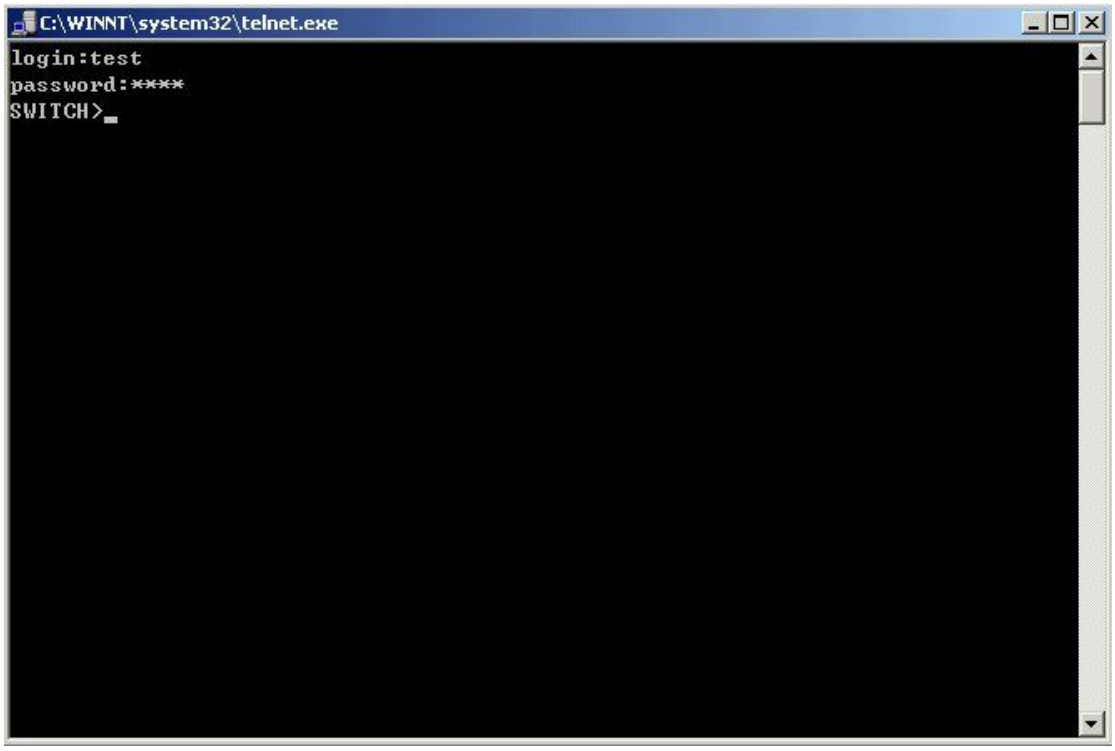


Figure 1-8 Telnet Configuration Interface

1.1.1.2.2 Management via HTTP

To manage the switch through HTTP, the following conditions must be met:

1) The switch has an IPv4/IPv6 address configured.
2) The IPv4/IPv6 address of the host (HTTP client) and the IPv4/IPv6 address of the switch VLAN interface are in the same network segment.
3) If condition 2) is not met, the HTTP client must reach an IPv4/IPv6 address of the switch through other devices, such as a router.

As with Telnet management, once the host can ping (or ping6) an IPv4/IPv6 address of the switch and the correct login password is entered, the host can access the switch through HTTP. The configuration steps are as follows:

Step 1: Configure the IP addresses and enable the HTTP server on the switch.

To configure the IP address of the switch through out-of-band management, see the Telnet management section.

To enable Web configuration, run the **ip http server** command in Global Mode:

```
Switch>enable
Switch#config
Switch(config)#ip http server
```

Step 2: Access the switch through HTTP from the host.

Open a Web browser on the host and enter the IP address of the switch, for example "10.1.128.251".

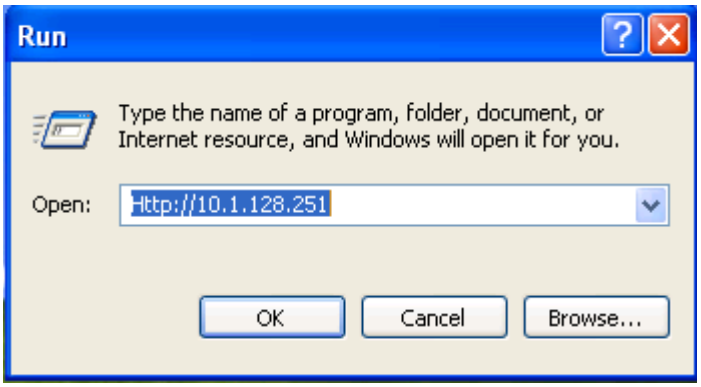


Figure 1-9 Accessing the Switch through HTTP

To access the switch through an IPv6 address, Firefox 1.5 or later is recommended. For example, if the IPv6 address of the switch is 3ffe:506:1:2::3, enter http://[3ffe:506:1:2::3]; the address must be enclosed in square brackets.

Step 3: Log in to the switch.

Log in to the Web configuration interface. A valid username and password are required; otherwise, the switch rejects the HTTP access. This protects the switch from unauthorized access. Therefore, when the Web interface is used to configure and manage the switch, configure the username and password of authorized users with the command **username <username> privilege <privilege> [password (0|7) <password>]**, and enable local authentication with the command **authentication line web login local**. The privilege option is required and must be 15. For example, if an authorized user has the username "admin" and the password "admin", the configuration is as follows:

```
Switch>enable
Switch#config
Switch(config)#username admin privilege 15 password 0 admin
Switch(config)#authentication line web login local
```

The Web login interface of the ECS8830F/ECS8854F is shown below:

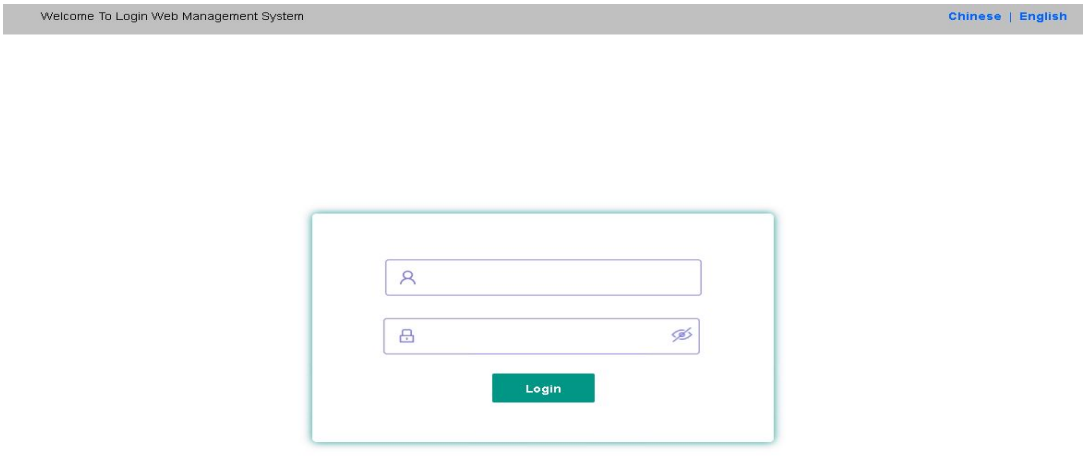


Figure 1-10 Web Login Interface

Enter the correct username and password; the main Web configuration interface is displayed as shown below.

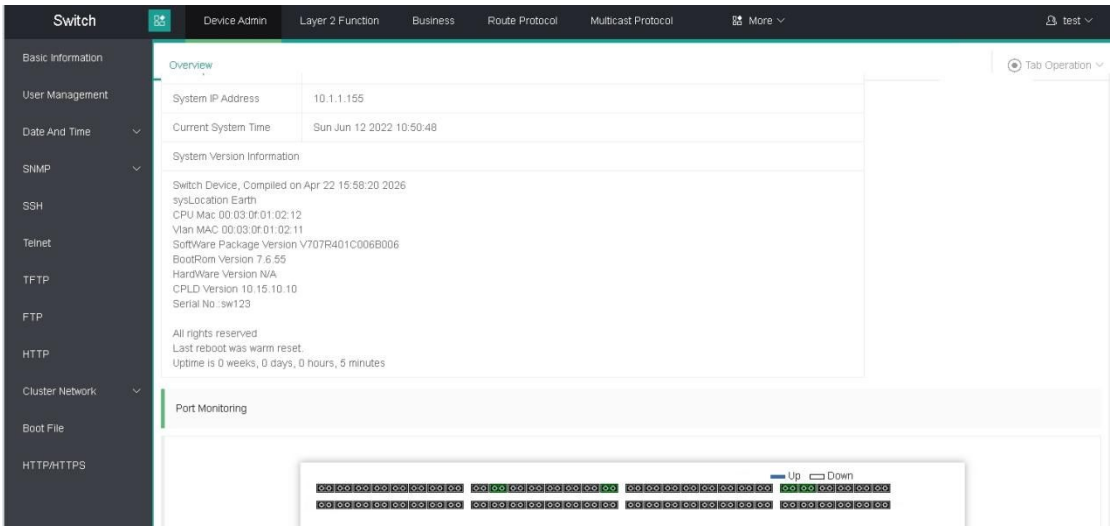


Figure 1-11 Main Web Configuration Interface

Note: When configuring the switch, use English letters for the switch name.

1.1.1.2.3 Manage the Switch via SNMP Network Management Software

To manage the switch with SNMP network management software, the following conditions must be met:

1) IP addresses are configured on the switch.
2) The IP address of the client host and the IP address of the VLAN interface of the switch are in the same network segment.
3) If condition 2) is not met, the client can reach an IP address of the switch through other devices, such as routers.
4) SNMP is enabled.

The host running the SNMP network management software must be able to ping the IP address of the switch, so that the software can discover the switch and perform read and write operations on it. Details on managing switches with SNMP network management software are not covered in this guide; see the user manual of the SNMP network management software.

1.1.2 CLI Interface

The switch provides three management interfaces: the CLI (Command Line Interface), the Web interface, and SNMP network management software. This chapter describes the CLI in detail. The Web interface provides functions similar to the CLI and is not described here; for SNMP network management software, see its user manual.

The CLI is familiar to most users. As mentioned above, out-of-band management and Telnet login both use the CLI to manage the switch.

The CLI is provided by the Shell program, which consists of a set of configuration commands. The commands are grouped by function, and each group corresponds to a configuration mode. The Shell of the switch provides the following features:

- ☞ Configuration modes
- ☞ Configuration syntax
- ☞ Shortcut keys
- ☞ Help function
- ☞ Input verification
- ☞ Fuzzy match support

1.1.2.1 Configuration Modes

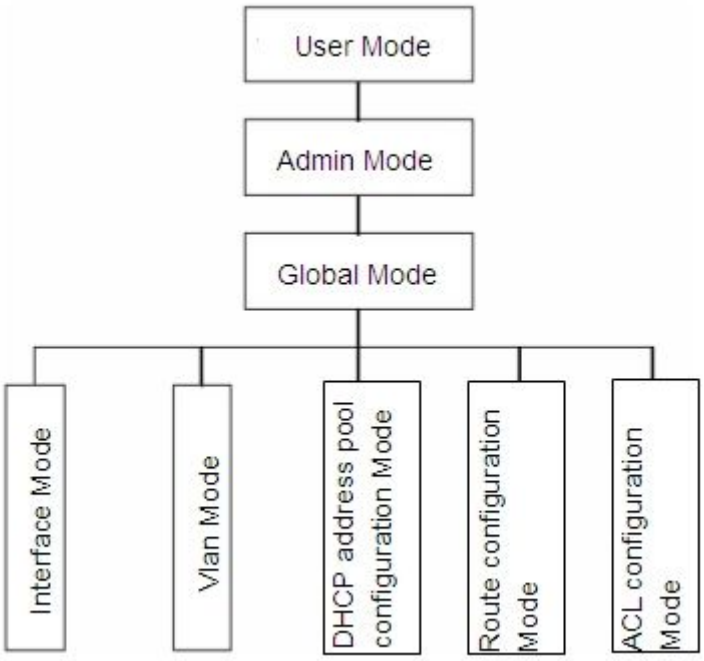


Figure 1-12 Shell Configuration Modes

1.1.2.1.1 User Mode

When you enter the CLI, you first enter the user login system. A common user enters User Mode by default; the prompt is "Switch>", where ">" indicates User Mode. Running the **exit** command in Admin Mode also returns to User Mode.

In User Mode, no configuration is allowed; only the clock and version information of the switch can be queried.

1.1.2.1.2 Admin Mode

An admin user enters Admin Mode by default after login. From User Mode, run the **enable** command and enter the admin password for the corresponding access level, if a password is set, to enter Admin Mode; the prompt is "Switch#". Running the **exit** command in Global Mode also returns to Admin Mode. The shortcut key Ctrl+Z returns to Admin Mode from any configuration mode (except User Mode).

In Admin Mode, you can query the switch configuration, the connection status, and the traffic statistics of all ports, and enter Global Mode to modify the switch configuration. Therefore, set a password for entering Admin Mode to prevent unauthorized access and malicious changes to the switch.

1.1.2.1.3 Global Mode

Run the **config** command in Admin Mode to enter Global Mode; the prompt is "Switch(config)#". Running the **exit** command in other configuration modes, such as Port Mode and VLAN Mode, returns to Global Mode.

In Global Mode, you can perform global configuration, such as the MAC address table, port mirroring, VLAN creation, IGMP snooping, and STP, and you can enter Port Mode to configure interfaces.

Interface Mode

Run the **interface** command in Global Mode to enter the specified interface mode. The switch provides three interface types, each with its own configuration mode: 1. VLAN interface; 2. Ethernet port; 3. port channel.

Interface Type	Entry	Operations	Exit
VLAN Interface	Run the **interface vlan <*Vlan-id*>** command in Global Mode.	Configures the IP addresses of the switch, and so on.	Run the **exit** command to return to Global Mode.
Ethernet Port	Run the **interface ethernet <interface-list>** command in Global Mode.	Configures the duplex mode, speed, and other settings of the Ethernet port.	Run the **exit** command to return to Global Mode.
Port channel	Run the **interface port-channel <port-channel-number>** command in Global Mode.	Configures port channel settings, such as duplex mode and speed.	Run the **exit** command to return to Global Mode.

VLAN Mode

Run the **vlan <vlan-id>** command in Global Mode to enter VLAN Mode, in which you can configure the member ports of the VLAN. Run the **exit** command to return from VLAN Mode to Global Mode.

DHCP Address Pool Mode

Run the **ip dhcp pool <name>** command in Global Mode to enter DHCP Address Pool Mode; the prompt is "Switch(Config-<name>-dhcp)#". In DHCP Address Pool Mode, you can configure the properties of the DHCP address pool. Run the **exit** command to return to Global Mode.

Router Mode

Routing Protocol	Entry	Operations	Exit
RIP Routing Protocol	Run the **router rip** command in Global Mode.	Configures RIP parameters.	Run the **exit** command to return to Global Mode.
OSPF Routing Protocol	Run the **router ospf** command in Global Mode.	Configures OSPF parameters.	Run the **exit** command to return to Global Mode.
BGP Routing Protocol	Run the **router bgp <AS mumber>** command in Global Mode.	Configures BGP parameters.	Run the **exit** command to return to Global Mode.

ACL Mode

ACL Type	Entry	Operations	Exit
Standard IP ACL Mode	Run the **ip access-list standard** command in Global Mode.	Configures parameters in Standard IP ACL Mode.	Run the **exit** command to return to Global Mode.
Extended IP ACL Mode	Run the **ip access-list extanded** command in Global Mode.	Configures parameters in Extended IP ACL Mode.	Run the **exit** command to return to Global Mode.

1.1.2.2 Configuration Syntax

The switch provides a variety of configuration commands. Although the commands differ, they all follow the same syntax. The general command format is as follows:

cmdtxt <*variable*> {enum1 | … | enumN } [option1 | … | optionN]

Conventions: **cmdtxt** in bold indicates a command keyword; **<variable>** indicates a variable parameter; **{enum1 | … | enumN}** indicates a mandatory parameter that must be selected from the set **enum1~enumN**; and square brackets ([ ]) in **[option1 | … | optionN]** indicate an optional parameter. A command line can combine "< >", "{ }", and "[ ]", such as **[<variable>]**, **{enum1 <variable>| enum2}**, and **[option1 [option2]]**.

Examples of actual configuration commands:

- **show version**: no parameters. This command has only keywords and no parameters; enter the command to run it.
- **vlan <vlan-id>**: a parameter value is required after the keyword.
- **firewall {enable | disable}**: enter **firewall enable** or **firewall disable**.
- **snmp-server community {ro | rw} <string>**: the following forms are possible:

```
snmp-server community ro <string>
snmp-server community rw <string>
```

1.1.2.3 Shortcut Key Support

The switch provides several shortcut keys for configuration, such as the Up, Down, Left, Right, and Space keys. If the terminal does not recognize the Up and Down keys, use Ctrl+P and Ctrl+N instead.

Key(s)	Function	
Back Space	Deletes the character before the cursor and moves the cursor back.	
Up “↑”	Shows the previous command entered. Up to ten recently entered commands can be shown.	
Down “↓”	Shows the next command entered. After using the Up key to recall previous commands, use the Down key to go forward to the next command.	
Left “←”	Moves the cursor one character to the left.	Use the Left and Right keys to modify an entered command.
Right “→”	Moves the cursor one character to the right.	
Ctrl +p	Same as the Up key "↑".	
Ctrl +n	Same as the Down key "↓".	
Ctrl +b	Same as the Left key "←".	
Ctrl +f	Same as the Right key "→".	
Ctrl +z	Returns directly to Admin Mode from any other configuration mode (except User Mode).	
Ctrl +c	Interrupts the running command, such as ping.	
Tab	When part of a command or keyword is entered, press Tab to complete it if there is no conflict.	

1.1.2.4 Help Function

The switch provides two ways to access help: the **help** command and "?".

Access to Help	Usage and function
Help	At any command prompt, enter **help** and press Enter to display a brief description of the help system.
“?”	1． At any command prompt, enter "?" to display the list of commands in the current mode with brief descriptions. 2． Enter "?" after a command keyword, separated by a space. If a parameter is expected at that position, its type and range are displayed; if a keyword is expected, the available keywords with brief descriptions are displayed; if "<cr>" is displayed, the command is complete and can be run by pressing Enter. 3． Enter "?" immediately after a string to display all commands that begin with that string.

1.1.2.5 Input Verification

1.1.2.5.1 Returned Information: success

The Shell checks the syntax of every command entered. If a correct command is entered in the corresponding mode and runs successfully, nothing is returned.

1.1.2.5.2 Returned Information: error

Output error message	Explanation
Unrecognized command or illegal parameter!	The command does not exist, or a parameter has an incorrect range, type, or format.
Ambiguous command	More than one interpretation is possible for the current input.
Invalid command or parameter	The command is recognized, but no valid parameter record is found.
This command is not exist in current mode	The command is recognized, but it cannot be used in the current mode.
Please configure precursor command "*" at first!	The command is recognized, but the prerequisite command has not been configured.
syntax error : missing "" before the end of command line!	Quotation marks are not used in pairs.

1.1.2.5.3 Fuzzy Match Support

The Shell supports fuzzy matching of commands and keywords. It recognizes a command or keyword correctly if the entered string causes no conflict.

For example:

1) For the command **show interfaces status ethernet1/0/1**, entering **sh in status ethernet1/0/1** works.
2) However, for the command **show running-config**, the system reports "> Ambiguous command!" if only **show r** is entered, because the Shell cannot tell whether **show run** or **show running-config** is meant. The Shell recognizes the command if **sh ru** is entered.

1.2 Basic Switch Configuration

1.2.1 Basic Configuration

Basic switch configuration includes commands for entering and exiting Admin Mode and interface modes, configuring and displaying the switch clock, and displaying the version information of the switch.

Command	Explanation
User Mode/Admin Mode	
enable [<*1-15*>] **disable**	The **enable** command enters Admin Mode from User Mode or changes the privilege level of the user. The **disable** command exits Admin Mode.
Admin Mode	
config [terminal]	Enters Global Mode from Admin Mode.
All Modes	
exit	Exits the current mode and returns to the previous mode; for example, from Global Mode to Admin Mode, and from Admin Mode to User Mode.
show privilege	Shows the privilege level of the current user.

All Modes except User Mode and Admin Mode	
end	Exits the current mode and returns to Admin Mode (in any mode other than User Mode and Admin Mode).
Admin Mode	
clock set <*HH:MM:SS*> [YYYY.MM.DD]	Sets the system date and time.
show version	Displays the version information of the switch.
set default	Restores the factory default configuration.
write	Saves the current configuration to flash memory.
reload	Hot-restarts the switch.
show cpu usage	Shows the CPU usage.
show cpu utilization	Shows the current CPU utilization.
show memory usage	Shows the memory usage.
Global Mode	
banner motd <LINE> **no banner motd**	Configures the message displayed after a Telnet or Console user passes login authentication.
web-auth privilege <1-15> **no web-auth privilege**	Configures the privilege level of users who log in to the switch through the Web.

1.2.2 Telnet Management

1.2.2.1 Telnet

1.2.2.1.1 Introduction to Telnet

Telnet is a simple remote terminal protocol for remote login. With Telnet, a user can log in to a remote host by its IP address or host name from a local workstation. Telnet sends the user's keystrokes to the remote host and the remote host's output to the user's screen over a TCP connection. This service is transparent: to the user, the keyboard and monitor appear to be connected directly to the remote host.

Telnet uses the client-server model: the local system is the Telnet client and the remote host is the Telnet server. The switch can act as either a Telnet server or a Telnet client.

When the switch acts as a Telnet server, users can log in to it with the Telnet client program included in Windows or other operating systems, as described in the In-band Management section. As a Telnet server, the switch allows up to 5 Telnet client TCP connections.

As a Telnet client, the switch can log in to other remote hosts with the **telnet** command in Admin Mode. The switch can establish a TCP connection to only one remote host at a time. To connect to another remote host, the current TCP connection must be closed first.

1.2.2.1.2 Telnet Configuration Task List

1. Configure the Telnet server
2. Telnet to a remote host from the switch

1. Configure the Telnet server

Command	Explanation
Global Mode	
telnet-server enable	Enables the Telnet server on the switch.

no telnet-server enable	The no command disables the Telnet server.
username <*user-name*> [privilege <*privilege*>] [password [0 | 7] <*password*>] **no username <*username*>**	Configures the Telnet username and password. The no command deletes the Telnet user.
aaa authorization config-commands **no aaa authorization config-commands**	Enables command authorization for users who log in through VTY (Telnet and SSH). The no command disables this function. Authorization is requested for commands only when this command is enabled and a command authorization method is configured.
authentication securityip <*ip-addr*> **no authentication securityip <*ip-addr*>**	Configures a secure IP address from which users can log in to the switch through Telnet. The no command deletes the secure Telnet address.
authentication securityipv6 <*ipv6-addr*> **no authentication securityipv6 <*ipv6-addr*>**	Configures a secure IPv6 address from which users can log in to the switch through Telnet. The no command deletes the secure Telnet address.
authentication ip access-class {<num-std>|<name>} **no authentication ip access-class**	Binds a standard IP ACL to Telnet/SSH/Web login. The no command removes the ACL binding.
authentication ipv6 access-class {<num-std>|<name>} in **no authentication ipv6 access-class**	Binds a standard IPv6 ACL to Telnet/SSH/Web login. The no command removes the ACL binding.
authentication line {console | vty | web} login method1 [method2 …] **no authentication line {console | vty | web} login**	Configures the login authentication method list for the Console, VTY (Telnet), or Web.
authentication enable method1 [method2 …] **no authentication enable**	Configures the authentication method list for the enable command.
authorization line {console | vty | web} exec method1 [method2 …] **no authorization line {console | vty | web} exec**	Configures the exec authorization method list for the Console, VTY (Telnet), or Web.
authorization line vty command <1-15> {local | radius | tacacs} (none|) **no authorization line vty command <1-15>**	Configures the command authorization method and the authorization priority for users who log in through VTY (Telnet and SSH). The no command restores the default method.
accounting line {console | vty} command <*1-15*> {start-stop | stop-only | none} method1 [method2…] **no accounting line {console | vty} command <*1-15*>**	Configures the accounting method list.
Admin Mode	
terminal monitor **terminal no monitor**	Displays debugging information for Telnet clients logged in to the switch. The no command disables the debugging information.
show users	Shows information about users logged in through Telnet or SSH, including the line number, username, and user IP address.
clear line vty <0-31>	Clears the user logged in on the

	specified line, forcing a Telnet or SSH user offline.

2. Telnet to a remote host from the switch

Command	Explanation
Admin Mode	
telnet [vrf <*vrf-name*>] {<*ip-addr*> | <*ipv6-addr*> | host <*hostname*>} [<*port*>]	Logs in to a remote host with the Telnet client of the switch.

1.2.2.2 SSH

1.2.2.2.1 Introduction to SSH

SSH (Secure Shell) is a protocol that provides secure remote access to network devices. It runs over reliable TCP/IP. Through key distribution, authentication, and encryption between the SSH server and the SSH client, a secure connection is established, and information transferred over it is protected from interception and decryption. The switch complies with SSH 2.0 and supports SSH 2.0 client software such as SSH Secure Client and PuTTY, which can be used to manage the switch remotely.

The switch currently supports RSA authentication, 3DES encryption, and SSH user password authentication.

1.2.2.2.2 SSH Server Configuration Task List

Command	Explanation
Global Mode	
ssh-server enable **no ssh-server enable**	Enables SSH on the switch. The no command disables SSH.
username <*username*> [privilege <*privilege*>] [password [0 | 7] <*password*>] **no username <*username*>**	Configures the username and password used by SSH client software to log in to the switch. The no command deletes the username.
ssh-server timeout <*timeout*> **no ssh-server timeout**	Configures the SSH authentication timeout. The no command restores the default timeout.
ssh-server authentication-retires <*authentication-retires*> **no ssh-server authentication-retries**	Configures the number of SSH authentication retries. The no command restores the default number of retries.
ssh-server host-key create rsa modulus <*moduls*>	Generates a new RSA host key on the SSH server.
Admin Mode	
terminal monitor **terminal no monitor**	Displays SSH debugging information on the SSH client side. The no command stops displaying it.
show crypto key	Shows the SSH key.
crypto key clear rsa	Clears the SSH key.

1.2.2.2.3 Example of SSH Server Configuration

Example 1:

Requirement: Enable the SSH server on the switch, and run SSH 2.0 client software, such as Secure Shell Client or PuTTY, on the terminal. Log in to the switch from the client with a username and password.

Configure the IP address, add an SSH user, and enable the SSH service on the switch. The SSH 2.0 client can then log in to the switch with the username and password to configure it.

```
Switch(config)#ssh-server enable
```

```
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ip address 100.100.100.200 255.255.255.0
Switch(Config-if-Vlan1)#exit
Switch(config)#username test privilege 15 password 0 test
```

In IPv6 networks, the terminal must run SSH client software that supports IPv6, such as PuTTY6. No switch configuration other than assigning an IPv6 address is required.

1.2.3 Configure Switch IP Addresses

By default, all Ethernet ports of the switch are Layer 2 (data link layer) ports that perform Layer 2 forwarding. A VLAN interface is a Layer 3 interface that can be assigned an IP address, which is also an IP address of the switch. All VLAN interface commands are configured in VLAN Interface Mode. The switch provides three IP address configuration methods:

- ☞ Manual
- ☞ BOOTP
- ☞ DHCP

Manual configuration assigns an IP address to the switch manually.

In BOOTP/DHCP mode, the switch acts as a BOOTP/DHCP client and broadcasts BOOTP requests to the BOOTP/DHCP servers, which assign an address when they receive the request. The switch can also act as a DHCP server and dynamically assign network parameters, such as IP addresses, gateway addresses, and DNS server addresses, to DHCP clients. DHCP server configuration is described in a later chapter.

1.2.3.1 Switch IP Addresses Configuration Task List

1. Create a VLAN interface
2. Manual configuration
3. BOOTP configuration
4. DHCP configuration

1. Create a VLAN interface

Command	Explanation
Global Mode	
interface vlan <*vlan-id*> **no interface vlan <*vlan-id*>**	Creates a VLAN interface (Layer 3 interface). The no command deletes the VLAN interface.

2. Manual configuration

Command	Explanation
VLAN Interface Mode	
ip address <*ip_address*> <*mask*> [secondary] **no ip address <*ip_address*> <*mask*> [secondary]**	Configures the IP address of the VLAN interface. The no command deletes the IP address of the VLAN interface.
ipv6 address <*ipv6-address / prefix-length*> [eui-64] **no ipv6 address <*ipv6-address / prefix-length*>**	Configures an IPv6 address, including an aggregatable global unicast address, a site-local address, or a link-local address. The no command deletes the IPv6 address.

3. BOOTP configuration

Command	Explanation
VLAN Interface Mode	

ip bootp-client enable **no ip bootp-client enable**	Enables the switch as a BOOTP client to obtain the IP address and gateway address through BOOTP negotiation. The no command disables the BOOTP client.

4. DHCP configuration

Command	Explanation
VLAN Interface Mode	
ip bootp-client enable **no ip bootp-client enable**	Enables the switch as a DHCP client to obtain the IP address and gateway address through DHCP negotiation. The no command disables the DHCP client.

1.2.4 SNMP Configuration

1.2.4.1 Introduction to SNMP

SNMP (Simple Network Management Protocol) is a standard network management protocol widely used in computer networks. SNMP has evolved through several versions. SNMPv1 [RFC1157] is the first version and is widely adopted by vendors for its simplicity and ease of implementation. SNMPv2c enhances SNMPv1 and supports hierarchical network management. SNMPv3 strengthens security by adding the USM (User-based Security Model) and VACM (View-based Access Control Model).

SNMP provides a simple way to exchange network management information between two points in a network. SNMP uses a polling mechanism for queries and transmits messages over UDP (a connectionless transport-layer protocol), so it is well supported by existing networks.

SNMP uses a manager-agent model with two parts: the NMS (Network Management Station) and the Agent. The NMS is the workstation that runs the SNMP client program and is the core of SNMP network management. The Agent is the server software running on the managed devices. The NMS manages all managed objects through the Agents. The switch supports the Agent function.

The NMS and the Agent communicate in client/server mode by exchanging standard messages: the NMS sends requests and the Agent responds. There are seven types of SNMP messages:

- Get-Request
- Get-Response
- Get-Next-Request
- Get-Bulk-Request
- Set-Request
- Trap
- Inform-Request

The NMS sends queries to the Agent with Get-Request, Get-Next-Request, Get-Bulk-Request, and Set-Request messages, and the Agent replies with Get-Response messages. In special situations, such as a port going up or down or a topology change, the Agent sends Trap messages to the NMS to report the event. The NMS can also enable RMON to be alerted to abnormal events; when an alarm event is triggered, the Agent sends a Trap message or logs the event according to the settings. Inform-Request is mainly used for communication between NMSs in hierarchical network management.

USM ensures transfer security through encryption and authentication. USM encrypts messages with a key derived from the user password, so that messages cannot be read in transit, and USM authentication ensures that messages cannot be altered in transit. USM uses DES-CBC encryption, and HMAC-MD5 and HMAC-SHA for authentication.

VACM classifies user access permissions. It places users with the same access permissions in the same group, and users cannot perform operations they are not authorized for.

1.2.4.2 Introduction to MIB

The network management information accessed by the NMS is defined and organized in a Management Information Base (MIB). A MIB is a predefined, hierarchically structured set of information that can be accessed by network management protocols and obtained from monitored network devices. ISO ASN.1 defines a tree structure for the MIB, and each MIB organizes its information in this tree. Each node in the tree has an OID (Object Identifier) and a brief description. An OID is a sequence of integers separated by periods that identifies the node and locates it in the MIB tree, as shown in the following figure.

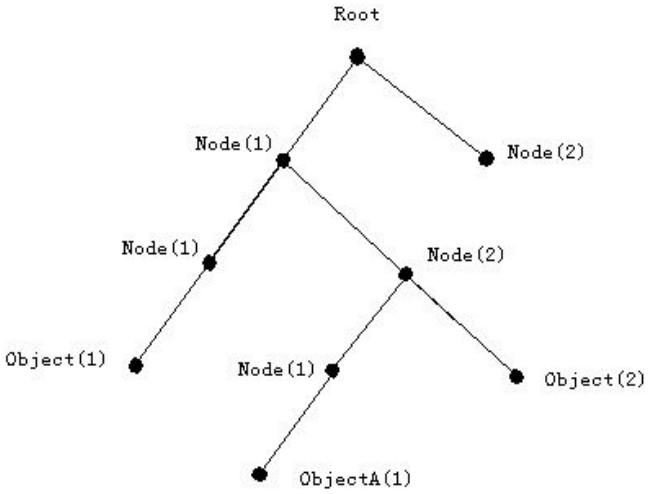


Figure 1-13 ASN.1 Tree Instance

In this figure, the OID of object A is 1.2.1.1. The NMS can locate this object by its unique OID and obtain its standard variables. A MIB defines a set of standard variables for monitored network devices according to this structure.

To browse the variables in the Agent MIB, run MIB browser software on the NMS. The Agent MIB usually consists of public MIBs and private MIBs. Public MIBs contain network management information that any NMS can access; private MIBs contain vendor-specific information that can be viewed and controlled with the vendor's support.

MIB-I [RFC1156] is the first public MIB implemented for SNMP and has been replaced by MIB-II [RFC1213]. MIB-II extends MIB-I and keeps the OIDs of the MIB-I tree. MIB-II contains subtrees called groups, whose objects cover all functional areas of network management. The NMS obtains network management information by accessing the MIB of the SNMP Agent.

The switch can act as an SNMP Agent and supports SNMPv1/v2c and SNMPv3. The switch supports the basic MIB-II, the RMON public MIB, and other public MIBs such as the BRIDGE MIB. The switch also supports private MIBs.

1.2.4.3 Introduction to RMON

RMON is the most important extension of standard SNMP. RMON is a set of MIB definitions that define standard network monitoring functions and interfaces for communication between SNMP management stations and remote monitors. RMON provides an efficient way to monitor activity within subnets.

The RMON MIB consists of 10 groups. The switch supports the most frequently used groups 1, 2, 3, and 9:

Statistics: Maintains basic usage and error statistics for each subnet monitored by the Agent.

History: Records periodic statistical samples from Statistics.

Alarm: Allows management console users to set sample intervals and alarm thresholds for any counter or integer recorded by the RMON Agent.

Event: A list of all events generated by the RMON Agent.

Alarm depends on Event. Statistics and History display current or historical subnet statistics. Alarm and Event provide a way to monitor changes in any integer data in the network and to raise alerts on abnormal events (by sending Traps or logging).

1.2.4.4 SNMP Configuration

1.2.4.4.1 SNMP Configuration Task List

1. Enable or disable the SNMP Agent
2. Configure the SNMP community string
3. Configure the IP address of the SNMP management station
4. Configure the engine ID
5. Configure users
6. Configure groups
7. Configure views
8. Configure Traps
9. Enable or disable RMON

1. Enable or disable the SNMP Agent

Command	Explanation
Global Mode	
snmp-server enabled **no snmp-server enabled**	Enables the SNMP Agent on the switch. The no command disables the SNMP Agent.

2. Configure the SNMP community string

Command	Explanation
Global Mode	
snmp-server community {ro | rw} {0 | 7} <*string*> [access {<*num-std*>|<*name*>}] [ipv6-access {<*ipv6-num-std*>|<*ipv6-name*>}] [read <*read-view-name*>] [write <*write-view-name*>] **no snmp-server community <*string*> [access {<*num-std*>|<*name*>}] [ipv6-access {<*ipv6-num-std*>|<*ipv6-name*>}]**	Configures a community string for the switch. The no command deletes the community string.

3. Configure the IP address of the SNMP management station

Command	Explanation
Global Mode	
snmp-server securityip { <*ipv4-address*> | <*ipv6-address*> } **no snmp-server securityip { <*ipv4-address*> | <*ipv6-address*> }**	Configures an IPv4/IPv6 secure address of an NMS that is allowed to access the switch. The no command deletes the secure address.
snmp-server securityip enable **snmp-server securityip disable**	Enables or disables secure IP address checking for NMS access.

4. Configure the engine ID

Command	Explanation
Global Mode	
snmp-server engineid <*engine-string*> **no snmp-server engineid**	Configures the local engine ID of the switch. This command is used for SNMPv3.

5. Configure users

Command	Explanation
Global Mode	
snmp-server user <*use-string*> <*group-string*> [{authPriv | authNoPriv} auth {md5 | sha} <*word*>] [access {<*num-std*>| <*name*>}] [ipv6-access {<*ipv6-num-std*>| <*ipv6-name*>}] **no snmp-server user <*user-string*> [access {<*num-std*>|<*name*>}] [ipv6-access {<*ipv6-num-std*>|<*ipv6-name*>}]**	Adds a user to an SNMP group. This command is used to configure USM for SNMPv3.

6. Configure groups

Command	Explanation
Global Mode	
snmp-server group <*group-string*> {noauthnopriv|authnopriv|authpriv} [[read <*read-string*>] [write <*write-string*>] [notify <*notify-string*>]] [access {<*num-std*>|<*name*>}] [ipv6-access {<*ipv6-num-std*>|<*ipv6-name*>}] **no snmp-server group <*group-string*> {noauthnopriv|authnopriv|authpriv} [access {<*num-std*>|<*name*>}] [ipv6-access {<*ipv6-num-std*>|<*ipv6-name*>}]**	Configures group information on the switch. This command is used to configure VACM for SNMPv3.

7. Configure views

Command	Explanation
Global Mode	
snmp-server view <*view-string*> <*oid-string*> {include|exclude} **no snmp-server view <*view-string*> [<*oid-string*>]**	Configures a view on the switch. This command is used for SNMPv3.

8. Configure Traps

Command	Explanation
Global Mode	
snmp-server enable traps **no snmp-server enable traps**	Enables the switch to send Trap messages. This command is used for SNMPv1/v2c/v3.
snmp-server host { <host-ipv4-address> | <host-ipv6-address> } {v1 | v2c | {v3 {noauthnopriv | authnopriv | authpriv}}} <*user-string*> **no snmp-server host { <host-ipv4-address> | <host-ipv6-address> } {v1 | v2c | {v3 {noauthnopriv | authnopriv | authpriv}}} <*user-string*>**	Configures the IPv4/IPv6 address of a host that receives SNMP Traps. For SNMPv1/v2c, this command also configures the Trap community string; for SNMPv3, it also configures the Trap username and security level. The no command deletes the IPv4 or IPv6 address.
snmp-server trap-source {<ipv4-address> | <ipv6-address>} **no snmp-server trap-source {<ipv4-address> | <ipv6-address>}**	Configures the source IPv4 or IPv6 address of Trap packets. The no command deletes the configuration.
Port Mode	
[no] switchport updown notification enable	Enables or disables sending Trap messages for port up/down events.

9. Enable or disable RMON

Command	Explanation
Global Mode	

rmon enable **no rmon enable**	Enables or disables RMON.

1.2.4.5 Typical SNMP Configuration Examples

The IP address of the NMS is 1.1.1.5; the IP address of the switch (Agent) is 1.1.1.9.

Scenario 1: The NMS management software uses SNMP to obtain data from the switch.

The switch configuration is as follows:

```
Switch(config)#snmp-server enable
Switch(config)#snmp-server community rw private
Switch(config)#snmp-server community ro public
Switch(config)#snmp-server securityip 1.1.1.5
```

The NMS can use the community string "private" to access the switch with read-write permission, or "public" to access it with read-only permission.

Scenario 2: The NMS receives Trap messages from the switch. (The NMS may verify the community string of Trap messages; in this scenario, the NMS uses the Trap community string "usertrap".)

The switch configuration is as follows:

```
Switch(config)#snmp-server enable
Switch(config)#snmp-server host 1.1.1.5 v1 usertrap
Switch(config)#snmp-server enable traps
```

Scenario 3: The NMS uses SNMPv3 to obtain information from the switch.

The switch configuration is as follows:

```
Switch(config)#snmp-server
Switch(config)#snmp-server user tester UserGroup authPriv auth md5 hellotst
Switch(config)#snmp-server group UserGroup AuthPriv read max write max notify
max
Switch(config)#snmp-server view max 1 include
```

Scenario 4: The NMS receives SNMPv3 Trap messages from the switch.

The switch configuration is as follows:

```
Switch(config)#snmp-server enable
Switch(config)#snmp-server host 10.1.1.2 v3 authpriv tester
Switch(config)#snmp-server enable traps
```

Scenario 5: The IPv6 address of the NMS is 2004:1:2:3::2, and the IPv6 address of the switch (Agent) is 2004:1:2:3::1. The NMS management software uses SNMP to obtain data from the switch.

The switch configuration is as follows:

```
Switch(config)#snmp-server enable
Switch(config)#snmp-server community rw private
Switch(config)#snmp-server community ro public
Switch(config)#snmp-server securityip 2004:1:2:3::2
```

The NMS can use the community string "private" to access the switch with read-write permission, or "public" to access it with read-only permission.

Scenario 6: The NMS receives Trap messages from the switch. (The NMS may verify the community string of Trap messages; in this scenario, the NMS uses the Trap community string "usertrap".)

The switch configuration is as follows:

```
Switch(config)#snmp-server host 2004:1:2:3::2 v1 usertrap
Switch(config)#snmp-server enable traps
```

1.2.4.6 SNMP Troubleshooting

If SNMP does not work properly because of physical connection failures, incorrect configuration, or other causes, troubleshoot as follows:

- ☞ Make sure the physical connection is good.
- ☞ The interface and the data link layer protocol are up (use the "show interface" command), and the connection between the switch and the host is verified by ping (use the "ping" command).
- ☞ The SNMP agent server function is enabled on the switch (use the "snmp-server" command).
- ☞ The secure IP address of the NMS (use the "snmp-server securityip" command) and the community string (use the "snmp-server community" command) are configured correctly. If either is incorrect, SNMP cannot communicate with the NMS properly.
- ☞ If the Trap function is required, enable Trap (use the "snmp-server enable traps" command) and configure the target host IP address and community string for Trap correctly (use the "snmp-server host" command) so that Trap messages can be sent to the specified host.
- ☞ If the RMON function is required, enable RMON first (use the "rmon enable" command).
- ☞ Use the "show snmp" command to verify sent and received SNMP messages, the "show snmp status" command to verify the SNMP configuration, and the "debug snmp packet" command to enable SNMP debugging and check the debug information.

If the problem persists, contact EnGenius technical support.

1.2.5 Switch Upgrade

The switch can be upgraded in two ways: BootROM upgrade and TFTP/FTP upgrade in Shell.

1.2.5.1 Switch System Files

The system files include the system image file and the boot file. Upgrading the switch means updating these two files by overwriting the old files with new ones.

The system image file is the compressed file containing the switch hardware drivers, software support programs, and so on, commonly called the IMG update file. On this device, the system image file is saved in the MMC storage medium, typically with the file name "nos.img".

The boot file initializes the switch and is commonly called the ROM update file (a large boot file can be compressed into an IMG file). On the switch, the boot file can be saved only in ROM, and its name must be boot.rom.

The system image file and the boot file are updated in the same way. The switch provides two update modes: 1. BootROM mode; 2. TFTP and FTP update in Shell mode. These two update methods are described in the following two sections.

1.2.5.2 BootROM Upgrade

BootROM upgrade supports one method: TFTP, which is configured with BootROM commands.

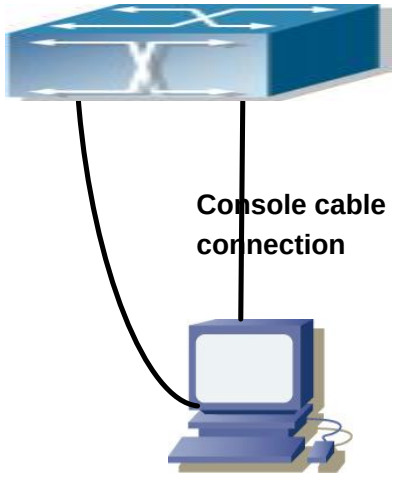


Figure 1-14 Typical topology for switch upgrade in BootROM mode

The upgrade procedure is as follows:

Step 1:

As shown in the figure, a PC is used as the console for the switch. A console cable connects the PC to the management port of the switch. The PC must have TFTP server software installed and must hold the boot file required for the upgrade.

Step 2:

Press Ctrl+B while the switch boots until the switch enters BootROM monitor mode. The result is shown below:

```
[Boot]:
```

Step 3:

In BootROM mode, run "setconfig" to set the IP address and mask of the switch in BootROM mode and the server IP address and mask. Assuming the switch address is 192.168.1.2, the PC address is 192.168.1.66, and TFTP upgrade is selected, the configuration is as follows:

```
[Boot]: setconfig
Host IP Address: [10.1.1.1] 192.168.1.2
Server IP Address: [10.1.1.2] 192.168.1.66
[Boot]:
```

Step 4:

Start the TFTP server program on the PC. Before downloading the upgrade file to the switch, ping the server from the switch to verify connectivity. If the ping succeeds, run the "load" command in BootROM mode on the switch; if it fails, troubleshoot to find the cause.

The following example downloads the update file boot.rom.

```
[Boot]: load boot.rom
TFTP from server 192.168.1.66; our IP address is 192.168.1.2
Filename 'boot.rom'.
Load address: 0x300000
Loading: ##########################################################################
   ################################
done
Bytes transferred = 496240 (79270 hex)
[Boot]:
```

Step 5:

Run **write boot.rom** in BootROM mode to save the update file, as shown below.

```
[Boot]: write boot.rom
File exists, overwrite? (Y/N)[N] y
Writing flash:/boot.rom......
```

```
Write flash:/boot.rom OK.
[Boot]:
```

Step 6:

After the upgrade succeeds, run the run or reboot command in BootROM mode to return to the CLI configuration interface.

```
[Boot]: run (or reboot)
```

Other commands in BootROM mode

1. dir command

Lists the files in the MMC.

```
[Boot]: dir mmc:/
  5399893   nos.img
[Boot]:
```

1.2.5.3 FTP/TFTP Upgrade

1.2.5.3.1 Introduction to FTP/TFTP

FTP (File Transfer Protocol) and TFTP (Trivial File Transfer Protocol) are file transfer protocols at the application layer (the fourth layer) of the TCP/IP protocol stack. They transfer files between hosts, and between hosts and switches, in a client-server model. Their differences are as follows.

FTP runs over TCP and provides a reliable, connection-oriented data stream transfer service. However, it does not provide file access authorization and uses a simple authentication mechanism (the username and password are transferred in plain text). An FTP file transfer requires two connections between the client and the server: a management connection and a data connection. The FTP client sends a transfer request to establish the management connection to port 21 on the server and negotiates the data connection through the management connection.

There are two types of data connections: active and passive.

In an active connection, the client sends the server the address and port number it will use for data transmission, and the management connection is maintained until the data transfer is complete. The server then establishes the data connection from port 20 (if it is free) to the address and port provided by the client; if port 20 is in use, the server automatically selects another port number for the data connection.

In a passive connection, the client uses the management connection to notify the server to set up a passive connection. The server creates its own data listening port and informs the client of it, and the client establishes the data connection to that port.

Because the data connection is established through a specified address and port, a third party can provide the data connection service.

TFTP runs over UDP and provides an unreliable data stream transfer service without user authentication or permission-based file access authorization. It ensures correct data transmission by sending and acknowledging packets and retransmitting packets after a timeout. The advantage of TFTP over FTP is that it is a simple, low-overhead file transfer service.

The switch can operate as an FTP/TFTP client or server. As an FTP/TFTP client, the switch can download configuration files or system files from remote FTP/TFTP servers (hosts or other switches) without affecting its normal operation, and as an FTP client it can also retrieve the file list from the server. The switch can also upload its current configuration files or system files to remote FTP/TFTP servers (hosts or other switches). As an FTP/TFTP server, the switch provides file upload and download services to authorized FTP/TFTP clients, and as an FTP server it also provides a file list service.

The following terms are frequently used with FTP/TFTP.

ROM: Short for EPROM (erasable programmable read-only memory). On the switch, EPROM is replaced by FLASH memory.

SDRAM: RAM in the switch, used for running the system software and storing the configuration sequence.

MMC: Stores system image files (for example, nos.img), supports high-capacity read/write operations, and participates directly in device boot-up and runtime operation. It can also store configuration files.

FLASH: Stores small amounts of data such as configuration files (for example, startup.cfg) and logs, and provides power-loss protection so that data persists through unexpected shutdowns.

System file: Includes the system image file and the boot file.

System image file: The compressed file containing the switch hardware drivers and software support programs, usually called the IMAGE upgrade file. The system image file of the switch is usually stored in the MMC. When the system image file is transferred through FTP in Global Mode, its file name is usually nos.img; other IMAGE system files are rejected. **Boot file:** The file that initializes the switch, also called the ROM upgrade file (a large file can be compressed into an IMAGE file). On the switch, the boot file can be saved only in ROM, and its name must be boot.rom.

Configuration file: Includes the startup configuration file and the running configuration file. Keeping these two files separate makes it easier to back up and update configurations.

Startup configuration file: The configuration sequence used when the switch starts. The startup configuration file is stored in nonvolatile storage, which corresponds to saving the configuration. If the device does not support CF, the configuration file is stored only in FLASH or MMC; if the device supports CF, it is stored in FLASH, MMC, or CF. If the device supports multiple configuration files, the configuration file is named as a .cfg file, and the default is startup.cfg. If the device does not support multiple configuration files, the startup configuration file must be named startup-config.

Running configuration file: The configuration sequence currently running on the switch, stored in RAM. In the current version, the running configuration sequence running-config can be saved from RAM to FLASH with the **write** command or the **copy running-config startup-config** command, so that it becomes the startup configuration file; this is called saving the configuration. To prevent unauthorized file uploads and simplify configuration, the switch requires the running configuration file to be named running-config.

Factory configuration file: The configuration file shipped with the switch, named factory-config. Run **set default** and **write**, and then restart the switch; the factory configuration file is loaded and overwrites the current startup configuration file.

1.2.5.3.2 FTP/TFTP Configuration

The switch is configured almost identically as an FTP client and as a TFTP client, so this manual describes the FTP and TFTP configuration procedures together.

1.2.5.3.2.1 FTP/TFTP Configuration Task List

1. Configure the FTP/TFTP client
    （1） Upload/download configuration files or system files.
    （2） View the server file list (FTP client only).
2. Configure the FTP server
    （1） Start the FTP server
    （2） Configure the FTP login username and password
    （3） Modify the FTP server connection idle time
    （4） Shut down the FTP server
3. Configure the TFTP server

（1） Start the TFTP server

（2） Configure the TFTP server connection idle time

（3） Configure the number of retransmissions before timeout for unacknowledged packets

（4） Shut down the TFTP server

1. Configure the FTP/TFTP client

(1) Upload/download files as an FTP/TFTP client

Command	Explanation
Admin Mode	
copy <*source-url*> <*destination-url*> [ascii | binary]	Uploads or downloads a file as an FTP/TFTP client.

(2) View the server file list as an FTP client

Admin Mode	
ftp-dir <*ftpServerUrl*>	Displays the file list on the server (FTP client). The format of ftpServerUrl is: ftp: //user: password@IPv4|IPv6 Address.

2. Configure the FTP server

(1) Start the FTP server

Command	Explanation
Global Mode	
ftp-server enable **no ftp-server enable**	Enables the FTP server. The no command shuts down the FTP server and prevents FTP users from logging in.

(2) Configure the FTP login username and password

Command	Explanation
Global Mode	
ip ftp username <*username*> password [0 | 7] <*password*> **no ip ftp username<*username*>**	Configures the FTP login username and password. The no command deletes the username and password.

(3) Modify the FTP server connection idle time

Command	Explanation
Global Mode	
ftp-server timeout <*seconds*>	Sets the connection idle time.

3. Configure the TFTP server

(1) Start the TFTP server

Command	Explanation
Global Mode	
tftp-server enable **no tftp-server enable**	Enables the TFTP server. The no command shuts down the TFTP server and prevents TFTP users from logging in.

(2) Modify the TFTP server connection idle time

Command	Explanation
Global Mode	
tftp-server retransmission-timeout <*seconds*>	Sets the maximum retransmission time within the timeout interval.

(3) Modify the number of TFTP server retransmissions

Command	Explanation

Global Mode	
tftp-server retransmission-number *<number>*	Sets the number of retransmissions for the TFTP server.

1.2.5.3.3 FTP/TFTP Configuration Examples

The configuration is the same for IPv4 and IPv6 addresses. The examples use IPv4 addresses only.

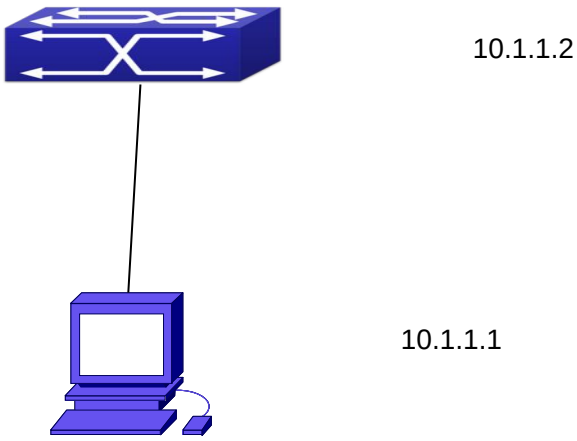


Figure 1-175 Downloading the nos.img file as an FTP/TFTP client

Scenario 1: The switch is used as an FTP/TFTP client. One of its ports connects to a computer that acts as an FTP/TFTP server with IP address 10.1.1.1. The IP address of the switch management VLAN is 10.1.1.2. Download the "nos.img" file from the computer to the switch.

☞ FTP configuration

Computer side configuration:

Start the FTP server software on the computer and set the username "Switch" and the password "superuser". Place the "12_30_nos.img" file in the appropriate FTP server directory on the computer.

Configure the switch as follows:

```
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ip address 10.1.1.2 255.255.255.0
Switch(Config-if-Vlan1)#no shut
Switch(Config-if-Vlan1)#exit
Switch(config)#exit
Switch#copy ftp: //Switch:switch@10.1.1.1/12_30_nos.img mmc:/nos.img
```

With the above commands, the switch downloads the "nos.img" file from the computer to the MMC.

☞ TFTP configuration

Computer side configuration:

Start the TFTP server software on the computer and place the "12_30_nos.img" file in the appropriate TFTP server directory on the computer.

Configure the switch as follows:

```
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ip address 10.1.1.2 255.255.255.0
Switch(Config-if-Vlan1)#no shut
Switch(Config-if-Vlan1)#exit
Switch(config)#exit
Switch#copy tftp: //10.1.1.1/12_30_nos.img mmc:/nos.img
```

Scenario 2: The switch is used as an FTP server. One of its ports connects to a computer that acts as an FTP client. Transfer the "nos.img" file from the switch to the computer and save it as 12_25_nos.img.

Configure the switch as follows:

```
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ip address 10.1.1.2 255.255.255.0
Switch(Config-if-Vlan1)#no shut
Switch(Config-if-Vlan1)#exit
Switch(config)#ftp-server enable
Switch(config)# username Admin password 0 superuser
```

Computer side configuration:

Log in to the switch with any FTP client software, using the username "Switch" and the password "superuser", and use the command "get mmc:/nos.img 12_25_nos.img" to download the "nos.img" file from the switch to the computer.

Scenario 3: The switch is used as a TFTP server. One of its ports connects to a computer that acts as a TFTP client. Transfer the "nos.img" file from the switch to the computer.

Configure the switch as follows:

```
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ip address 10.1.1.2 255.255.255.0
Switch(Config-if-Vlan1)#no shut
Switch(Config-if-Vlan1)#exit
Switch(config)#tftp-server enable
```

Computer side configuration:

Log in to the switch with any TFTP client software and use the "tftp" command to download the "nos.img" file from the switch to the computer.

Scenario 4: The switch acts as an FTP client to view the file list on the FTP server. Conditions: the switch connects through an Ethernet port to a computer that acts as an FTP server with IP address 10.1.1.1; the switch acts as an FTP client, and the IP address of its management VLAN1 interface is 10.1.1.2.

FTP configuration:

PC side:

Start the FTP server software on the PC and set the username "Switch" and the password "superuser".

```
Switch:
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ip address 10.1.1.2 255.255.255.0
Switch(Config-if-Vlan1)#no shut
Switch(Config-if-Vlan1)#exit
Switch#copy ftp: //Switch: superuser@10.1.1.1
220 Serv-U FTP-Server v2.5 build 6 for WinSock ready...
331 User name okay, need password.
230 User logged in, proceed.
200 PORT Command successful.
150 Opening ASCII mode data connection for /bin/ls.
recv total = 480
nos.img
nos.rom
parsecommandline.cpp
position.doc
qmdict.zip
…(some display omitted here)
show.txt
snmp.TXT
226 Transfer complete.
```

1.2.5.3.4 FTP/TFTP Troubleshooting

1.2.5.3.4.1 FTP Troubleshooting

When uploading or downloading system files with FTP, make sure the link is connected: use the "Ping" command to verify connectivity between the FTP client and the server before running the FTP program. If the ping fails, troubleshoot the link to restore connectivity.

☞ The following messages are displayed when files are sent successfully. Otherwise, verify link connectivity and run the "copy" command again.

```
220 Serv-U FTP-Server v2.5 build 6 for WinSock ready...
331 User name okay, need password.
230 User logged in, proceed.
200 PORT Command successful.
nos.img file length = 1526021
read file ok
send file
150 Opening ASCII mode data connection for nos.img.
226 Transfer complete.
close ftp client.
```

☞ The following messages are displayed when files are received successfully. Otherwise, verify link connectivity and run the "copy" command again.

```
220 Serv-U FTP-Server v2.5 build 6 for WinSock ready...
331 User name okay, need password.
230 User logged in, proceed.
200 PORT Command successful.
recv total = 1526037
**********************
write ok
150 Opening ASCII mode data connection for nos.img (1526037 bytes).
226 Transfer complete.
```

When the switch is upgrading a system file or system startup file through FTP, do not restart the switch until "close ftp client" or "226 Transfer complete." is displayed, which indicates that the upgrade succeeded; otherwise, the switch may fail to start. If the upgrade of the system file or system startup file through FTP fails, try again or upgrade in BootROM mode.

1.2.5.3.4.2 TFTP Troubleshooting

When uploading or downloading system files with TFTP, make sure the link is connected: use the "**Ping**" command to verify connectivity between the TFTP client and the server before running the TFTP program. If the ping fails, troubleshoot the link to restore connectivity.

☞ The following messages are displayed when files are sent successfully. Otherwise, verify link connectivity and run the "copy" command again.

```
nos.img file length = 1526021
read file ok
begin to send file, wait...
file transfers complete.
Close tftp client.
```

☞ The following messages are displayed when files are received successfully. Otherwise, verify link connectivity and run the "copy" command again.

```
begin to receive file, wait...
recv 1526037
**********************
write ok
transfer complete
close tftp client.
```

When the switch is upgrading a system file or system startup file through TFTP, do not restart the switch until "close tftp client" is displayed, which indicates that the upgrade succeeded; otherwise, the switch may fail to start. If the upgrade of the system file or system startup file through TFTP fails, try again or upgrade in BootROM mode.

1.3 File System

1.3.1 Introduction to File Storage Devices

The primary file storage medium of the switch is the MMC, which is commonly used to store system image files (IMG files). FLASH is used to store system boot files (ROM files) and configuration files (CFG files).

Files in the MMC can be copied, deleted, or renamed in Shell or BootROM mode.

1.3.2 File System Operation Configuration Task List

1. Create a subdirectory
2. Delete a subdirectory
3. Change the current working directory of the storage device
4. Display the current working directory
5. Display information about a specified file or directory
6. Delete a specified file from the file system
7. Rename a file
8. Copy a file

1. Create a subdirectory

Command	Explanation
Admin Mode	
mkdir <*directory*>	Creates a subdirectory in a specified directory on a device.

2. Delete a subdirectory

Command	Explanation
Admin Mode	
rmdir <*directory*>	Deletes a subdirectory in a specified directory on a device.

3. Change the current working directory of the storage device

Command	Explanation
Admin Mode	
cd <*directory*>	Changes the current working directory of the storage device.

4. Display the current working directory

Command	Explanation
Admin Mode	
pwd	Displays the current working directory.

6. Display information about a specified file or directory

Command	Explanation
Admin Mode	
dir [WORD|all]	Displays information about a specified file or

	directory on the storage device.

7. Delete a specified file from the file system

Command	Explanation
Admin Mode	
delete <*file-url*>	Deletes the specified file from the file system.

8. Rename a file

Command	Explanation
Admin Mode	
rename <*source-file-url*> <*dest-file*>	Renames a specified file on the switch.

9. Copy a file

Command	Explanation
Admin Mode	
copy <*source-file-url* > <*dest-file-url*>	Copies a specified file on the switch and saves it as a new file.

1.3.3 Typical Applications

Copy the IMG file mmc:/nos.img stored in the MMC to mmc:/nos-6.1.11.0.img.

Configure the switch as follows:

```
Switch#copy mmc:/nos.img mmc:/nos-6.1.11.0.img
Copy mmc:/nos.img to mmc:/nos-6.1.11.0.img? [Y:N] y
Copyed file mmc:/nos.img to mmc:/nos-6.1.11.0.img.
```

Copy the CFG file flash:/startup.cfg stored in FLASH to mmc:/startup.cfg.

Configure the switch as follows:

```
Switch#copy flash:/startup.cfg mmc:/startup.cfg
Write ok.
```

1.3.4 Troubleshooting

If errors occur during file system operations, check whether they are caused by the following:

- ☞ The file name or path is entered incorrectly.
- ☞ When renaming a file, the file is in use, or the new file name is already used by an existing file or directory.

1.4 USB

1.4.1 Introduction

When a USB device is inserted or removed, the switch detects the hot insertion or removal and mounts or unmounts the USB device.

When a USB device is inserted, the switch mounts the USB file system. Files on the USB device can be read, copied, deleted, and renamed, and the device can also be used to restore the configuration, download files, and save files.

The device supports USB overcurrent warning. When the actual current exceeds the rated current of the device, the switch notifies the user that the temperature is too high; inserting the USB device in this state risks burning out the device.

1.4.2 USB Function Configuration Task List

1. Mount the USB device and enter the USB drive
2. Display USB drive information
3. Copy a source file to a destination file
4. Delete a file
5. Rename a file
6. Update the configuration file from the USB drive to the switch
7. Update the BootROM file from the USB drive to the switch
8. Update the IMG file from the USB drive to the switch
9. Create a directory
10. Delete an existing directory
11. Unmount the USB device

1. Mount the USB device and enter the USB drive

Command	Explanation
Admin Mode	
cd usb:	Enters the USB drive.

2. Display USB drive information

Command	Explanation
Admin Mode	
dir	Displays information about the USB drive.

3. Copy a source file to a destination file

Command	Explanation
Admin Mode	
copy source destination	Copies the source file to the destination file.

4. Delete a file

Command	Explanation
Admin Mode	
delete filename	Deletes the file.

5. Rename a file

Command	Explanation
Admin Mode	
rename source destitation	Renames the source file to the destination file name.

6. Update the configuration file from the USB drive to the switch

Command	Explanation
Admin Mode	
copy usb:/startup.cfg startup.cfg	Updates the configuration file from the USB drive to the switch. The reverse direction is also supported: copy startup.cfg usb:/startup.cfg

7. Update the BootROM file from the USB drive to the switch

Command	Explanation

Admin Mode	
copy usb:/boot.rom boot.rom	Updates the BootROM file from the USB drive to the switch. The reverse direction is also supported: copy boot.rom usb:/boot.rom

8. Update the IMG file from the USB drive to the switch

Command	Explanation
Admin Mode	
copy usb:/nos.img nos.img	Updates the IMG file from the USB drive to the switch. The reverse direction is also supported: copy nos.img usb:/nos.img

9. Create a directory

Command	Explanation
Admin Mode	
mkdir	Creates a directory.

10. Delete an existing directory

Command	Explanation
Admin Mode	
rmdir	Deletes an existing directory.

1.4.3 USB Function Examples

Delete source1.txt from the USB drive, and rename tt.txt on the USB drive to tttt.txt.

Create the directory sw1 on the USB drive.

```
Switch#delete source1.txt
Switch#rename usb:/tt.txt usb:/tttt.txt
Switch#mkdir sw1
```

1.4.4 USB Function Troubleshooting

- Currently, the USB device can be mounted and unmounted only in CLI user operation mode. This command is not supported in non-CLI user operation modes.
- Make sure the switch is powered on and the USB device is inserted correctly. The files on the USB device are mounted to the switch file system automatically.
- The dir command entered directly displays the files on the flash drive by default. To display the files on the USB drive, enter **cd usb:** to enter the USB drive, and then enter **dir** to display the file information. Alternatively, use the absolute path and enter **dir usb:** to display the files.

1.5 Device Management

1.5.1 Device Management Overview

The device management function of the switch provides information about line card status, line card operation debugging, and power supply and fan status. It supports maintenance and management of the physical devices, restart of the switch and line cards, and hot swapping of cards. The switch supports dual-master mode: if two master management cards are present in the system, the master control board in the lower-numbered slot becomes the Active Master and the other becomes the Standby Master.

1.5.2 Device Management Configuration

1.5.2.1 Monitoring and Debugging Tasks

1. Display chip information
2. Display fan status
3. Display power supply status

1. Display chip information

Command	Explanation
Admin Mode	
show [member <member-id>] slot <slot-id>	Displays basic information about each chip.

2. Display fan status

Command	Explanation
Admin Mode	
show fan	Displays whether the fan tray is in place, its running status, and the fan speed.

3. Display power supply status

Command	Explanation
Admin Mode	
show power	Displays whether the power supply is in place and its running status.

Chapter 2 Network Security Operations Manual

2.1 Introduction to Network Security Features

Network security is the ability to protect a network from attacks, intrusions, interference, damage, unauthorized use, and accidents by taking necessary measures, so that the network operates stably and reliably and the integrity, confidentiality, availability, authenticity, and controllability of network data are ensured. It covers all aspects of computer networks, including hardware, software, data, and the overall security of the system. Network security has become an important part of national security and is directly related to the stability and development of all fields, including politics, the economy, culture, society, ecology, and defense.
This chapter focuses on securing passwords through encryption and password complexity, and on ensuring the integrity and confidentiality of data.

2.2 Network Security Feature Configuration

2.2.1 Port Number Modification Configuration Task Sequence

1. Modify the default SSH service port on the server
2. Modify the default Telnet service port on the server
3. Modify the default SNMP service port on the server

Command	Explanation
Global Mode	
[no] ssh-server dst-port <*port-number*>	Modifies the default port number of the SSH server. The no command disables this function.
[no] telnet-server dst-port <*port-number*>	Modifies the default port number of the Telnet server. The no command disables this function.
[no] snmp-server dst-port <port-number>	Modifies the default port number of the SNMP server. The no command disables this function.

2.2.2 Encryption Algorithm Modification Configuration Task Sequence

1. Modify the SSL encryption algorithm
2. Enable IPv4 fragment checking

Command	Explanation
Global Mode	
[no] dosattack-check tcp-flags enable	Enables TCP flag checking. The no command disables it.
[no] dosattack-check ipv4-first-fragment enable	Enables IPv4 fragment checking; the no command disables it. This command has no effect when used alone. If this function is not enabled, the switch does not drop IPv4 fragment packets that contain illegal TCP flags.

2.2.3 Password Security Function Configuration Task Sequence

1．Modify the default password

2．Force a change of the default password

3．Check password strength

4．Configure the password validity period

5．Display error message prompts

6．Configure password echo

7．Force secure password input

Command	Explanation
Global Mode	
[no] dosattack-check srcport-equal-dstport enable	Enables port spoofing protection. The no command disables it.
[no] dosattack-check ipv4-first-fragment enable	Enables IPv4 fragment checking; the no command disables it. This command has no effect when used alone. If this function is not enabled, the switch does not drop IPv4 fragment packets whose source port equals the destination port.

2.2.4 Sensitive Information Encryption Function Configuration Task Sequence

1．Enable the encryption function

2．Configure the encryption mode

Command	Explanation
Global Mode	
[no] dosattack-check srcport-equal-dstport enable	Enables port spoofing protection. The no command disables it.
[no] dosattack-check ipv4-first-fragment enable	Enables IPv4 fragment checking; the no command disables it. This command has no effect when used alone. If this function is not enabled, the switch does not drop IPv4 fragment packets whose source port equals the destination port.

2.2.5 Important File Verification Function Configuration Task Sequence

1. Enable IMG and configuration file verification

Command	Explanation
Global Mode	
[no] boot (img | startup-config) check enable	Enables IMG file verification or configuration file verification. The no command disables it.

Network Security Feature Examples

Example 1:

Configuration requirements: The switch checks the configuration file at startup, a login password is required, and the password validity period is 10 days.

Configuration steps:

```
Switch(config)#
Switch(config)#authentication line console login local
Switch(config)#dosattack-check ipv4-first-fragment enable
Switch(config)#service user password valid-time 10
```

Example 2:

Configuration requirements: 1. Passwords can be entered only in ciphertext, displayed as asterisks (*), and must contain at least 10 characters, including digits, uppercase and lowercase letters, and special characters. 2. Passwords in the configuration are encrypted with SM4 and are displayed differently each time.

Configuration steps:

```
Switch(config)#
Switch(config)#service password-encryption
Switch(config)#service password-encryption type user algo sm4 salt
Error:user admin is Encrypt by other algo, please delete first
Switch(config)#no username admin
Switch(config)#service password-encryption type user algo sm4 salt
Switch(config)#userpassword security-config
Switch(config)#password feedback star
Switch(config)#userpassword restriction min-length 10 format-mix 4
Switch(config)#username aaa password
Password:***********
Invalid password, password must no lease than 4 type(ex:0-9,A-Z,a-z,special
character) mixed format!
Switch(config)#username aaa password
Password:*************
Switch(config)#show running-config |include aaa
username aaa password 7 Zjs+r7VvTdI4oHwSJXHdOn3pim5RGKcI5ZOPFUXchjU=
Switch(config)#show running-config |include aaa
username aaa password 7 Zjs+r7VvTdI4oHwSJXHdOiwj4bAHIoyEj9vRRNVh+Sc=
```

Example 3:

Configuration requirements: When a user connects to the device through SSH, the 3des-cbc and 3des-ctr algorithms cannot be used.

Configuration steps:

```
Switch(config)#
Switch(config)#ssh-server enable
ssh is enabled successfully.
Switch(config)#ssh-server encryption-algorithm add all
Switch(config)#ssh-server encryption-algorithm remove 3des-cbc 3des-ctr
```

Chapter 3 Layer 2 Services Configuration

3.1 Port Configuration

3.1.1 Introduction to Port

To configure network ports, use the interface ethernet <interface-list> command to enter Port Mode for the corresponding Ethernet ports, where <interface-list> specifies one or more ports. If <interface-list> contains multiple ports, separate them with ';' or '-': ';' separates discrete port numbers and '-' specifies a range of consecutive port numbers. For example, to perform an operation on ports 2, 3, 4, and 5, use: interface ethernet 1/0/2-5. Port speed, duplex mode, and flow control can be configured in Port Mode, and the performance of the corresponding ports changes accordingly.

3.1.2 Network Port Configuration Task List

1. Enter Port Mode
2. Configure the properties of the network ports

(2) Enable or disable ports

(3) Configure port names

(4) Configure port cable types

(5) Configure port speed and duplex mode

(6) Configure bandwidth control

(7) Configure flow control

(8) Enable or disable the port loopback function

(9) Configure the broadcast storm control function of the switch

(10) Configure the port scan mode

(11) Configure rate-violation control of the port

(12) Configure the port-rate-statistics interval

(13) Configure the port to discard tagged or untagged packets

1. Enter Port Mode

Command	Explanation
Global Mode	
interface ethernet <*interface-list*>	Enters Port Mode for the specified Ethernet ports.

2. Configure the properties of the Ethernet ports

Command	Explanation
Port Mode	
media-type {copper | fiber}	Sets the combo port mode (combo ports only).
shutdown **no shutdown**	Disables/enables the specified ports.
description <string> **no description**	Assigns a name to the specified ports or removes it.

speed-duplex {auto [10 [100 [1000]] [auto | full | half |]] | force10-half | force10-full | force100-half | force100-full | force100-fx [module-type {auto-detected | no-phy-integrated | phy-integrated}] | {{force1g-half | force1g-full} [nonegotiate [master | slave]]}| force10g-full} **no speed-duplex**	Sets the port speed and duplex mode of 100/1000Base-TX or 100Base-FX ports. The no command restores the default setting (speed and duplex mode are negotiated automatically).
negotiation {on|off}	Enables/disables auto-negotiation on 1000Base-FX ports.
bandwidth control <*bandwidth*> [both | receive | transmit] **no bandwidth control**	Sets or cancels the bandwidth limit for incoming/outgoing traffic on the specified ports.
flow control **no flow control**	Enables/disables flow control on the specified ports.
loopback **no loopback**	Enables/disables the loopback test function on the specified ports.
storm-control {unicast | broadcast | multicast} <packets>	Enables storm control for broadcast, multicast, and unknown-destination unicast traffic (collectively referred to as broadcast), and sets the allowed number of packets. The no command disables storm control.
port-scan-mode {interrupt | poll} **no port-scan-mode**	Configures the port scan mode as interrupt or poll. The no command restores the default port scan mode.
rate-violation <200-2000000> [recovery <0-86400>] **no rate-violation**	Sets the maximum packet reception rate of a port. If a port's received packet rate exceeds this limit, the port is shut down and the configured recovery time (default 300s) applies before it recovers. The no command disables the rate-violation function on a port.
switchport discard packet { tag | untag } **no switchport discard packet { tag | untag }**	Configures the port to discard tagged or untagged packets. The **no** command cancels this restriction, allowing the port to receive tagged or untagged packets again.
Global Mode	
port-rate-statistics interval <interval - value>	Configures the port-rate-statistics interval.

3.1.3 Port Configuration Example

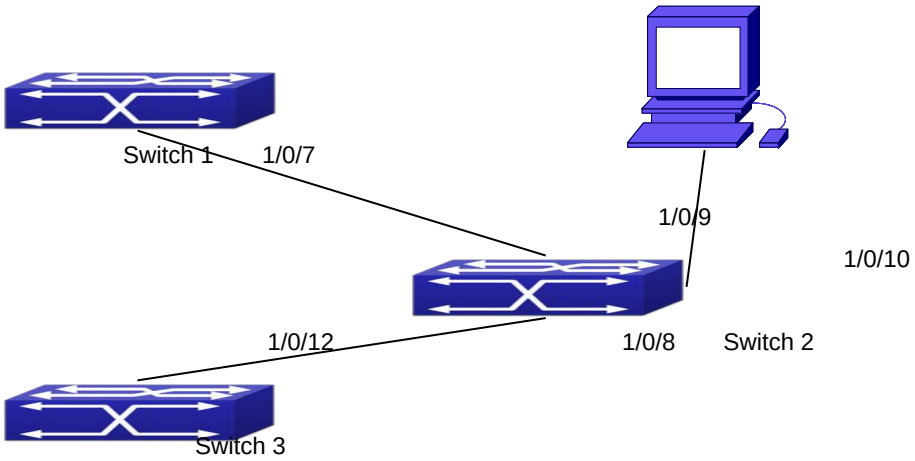


Figure 3-1 Port Configuration Example

No VLAN is configured on the switches; the default VLAN 1 is used.

Switch	Port	Property
Switch1	1/0/7	Ingress bandwidth limit: 50 M
Switch2	1/0/8	Mirror source port
	1/0/9	100Mbps full, mirror source port
	1/0/10	1000Mbps full, mirror destination port
Switch3	1/0/12	100Mbps full

The configurations are as follows:

```
Switch1:
Switch1(config)#interface ethernet 1/0/7
Switch1(Config-If-Ethernet1/0/7)#bandwidth control 50000 both
Switch2:
Switch2(config)#interface ethernet 1/0/9
Switch2(Config-If-Ethernet1/0/9)#speed-duplex force100-full
Switch2(Config-If-Ethernet1/0/9)#exit
Switch2(config)#interface ethernet 1/0/10
Switch2(Config-If-Ethernet1/0/10)#speed-duplex force1g-full
Switch2(Config-If-Ethernet1/0/10)#exit
Switch2(config)#monitor session 1 source interface ethernet 1/0/8;1/0/9
Switch2(config)#monitor session 1 destination interface ethernet 1/0/10
Switch3:
Switch3(config)#interface ethernet 1/0/12
Switch3(Config-If-Ethernet1/0/12)#speed-duplex force100-full
Switch3(Config-If-Ethernet1/0/12)#exit
```

3.1.4 Port Troubleshooting

Here are some situations that occur frequently in port configuration, along with recommended solutions:

- Two connected fiber ports do not link up if one port is set to auto-negotiation and the other to a forced speed/duplex. This behavior is defined by IEEE 802.3.
- The following combinations are not recommended: enabling flow control and multicast limiting on the same port, or enabling broadcast, multicast, and unknown-destination unicast storm control together with port bandwidth limiting on the same port. With these combinations, port throughput may fall below the expected performance.

3.2 Port Isolation

3.2.1 Introduction to Port Isolation Function

Port isolation is an independent, port-based function that isolates the traffic of different ports from one another. With port isolation, ports within a VLAN can be isolated from each other, which saves VLAN resources and enhances network security. After this function is configured, the ports in a port isolation group are isolated from each other, while ports belonging to different isolation groups, or to no group at all, can forward data to one another normally. A switch can have no more than 30 port isolation groups.

3.2.2 Port Isolation Task Sequence

1. Create a port isolation group
2. Add Ethernet ports to the group
3. Specify the flow to be isolated
4. Display the port isolation configuration

1. Create a port isolation group

Command	Explanation
Global Mode	
isolate-port group <*WORD*> **no isolate-port group <*WORD*>**	Creates a port isolation group. The no command deletes the port isolation group.

2. Add Ethernet ports to the group

Command	Explanation
Global Mode	
isolate-port group <*WORD*> switchport interface [ethernet | port-channel] <*IFNAME*> **no isolate-port group <*WORD*> switchport interface [ethernet | port-channel] <*IFNAME*>**	Adds one or more ports to a port isolation group, isolating them from the other ports in the group. The no command removes the port or ports from the port isolation group.

3. Specify the flow to be isolated

Command	Explanation
Global Mode	
isolate-port apply [<l2|l3|all>]	Applies the port isolation configuration to isolate layer-2 traffic, layer-3 traffic, or all traffic.

4. Display the port isolation configuration

Command	Explanation
Admin Mode and Global Mode	
show isolate-port group [<*WORD*>]	Displays the port isolation configuration, including all configured port isolation groups and the Ethernet ports in each group.

3.2.3 Port Isolation Typical Example

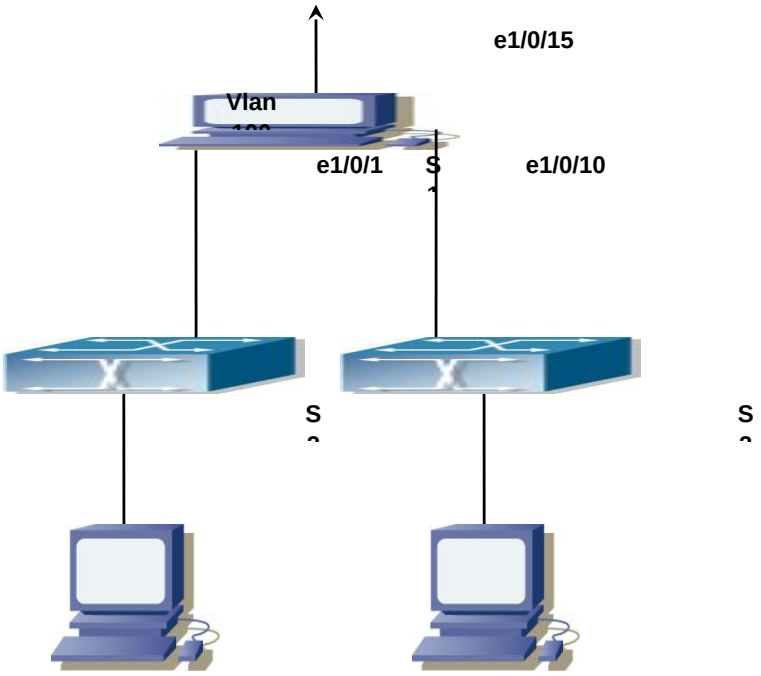


Figure 3-2 Typical example of port isolation

The topology and switch configuration are shown in the figure above; e1/0/1, e1/0/10, and e1/0/15 all belong to VLAN 100. The requirement is that, after port isolation is enabled on switch S1, e1/0/1 and e1/0/10 on S1 cannot communicate with each other, while both can communicate with the uplink port e1/0/15. That is, communication between any pair of downlink ports is blocked, while communication between any downlink port and the specified uplink port works normally. The uplink port can communicate with any port normally.

Configuration of S1:

```
Switch(config)#isolate-port group test
Switch(config)#isolate-port group test switchport interface ethernet
1/0/1;1/0/10
```

3.3 Port Loopback Detection

3.3.1 Introduction to Port Loopback Detection

As switch technology has developed, more and more users access the network through Ethernet switches. In an enterprise network, users typically connect through layer-2 switches, which creates an urgent need for both Internet access and internal layer-2 interworking between users. In layer-2 switching, frames are forwarded based on MAC addresses, so the accuracy of MAC addressing is key to correct interworking: when a port receives a frame from an unknown source MAC address, the switch learns that address on the receiving port, so that subsequent frames destined for that MAC address can be forwarded directly. The address only needs to be learned once.

If a MAC address the switch has already learned appears again on a different source port, the switch updates its table to associate that MAC address with the new port. As a result, if a loopback exists in the link, every MAC address in the layer-2 network can end up associated with the port where the loopback occurs (with addresses frequently shifting from one port to another), which can cause the layer-2 network to collapse. This is why it is necessary to check for port loopbacks in the network. When a loopback is detected, the detecting device should alert the network management system, so

that the network manager can discover, locate, and resolve the problem before it causes a prolonged network outage.

Because loopback detection can dynamically determine whether a loopback exists on a link and whether it has cleared, devices that support port control (such as port isolation and port MAC-address-learning control) can respond automatically. This reduces both the burden on network managers and the response time, minimizing the impact of loopbacks on the network.

3.3.2 Port Loopback Detection Configuration Task List

1．Configure the time interval of loopback detection
2．Enable port loopback detection
3．Configure the port loopback detection control method
4．Display and debug port loopback detection information
5．Configure the loopback-detection control mode (automatic recovery enabled or disabled)

1. Configure the time interval of loopback detection

Command	Explanation
Global Mode	
loopback-detection interval-time *<loopback> <no-loopback>* **no loopback-detection interval-time**	Configures the time interval of loopback detection. The no command restores the default interval.

2. Enable port loopback detection

Command	Explanation
Port Mode	
loopback-detection specified-vlan <vlan-*list*> **no loopback-detection specified-vlan <vlan-*list*>**	Enables port loopback detection for the specified VLANs. The no command disables it.

3. Configure the port loopback detection control method

Command	Explanation
Port Mode	
loopback-detection control {shutdown | block| learning} **no loopback-detection control**	Configures the control method of port loopback detection. The no command disables port loopback detection control.

4. Display and debug port loopback detection information

Command	Explanation
Admin Mode	
debug loopback-detection **no debug loopback-detection**	Enables debug information for the port loopback detection module. The no command disables the debug information.
show loopback-detection [interface *<interface-list>*]	Displays the loopback detection state and result of all ports if no parameter is specified; otherwise, displays the state and result of the specified ports.

5. Configure the loopback-detection control mode (automatic recovery enabled or disabled)

Command	Explanation
Global Mode	
loopback-detection control-recovery timeout <0-3600>	Configures the loopback-detection control mode (automatic recovery enabled or disabled) or the recovery time.

3.3.3 Port Loopback Detection Example

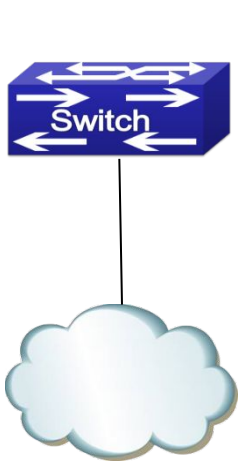


Figure 3-3 Typical example of port loopback detection

As shown in the figure above, the switch detects loopbacks in the network topology. After loopback detection is enabled on the port connecting the switch to the external network, the switch notifies the connected network of the loopback and controls the port to ensure normal operation of the entire network.

Configuration task sequence of the switch:

```
Switch(config)#loopback-detection interval-time 35 15
Switch(config)#interface ethernet 1/0/1
Switch(Config-If-Ethernet1/0/1)#loopback-detection special-vlan 1-3
Switch(Config-If-Ethernet1/0/1)#loopback-detection control block
```

If the block control method is used, MSTP must be enabled globally, and the mapping between spanning-tree instances and VLANs must be configured.

```
Switch(config)#spanning-tree
Switch(config)#spanning-tree mst configuration
Switch(Config-Mstp-Region)#instance 1 vlan 1
Switch(Config-Mstp-Region)#instance 2 vlan 2
Switch(Config-Mstp-Region)#
```

3.3.4 Port Loopback Detection Troubleshooting

Port loopback detection is disabled by default and should be enabled only when required.

3.4 ULDP

3.4.1 Introduction to ULDP

A unidirectional link is a common error state in networks, especially on fiber links. It means that only one port of the link can receive messages from the other, while the reverse direction cannot. Because the physical layer of the link is up and appears to work normally, the physical layer's own checking mechanisms cannot detect the communication problem between the devices. As shown in the figures below, a fault in a fiber connection cannot be detected through physical-layer mechanisms such as auto-negotiation.

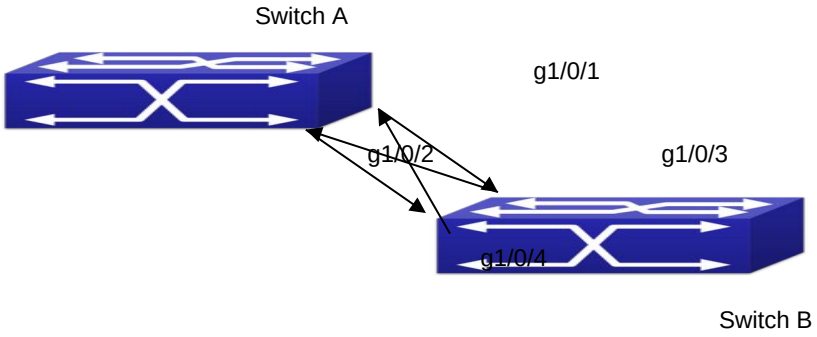


Figure 3-2 Fiber Cross Connection

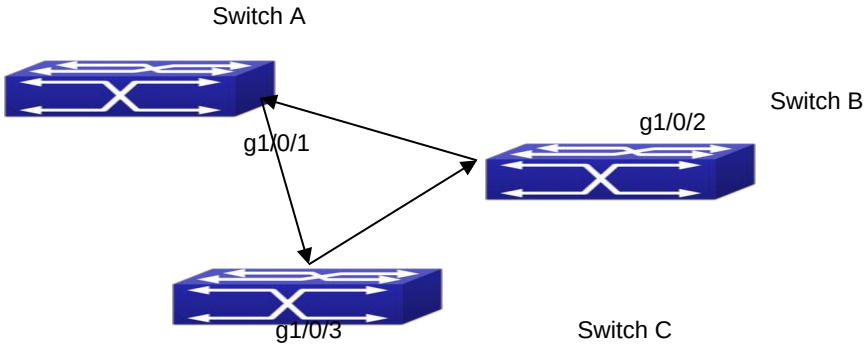


Figure 3-3 One End of Each Fiber Not Connected

This problem often occurs in the following situations: GBIC (Gigabit Interface Converter) or interface problems, software problems, or hardware that becomes unavailable or operates abnormally. A unidirectional link can cause a series of problems, such as spanning-tree topology loops and broadcast black holes.

ULDP (Unidirectional Link Detection Protocol) helps avoid the problems described above. On switches connected via fiber or copper Ethernet cabling (such as Category 5 twisted pair), ULDP can monitor the link state of physical links. Whenever a unidirectional link is discovered, it sends a warning to the user and can disable the port automatically, or manually depending on the user's configuration.

ULDP on a switch recognizes remote devices and checks the correctness of link connections by exchanging ULDP messages. When ULDP is enabled on a port, a protocol state machine starts: different types of messages are sent at different states of the state machine to verify the link's connection state by exchanging information with the remote device. ULDP can dynamically learn the interval at which the remote device sends notification messages and adjust the local TTL (time to live) accordingly. ULDP also provides a reset mechanism: when a port is disabled by ULDP, the reset mechanism can check the link again. The notification and reset intervals in ULDP are configurable, so that ULDP can respond faster to connection errors in different network environments.

For ULDP to work correctly, the link must operate in duplex mode, and ULDP must be enabled on both ends of the link using the same authentication method and password.

3.4.2 ULDP Configuration Task Sequence

1. Enable ULDP globally
2. Enable ULDP on a port
3. Configure aggressive mode globally
4. Configure aggressive mode on a port
5. Configure the method to shut down a unidirectional link
6. Configure the interval of Hello messages
7. Configure the Recovery interval
8. Reset ports shut down by ULDP
9. Display and debug ULDP information

1. Enable ULDP globally

Command	Explanation
Global Mode	
uldp enable **uldp disable**	Enables or disables ULDP globally.

2. Enable ULDP on a port

Command	Explanation
Port Mode	
uldp enable **uldp disable**	Enables or disables ULDP on a port.

3. Configure aggressive mode globally

Command	Explanation
Global Mode	
uldp aggressive-mode **no uldp aggressive-mode**	Sets the global working mode (aggressive mode). The no command restores the normal mode.

4. Configure aggressive mode on a port

Command	Explanation
Port Mode	
uldp aggressive-mode **no uldp aggressive-mode**	Sets the working mode of the port (aggressive mode). The no command restores the normal mode.

5. Configure the method to shut down a unidirectional link

Command	Explanation
Global Mode	

uldp manual-shutdown **no uldp manual-shutdown**	Configures the method used to shut down a unidirectional link (manual shutdown). The no command restores automatic shutdown.

6. Configure the interval of Hello messages

Command	Explanation
Global Mode	
uldp hello-interval <*integer*> **no uldp hello-interval**	Configures the interval of Hello messages, ranging from 5 to 100 seconds. The default is 10 seconds.

7. Configure the Recovery interval

Command	Explanation
Global Mode	
uldp recovery-time <*integer*> **no uldp recovery-time <*integer*>**	Configures the Recovery reset interval, ranging from 30 to 86400 seconds. The default is 0 seconds.

8. Reset ports shut down by ULDP

Command	Explanation
Global Mode or Port Mode	
uldp reset	Resets all ports in Global Mode; resets the specified port in Port Mode.

9. Display and debug ULDP information

Command	Explanation
Admin Mode	
show uldp [interface ethernet IFNAME]	Displays ULDP information. Without a parameter, displays global ULDP information; with a port specified, displays global information and the neighbor information of the port.
debug uldp fsm interface ethernet <*IFname*> **no debug uldp fsm interface ethernet <*IFname*>**	Enables or disables debugging of state machine transition information on the specified port.
debug uldp error **no debug uldp error**	Enables or disables debugging of error information.
debug uldp event **no debug uldp event**	Enables or disables debugging of event information.
debug uldp packet {receive|send} **no debug uldp packet {receive|send}**	Enables or disables debugging of the types of messages received and sent on all ports.
debug uldp {hello|probe|echo| unidir|all} [receive|send] interface ethernet <*IFname*> **no debug uldp {hello|probe|echo| unidir|all} [receive|send] interface ethernet <*IFname*>**	Enables or disables debugging of the detailed content of a particular type of message received and sent on the specified port.

3.4.3 ULDP Typical Example

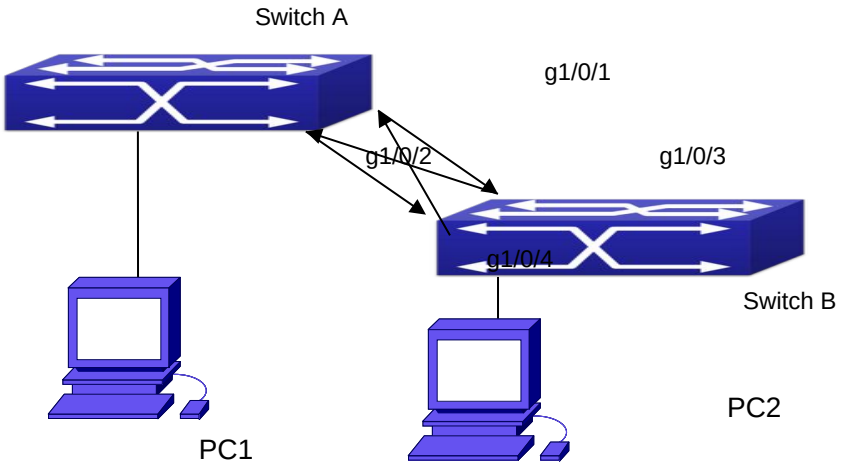


Figure 3-4 Fiber Cross Connection

In the network topology shown in the figure above, ports g1/0/1 and g1/0/2 of SWITCH A, as well as ports g1/0/3 and g1/0/4 of SWITCH B, are all fiber ports, and the connections are cross-connected. The physical layer is up and appears to work normally, but the data link layer is abnormal. ULDP can discover and disable this kind of link error. The result is that ports g1/0/1 and g1/0/2 of SWITCH A and ports g1/0/3 and g1/0/4 of SWITCH B are all shut down by ULDP. The ports work normally (are not shut down) only when the connections are correct.

```
Switch A configuration sequence:
SwitchA(config)#uldp enable
SwitchA(config)#interface ethernet 1/0/1
SwitchA(Config-If-Ethernet1/0/1)#uldp enable
SwitchA(Config-If-Ethernet1/0/1)#exit
SwitchA(config)#interface ethernet 1/0/2
SwitchA(Config-If-Ethernet1/0/2)#uldp enable
Switch B configuration sequence:
SwitchB(config)#uldp enable
SwitchB(config)#interface ethernet1/0/3
SwitchB(Config-If-Ethernet1/0/3)#uldp enable
SwitchB(Config-If-Ethernet1/0/3)#exit
SwitchB(config)#interface ethernet 1/0/4
SwitchB(Config-If-Ethernet1/0/4)#uldp enable
```

As a result, ports g1/0/1 and g1/0/2 of SWITCH A are shut down by ULDP, and notification information appears on the CRT terminal of PC1.

```
%Oct 29 11:09:50 2007 A unidirectional link is detected!  Port Ethernet1/0/1
need to be shutted down!
%Oct 29 11:09:50 2007 Unidirectional port Ethernet1/0/1 shut down!
%Oct 29 11:09:50 2007 A unidirectional link is detected!  Port Ethernet1/0/2
need to be shutted down!
%Oct 29 11:09:50 2007 Unidirectional port Ethernet1/0/2 shutted down!
```

Ports g1/0/3 and g1/0/4 of SWITCH B are shut down by ULDP, and notification information appears on the CRT terminal of PC2.

```
%Oct 29 11:09:50 2007 A unidirectional link is detected!  Port Ethernet1/0/3
need to be shutted down!
%Oct 29 11:09:50 2007 Unidirectional port Ethernet1/0/3 shutted down!
%Oct 29 11:09:50 2007 A unidirectional link is detected!  Port Ethernet1/0/4
need to be shutted down!
%Oct 29 11:09:50 2007 Unidirectional port Ethernet1/0/4 shutted down!
```

3.4.4 ULDP Troubleshooting

Configuration notes:

- To ensure that ULDP can detect that one of the fiber ports is not connected, or that the ports are incorrectly cross-connected, the ports must operate in duplex mode and have the same rate.
- If the auto-negotiation mechanism of the fiber ports, with one port misconnected, determines the working mode and rate of the ports, ULDP will not take effect whether it is enabled or not. In this situation, the port is considered "Down".
- To ensure that neighbors are correctly established and unidirectional links are correctly discovered, both ends of the link are required to enable ULDP, using the same authentication method and password. Currently, no password is required on either end.
- The interval for sending Hello messages can be changed (it is 10 seconds by default and ranges from 5 to 100 seconds), so that ULDP can respond faster to connection errors in different network environments. However, this interval should be less than one-third of the STP convergence time. If the interval is too long, an STP loop may form before ULDP discovers and shuts down the unidirectional link. If the interval is too short, the load on the port will increase, reducing available bandwidth.
- ULDP does not handle any LACP events. It treats every link in a TRUNK group (such as Port-channel or TRUNK ports) as independent and handles each one individually.
- ULDP is not compatible with similar protocols from other vendors, which means ULDP cannot be used on one end of a link with a similar protocol from another vendor on the other end.
- ULDP is disabled by default. After ULDP is enabled globally, the debug switch can also be enabled to check debug information. Several DEBUG commands are provided to print debug information, such as events, state machine transitions, errors, and messages. Different types of message information can also be printed depending on the parameters.
- The Recovery timer is disabled by default and is enabled only after the recovery time (30-86400 seconds) has been configured.
- The reset command and reset mechanism can only reset ports that were automatically shut down by ULDP. Ports shut down manually by users or by other modules are not reset by ULDP.

3.5 LLDP

3.5.1 Introduction to LLDP

Link Layer Discovery Protocol (LLDP) is a protocol defined in 802.1AB. It enables neighboring devices to advertise their own status to other devices, and enables every port on a device to store information about its neighbors. If necessary, ports can also send update information directly to their connected neighbor devices, and those neighbor devices store the information in standard SNMP MIBs. The network management system can then check the layer-2 connection state from the MIB. LLDP does not configure or control network elements or traffic; it only reports layer-2 configuration information. Another part of 802.1AB is to use the information LLDP provides to detect layer-2 conflicts. IEEE currently reuses the existing physical topology, interface, and Entity MIBs defined by the IETF.

In simple terms, LLDP is a neighbor discovery protocol. It defines a standard method for Ethernet devices, such as switches, routers, and WLAN access points, to advertise their existence to other nodes in the network and to store discovery information about all neighboring devices. For example, detailed device configuration and discovery information can both be advertised using this protocol.

Specifically, LLDP defines a general set of advertisement information, a transport protocol for that advertisement, and a method for storing the received advertisement information. A device advertising its own information can place multiple pieces of advertisement information into a single LAN data

packet for transport, using the type-length-value (TLV) field format. All devices that support LLDP must support device ID and port ID advertisement; most devices are also expected to support system name, system description, and system capability advertisement. System name and system description advertisements can also provide useful information for collecting network traffic data. System description advertisement can include data such as the full name of the advertising device, its hardware type, and its software/operating-system version information.

The 802.1AB Link Layer Discovery Protocol makes it easier to search for problems in an enterprise network and strengthens the ability of network management tools to discover and maintain an accurate network topology.

Many types of network management software use an "Automated Discovery" function to track topology changes and status, but at best most can only reach layer three and classify devices by IP subnet. This kind of data is very basic, capturing only high-level events such as devices being added or removed, rather than details of where and how these devices operate within the network.

Layer-2 discovery covers information such as which devices have which ports and which switches connect to other devices, and it can also display the paths between clients, switches, routers, application servers, and network servers. Such details are very useful for scheduling and investigating the source of network failures.

LLDP is a very useful management tool that provides accurate information for network mirroring, traffic data, and troubleshooting network problems.

3.5.2 LLDP Configuration Task Sequence

1. Enable LLDP globally
2. Configure the port-based LLDP function switch
3. Configure the LLDP operating state of the port
4. Configure the interval of LLDP update messages
5. Configure the aging time multiplier of LLDP messages
6. Configure the sending delay of update messages
7. Configure the interval of sending Trap messages
8. Enable the Trap function of the port
9. Configure the optional information-sending attribute of the port
10. Configure the size of the Remote Table of the port
11. Configure the type of operation when the Remote Table of the port is full
12. Display and debug LLDP information

1. **Enable LLDP globally**

Command	Explanation
Global Mode	
lldp enable **lldp disable**	Enables or disables LLDP globally.

2. **Configure the port-based LLDP function switch**

Command	Explanation
Port Mode	
lldp enable **lldp disable**	Enables or disables LLDP on the port.

3. **Configure the LLDP operating state of the port**

Command	Explanation

Port Mode	
lldp mode (send|receive|both|disable)	Configures the LLDP operating state of the port.

4. **Configure the interval of LLDP update messages**

Command	Explanation
Global Mode	
lldp tx-interval <*integer*> **no lldp tx-interval**	Configures the interval of LLDP update messages. The no command restores the default value.

5. **Configure the aging time multiplier of LLDP messages**

Command	Explanation
Global Mode	
lldp msgTxHold <*value*> **no lldp msgTxHold**	Configures the aging time multiplier of LLDP messages. The no command restores the default value.

6. **Configure the sending delay of update messages**

Command	Explanation
Global Mode	
lldp transmit delay <*seconds*> **no lldp transmit delay**	Configures the sending delay of update messages. The no command restores the default value.

7. **Configure the interval of sending Trap messages**

Command	Explanation
Global Mode	
lldp notification interval <*seconds*> **no lldp notification interval**	Configures the interval of sending Trap messages. The no command restores the default value.

8. **Enable the Trap function of the port**

Command	Explanation
Port Mode	
lldp trap <*enable*|*disable*>	Enables or disables the Trap function of the port.

9. **Configure the optional information-sending attribute of the port**

Command	Explanation
Port Mode	
lldp transmit optional tlv [portDesc] [sysName] [sysDesc] [sysCap] **no lldp transmit optional tlv**	Configures the optional TLVs sent by the port. The no command restores the default value.

10. **Configure the size of the Remote Table of the port**

Command	Explanation
Port Mode	
lldp neighbors max-num < *value* > **no lldp neighbors max-num**	Configures the size of the Remote Table of the port. The no command restores the default value.

11. **Configure the type of operation when the Remote Table of the port is full**

Command	Explanation
Port Mode	

lldp tooManyNeighbors {discard | delete}	Configures the action taken when the Remote Table of the port is full.

12. **Display and debug LLDP information**

Command	Explanation
Admin Mode and Global Mode	
show lldp	Displays the current LLDP configuration.
show lldp interface ethernet *<IFNAME>*	Displays the LLDP configuration of the specified port.
show lldp traffic	Displays all LLDP counters.
show lldp neighbors interface ethernet < *IFNAME* >	Displays information about the LLDP neighbors of the specified port.
show debugging lldp	Displays all ports on which LLDP debugging is enabled.
Admin Mode	
debug lldp **no debug lldp**	Enables or disables the LLDP DEBUG switch.
debug lldp packets interface ethernet *<IFNAME>* **no debug lldp packets interface ethernet *<IFNAME>***	Enables or disables debugging of LLDP packets received and sent on the port, in Port Mode or Global Mode.
Port Mode	
clear lldp remote-table	Clears the Remote Table of the port.

3.5.3 LLDP Typical Example

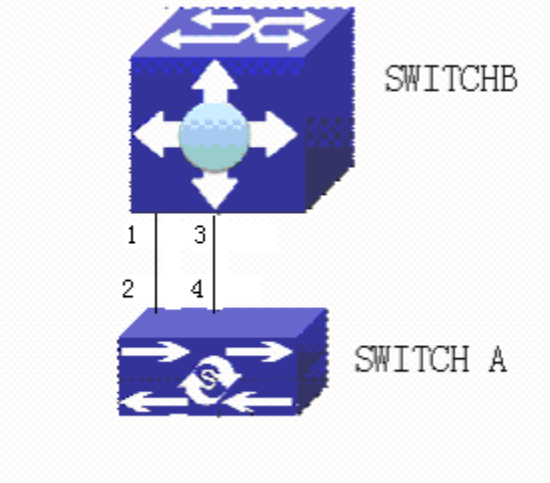


Figure 3-7 LLDP typical configuration example

In the network topology shown above, ports 1 and 3 of SWITCH B are connected to ports 2 and 4 of SWITCH A. Port 1 of SWITCH B is configured in receive-only mode, and the optional TLVs of port 4 of SWITCH A are configured as portDesc and sysCap.

SWITCH A configuration sequence:

```
SwitchA(config)# lldp enable
SwitchA(config)#interface ethernet 1/0/4
SwitchA(Config-If-Ethernet1/0/4)#lldp transmit optional tlv portDesc sysCap
SwitchA(Config-If-Ethernet1/0/4)exit
```

SWITCH B configuration sequence:

```
SwitchB(config)#lldp enable
SwitchB(config)#interface ethernet1/0/1
```

```
SwitchB(Config-If-Ethernet1/0/1)#lldp mode receive
SwitchB(Config-If-Ethernet1/0/1)#exit
```

3.5.4 LLDP Function Troubleshooting

☞ LLDP is disabled by default. After LLDP is enabled globally, the debug switch "**debug lldp**" can also be enabled to check debug information.

☞ The LLDP **show** commands display configuration information in Global Mode or Port Mode.

3.6 LLDP-MED

3.6.1 Introduction to LLDP-MED

LLDP-MED (Link Layer Discovery Protocol-Media Endpoint Discovery) is based on IEEE 802.1AB LLDP (Link Layer Discovery Protocol). LLDP provides a standard link-layer discovery method: it sends local device information (including major capabilities, management IP address, device ID, and port ID) as TLV (type/length/value) triplets in LLDPDUs (Link Layer Discovery Protocol Data Units) to directly connected neighbors. The neighbors store the received device information in a standard management information base (MIB). This allows a network management system to quickly detect and identify the communication status of the link.

802.1AB LLDP does not transmit or manage voice device information. To simplify the deployment and management of voice devices, LLDP-MED TLVs provide various information, such as PoE (Power over Ethernet), network policy, and location information for emergency telephone services.

3.6.2 LLDP-MED Configuration Task Sequence

1. Basic LLDP-MED configuration

Command	Explanation
Port Mode	
lldp transmit med tlv all **no lldp transmit med tlv all**	Configures the specified port to send all LLDP-MED TLVs. The no command disables the function.
lldp transmit med tlv capability **no lldp transmit med tlv capability**	Configures the specified port to send the LLDP-MED Capability TLV. The no command disables the capability.
lldp transmit med tlv networkPolicy **no lldp transmit med tlv networkPolicy**	Configures the specified port to send the LLDP-MED Network Policy TLV. The no command disables the capability.
lldp transmit med tlv extendPoe **no lldp transmit med tlv extendPoe**	Configure the specified port to send LLDP-MED Extended Power-Via-MDI TLV. The no command disables the capability.
lldp transmit med tlv location **no lldp transmit med tlv location**	Configures the specified port to send the LLDP-MED Location Identification TLV. The no command disables the capability.

lldp transmit med tlv inventory **no lldp transmit med tlv inventory**	Configures the port to send LLDP-MED Inventory Management TLVs. The no command disables the capability.
network policy {voice | voice-signaling | guest-voice | guest-voice-signaling | softphone-voice | video-conferencing | streaming-video | video-signaling} [status {enable | disable}] [tag {tagged | untagged}] [vid {<*vlan-id*> | dot1p}] [cos <*cos-value*>] [dscp <*dscp-value*>] **no network policy {voice | voice-signaling | guest-voice | guest-voice-signaling | softphone-voice | video-conferencing | streaming- video | video-signaling}**	Configures the network policy of the port, including the VLAN ID, the supported application (such as voice and video), the application priority, and the policy used.
civic location {dhcp server | switch | endpointDev} <*country-code*> **no civic location**	Configures the device type and country code of the location in Civic Address LCI format and enters Civic Address LCI address mode. The no command cancels all location configuration in Civic Address LCI format.
ecs location <*tel-number*> **no ecs location**	Configures the location in ECS ELIN format on the port. The no command cancels the configured location.
lldp med trap {enable | disable}	Enables or disables LLDP-MED traps for the specified port.
Civic Address LCI Address Mode	
{description-language | province-state | city | county | street | locationNum | location | floor | room | postal | otherInfo} <*address*> **no {description-language | province-state | city | county | street | locationNum | location | floor | room | postal | otherInfo}**	Configures the detailed address after entering Civic Address LCI address mode on the port.
Global Mode	
lldp med fast count <*value*> **no lldp med fast count**	Sets the number of LLDP packets with LLDP-MED TLVs sent rapidly when the LLDP-MED fast start mechanism is triggered. The no command restores the default value.
Admin Mode	
show lldp	Displays the global LLDP and LLDP-MED configuration.
show lldp [interface ethernet <*IFNAME*>]	Displays the LLDP and LLDP-MED configuration of the specified port.
show lldp neighbors [interface ethernet <*IFNAME*>]	Displays the LLDP and LLDP-MED configuration of the neighbors.
show lldp traffic	Displays statistics on LLDP and LLDP-MED packets sent and received on the ports.

3.6.3 LLDP-MED Example

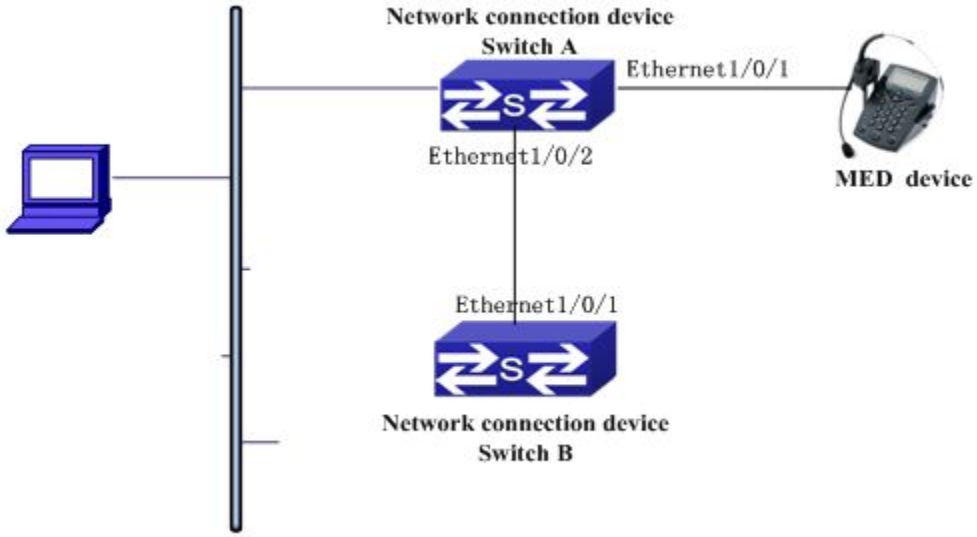


Figure 3-5 Basic LLDP-MED configuration topology

1) Configure Switch A

```
SwitchA(config)#interface ethernet1/0/1
SwitchA (Config-If-Ethernet1/0/1)# lldp enable
SwitchA (Config-If-Ethernet1/0/1)# lldp mode both (this configuration can be
omitted, the default mode is RxTx)
SwitchA (Config-If-Ethernet1/0/1)# lldp transmit med tlv capability
SwitchA (Config-If-Ethernet1/0/1)# lldp transmit med tlv network policy
SwitchA (Config-If-Ethernet1/0/1)# lldp transmit med tlv inventory
SwitchB (Config-If-Ethernet1/0/1)# network policy voice tag tagged vid 10 cos
5 dscp 15
SwitchA (Config-If-Ethernet1/0/1)# exit
SwitchA (config)#interface ethernet1/0/2
SwitchA (Config-If-Ethernet1/0/2)# lldp enable
SwitchA (Config-If-Ethernet1/0/2)# lldp mode both
```

2) Configure Switch B

```
SwitchB (config)#interface ethernet1/0/1
SwitchB(Config-If-Ethernet1/0/1)# lldp enable
SwitchB (Config-If-Ethernet1/0/1)# lldp mode both
SwitchB (Config-If-Ethernet1/0/1)# lldp transmit med tlv capability
SwitchB (Config-If-Ethernet1/0/1)# lldp transmit med tlv network policy
SwitchB (Config-If-Ethernet1/0/1)# lldp transmit med tlv inventory
SwitchB (Config-If-Ethernet1/0/1)# network policy voice tag tagged vid 10 cos
4
```

3) Verify the configuration

```
# Show the global status and interface status on Switch A.
SwitchA# show lldp neighbors interface ethernet 1/0/1
Port name : Ethernet1/0/1
Port Remote Counter : 1
TimeMark :20
ChassisIdSubtype :4
ChassisId :00-03-0f-00-00-02
PortIdSubtype :Local
PortId :1
PortDesc :****
SysName :****
SysDesc :*****
SysCapSupported :4
```

```
SysCapEnabled :4
LLDP MED Information :
MED Codes:
(CAP)Capabilities, (NP) Network Policy
(LI) Location Identification, (PSE)Power Source Entity
(PD) Power Device, (IN) Inventory
MED Capabilities:CAP,NP,PD,IN
MED Device Type: Endpoint Class III
Media Policy Type :Voice
Media Policy :Tagged
Media Policy Vlan id :10
Media Policy Priority :3
Media Policy Dscp :5
Power Type : PD
Power Source :Primary power source
Power Priority :low
Power Value :15.4 (Watts)
Hardware Revision:
Firmware Revision:4.0.1
Software Revision:6.2.30.0
Serial Number:
Manufacturer Name:****
Model Name:Unknown
Assert ID:Unknown
IEEE 802.3 Information :
 auto-negotiation support: Supported
 auto-negotiation support: Not Enabled
 PMD auto-negotiation advertised capability: 1
 operational MAU type: 1
SwitchA# show lldp neighbors interface ethernet 1/0/2
Port name : interface ethernet 1/0/2
Port Remote Counter：1
Neighbor Index: 1
Port name : Ethernet1/0/2
Port Remote Counter : 1
TimeMark :20
ChassisIdSubtype :4
ChassisId :00-03-0f-00-00-02
PortIdSubtype :Local
PortId :1
PortDesc :Ethernet1/0/1
SysName :****
SysDesc :*****
SysCapSupported :4
SysCapEnabled :4
```

Explanation:

1) Ethernet2 of switch A and Ethernet1 of switch B are both ports of network connection devices, so they do not proactively send LLDP packets with MED TLV information. Although Ethernet1 of switch B is configured to send MED TLV information, it does not send the related MED information, so the corresponding Remote Table on Ethernet2 of switch A contains no MED information.

2) An LLDP-MED device can proactively send LLDP packets with MED TLVs, so the corresponding Remote Table on Ethernet1 of switch A contains LLDP-MED information.

3.6.4 LLDP-MED Troubleshooting

If problems occur when configuring LLDP-MED, check whether they are caused by the following:

- ☞ Global LLDP is not enabled.
- ☞ A network connection device sends LLDP-MED TLVs only after it has received LLDP packets with LLDP-MED TLVs from a neighboring MED device. If the port of a network connection device

has the command for sending LLDP-MED TLVs configured but the packets it sends still lack LLDP-MED TLVs, no MED information has been received and the port has not enabled sending LLDP-MED information.

- ☞ If a neighbor device has sent LLDP-MED information to the network connection device, but the **show lldp neighbors** command shows no LLDP-MED information, the LLDP-MED information sent by the neighbor is erroneous.

3.7 Port Channel

3.7.1 Introduction to Port Channel

To understand Port Channel, it helps to first understand Port Group. A Port Group is a group of physical ports at the configuration level; only physical ports in a Port Group can take part in link aggregation and become member ports of a Port Channel. Logically, a Port Group is not a single port but a sequence of ports. Under certain conditions, the physical ports in a Port Group are aggregated to form a Port Channel, which has all the properties of a logical port and thus becomes an independent logical port. Port aggregation is a process of logical abstraction that combines a set of ports (a port sequence) with the same properties into a single logical port. A Port Channel is a collection of physical ports used logically as one physical port. A Port Channel can be used like a normal port; it not only increases network bandwidth but also provides link backup. Port aggregation is typically used when a switch is connected to routers, PCs, or other switches.

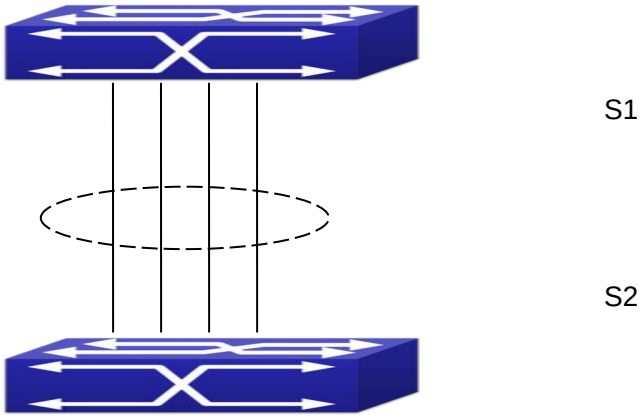


Figure 3-6 Port aggregation

As shown above, the four ports of S1 are aggregated into a Port Channel, whose bandwidth is the total of all four ports. If traffic from S1 needs to be transferred to S2 through the Port Channel, the traffic allocation is calculated based on the source MAC address and the lowest bit of the target MAC address. The result of this calculation determines which port carries the traffic. If a port in the Port Channel fails, the other ports take over that port's traffic using a traffic-allocation algorithm, which is carried out by the hardware.

The switch offers two methods for configuring port aggregation: manual Port Channel creation and dynamic Port Channel creation with LACP (Link Aggregation Control Protocol). Port aggregation can only be performed on ports in full-duplex mode.

For a Port Channel to work properly, its member ports must have the following properties in common:

- ☞ All ports are in full-duplex mode.
- ☞ All ports have the same speed.
- ☞ All ports are Access ports belonging to the same VLAN, or are all TRUNK ports, or are all Hybrid ports.

☞ If the ports are all TRUNK ports or Hybrid ports, their "Allowed VLAN" and "Native VLAN" settings must also match.

If a Port Channel is configured manually or dynamically on the switch, the system automatically sets the port with the smallest number as the Master Port of the Port Channel. If the spanning tree function is enabled on the switch, the spanning tree protocol treats the Port Channel as a logical port and sends BPDU frames via the master port.

Port aggregation is closely tied to switch hardware. The switch supports physical port aggregation between any two switches, with a maximum of 128 groups and up to 8 ports per port group.

Once ports are aggregated, they can be used as a normal port. The switch has a built-in aggregation interface configuration mode (Port-channel Mode), in which related configuration can be performed just as in VLAN or physical interface configuration mode.

3.7.2 Brief Introduction to LACP

LACP (Link Aggregation Control Protocol) is a protocol based on the IEEE 802.3ad standard used to implement dynamic link aggregation. LACP uses LACPDUs (Link Aggregation Control Protocol Data Units) to exchange information with the other end.

After LACP is enabled on a port, that port sends an LACPDU to the other end to advertise the system priority, the system's MAC address, the port priority, the port ID, and the operation key. When the other end receives this information, it compares it with the stored information of other ports to select which ports can be aggregated, so both sides can agree on which ports join or leave the dynamic aggregation group.

The operation key is generated by LACP based on the combination of configuration settings (speed, duplex, basic configuration, management key) of the ports to be aggregated.

After LACP is enabled on a dynamic aggregation port, the management key is 0 by default. After LACP is enabled on a static aggregation port, the port's management key is the same as the ID of the aggregation group.

In a dynamic aggregation group, members of the same group share the same operation key; in a static aggregation group, the active ports share the same operation key.

Port aggregation combines multiple ports into an aggregation group, so as to balance inbound/outbound traffic across the member ports of the group and provide better reliability.

3.7.2.1 Static LACP Aggregation

Static LACP aggregation is enforced by user configuration and does not require LACP to be enabled. When configuring static LACP aggregation, use "on" mode to force the port into the aggregation group.

3.7.2.2 Dynamic LACP Aggregation

1. Summary of dynamic LACP aggregation

Dynamic LACP aggregation is created and deleted automatically by the system; member ports of a dynamic LACP aggregation cannot be added or deleted manually. Ports that have the same speed and duplex settings, are connected to the same device, and have the same basic configuration can be dynamically aggregated together. Even a single port can form a dynamic aggregation on its own, known as single-port aggregation. In dynamic aggregation, LACP is enabled on the port.

2. Port states in a dynamic aggregation group

In a dynamic aggregation group, ports have two states: selected or standby. Both selected and standby ports can send and receive LACP packets, but standby ports cannot forward data packets.

Because of the limit on the maximum number of ports in an aggregation group, if the current number of member ports exceeds that limit, the system on this end negotiates with the other end to decide port states based on port ID. The negotiation steps are as follows:

Compare the device IDs (system priority plus system MAC address). First compare the system priorities; if they are the same, compare the system MAC addresses. The end with the smaller device ID has the higher priority.

Compare the port IDs (port priority plus port ID). On the device with the higher device priority, first compare the port priorities; if they are the same, compare the port IDs. The port with the smaller port ID is selected, and the others become standby ports.

In an aggregation group, the selected port with the smallest port ID becomes the master port; the other selected ports become member ports.

3.7.3 Introduction to Load Balance

Network traffic and data flow are constantly increasing, along with processing and computing demands. If a large volume of data flows through a single physical port of the switch at once, it can cause network congestion. If a switch has too many physical ports in use, it can waste resources. Load Balance is a method to expand network device and server bandwidth, increase throughput, improve network flexibility, and strengthen data processing.

3.7.4 Port Channel Configuration Task List

1. Create a port group in Global Mode
2. Add ports to the specified group in Port Mode of each port
3. Enter Port-channel Mode
4. Set the load-balance method of the switch
5. Set the LACP system priority
6. Set the LACP port priority of the current port
7. Set the LACP timeout mode of the current port

1. Create a port group

Command	Explanation
Global Mode	
port-group <*port-group-number*> **no port-group <*port-group-number*>**	Creates or deletes a port group.

2. Add physical ports to the port group

Command	Explanation
Port Mode	
port-group <*port-group-number*> mode {active | passive | on} **no port-group**	Adds the ports to the port group and sets their mode. The no command removes the ports from the port group.

3. Enter Port-channel Mode

Command	Explanation
Global Mode	
interface port-channel <*port-channel-number*>	Enters Port-channel Mode.

4. Set the load-balance method of the switch

Command	Explanation
Global Mode	

load-balance {dst-src-mac | dst-src-ip | dst-src-mac-ip}	Sets the load-balance method of the switch; it applies to both port groups and ECMP.

5. Set the LACP system priority

Command	Explanation
Global Mode	
lacp system-priority <*system-priority*> **no lacp system-priority**	Sets the LACP system priority. The no command restores the default value.

6. Set the LACP port priority of the current port

Command	Explanation
Port Mode	
lacp port-priority <*port-priority*> **no lacp port-priority**	Sets the LACP port priority. The no command restores the default value.

7. Set the LACP timeout mode of the current port

Command	Explanation
Port Mode	
lacp timeout {short | long} **no lacp timeout**	Sets the LACP timeout mode. The no command restores the default value.

3.7.5 Port Channel Examples

Scenario 1: Configuring a Port Channel with LACP

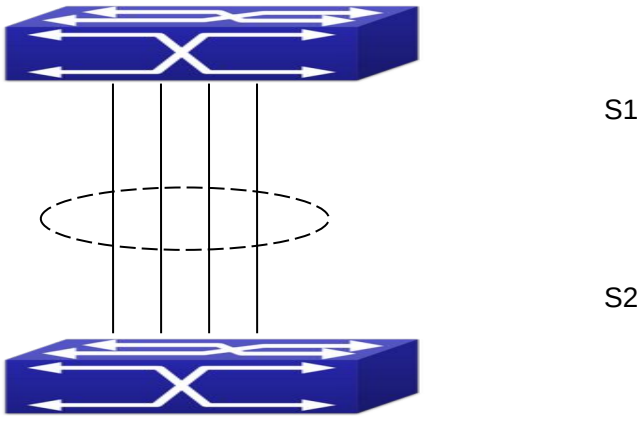


Figure 3-10 Configuring a Port Channel with LACP

As shown in the figure, ports 1, 2, 3, and 4 of S1 are access ports and are added to group 1 in active mode. Ports 6, 8, 9, and 10 of S2 are access ports and are added to group 2 in passive mode. All the ports must be connected with cables.

The configuration steps are as follows:

```
Switch1#config
Switch1(config)#interface ethernet 1/0/1-4
Switch1(Config-If-Port-Range)#port-group 1 mode active
Switch1(Config-If-Port-Range)#exit
Switch1(config)#interface port-channel 1
Switch1(Config-If-Port-Channel1)#
Switch2#config
Switch2(config)#port-group 2
Switch2(config)#interface ethernet 1/0/6
Switch2(Config-If-Ethernet1/0/6)#port-group 2 mode passive
```

```
Switch2(Config-If-Ethernet1/0/6)#exit
Switch2(config)#interface ethernet 1/0/8-10
Switch2(Config-If-Port-Range)#port-group 2 mode passive
Switch2(Config-If-Port-Range)#exit
Switch2(config)#interface port-channel 2
Switch2(Config-If-Port-Channel2)#
```

Configuration result:

After a while, the shell reports that the ports are aggregated successfully. Ports 1, 2, 3, and 4 of S1 form an aggregated port named "Port-Channel1", and ports 6, 8, 9, and 10 of S2 form an aggregated port named "Port-Channel2". Each can be configured in its own Port-channel Mode.

Scenario 2: Configuring a Port Channel in ON mode

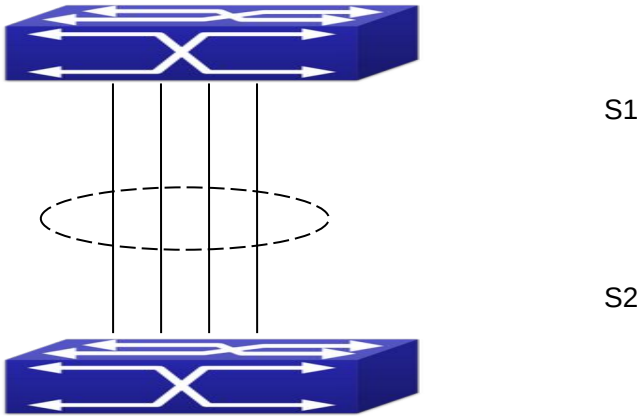


Figure 3-11 Configuring a Port Channel in ON mode

As shown in the figure, ports 1, 2, 3, and 4 of S1 are access ports and are added to group 1 in "on" mode. Ports 6, 8, 9, and 10 of S2 are access ports and are added to group 2 in "on" mode.

The configuration steps are as follows:

```
Switch1#config
Switch1(config)#interface ethernet 1/0/1
Switch1(Config-If-Ethernet1/0/1)#port-group 1 mode on
Switch1(Config-If-Ethernet1/0/1)#exit
Switch1(config)#interface ethernet 1/0/2
Switch1 (Config-If-Ethernet1/0/2)#port-group 1 mode on
Switch1 (Config-If-Ethernet1/0/2)#exit
Switch1 (config)#interface ethernet 1/0/3
Switch1 (Config-If-Ethernet1/0/3)#port-group 1 mode on
Switch1 (Config-If-Ethernet1/0/3)#exit
Switch1 (config)#interface ethernet 1/0/4
Switch1 (Config-If-Ethernet1/0/4)#port-group 1 mode on
Switch1 (Config-If-Ethernet1/0/4)#exit
Switch2#config
Switch2(config)#port-group 2
Switch2(config)#interface ethernet 1/0/6
Switch2 (Config-If-Ethernet1/0/6)#port-group 2 mode on
Switch2 (Config-If-Ethernet1/0/6)#exit
Switch2 (config)#interface ethernet 1/0/8-10
Switch2(Config-If-Port-Range)#port-group 2 mode on
Switch2(Config-If-Port-Range)#exit
```

Configuration result:

Ports 1, 2, 3, and 4 of S1 are added to port-group 1 in order. A group in "on" mode is joined forcibly, and the switch at the other end does not exchange LACP PDUs to complete aggregation. Aggregation completes immediately when the command to add port 1/0/2 to port-group 1 is entered: ports 1 and 2 aggregate into port-channel 1. When port 1/0/3 joins port-group 1, port-channel 1 (ports

1 and 2) is ungrouped and re-aggregated with port 3 to form port-channel 1. When port 1/0/4 joins port-group 1, port-channel 1 (ports 1, 2, and 3) is ungrouped and re-aggregated with port 4 to form port-channel 1. (Note that whenever a new port joins an aggregated port group, the group is first ungrouped and then re-aggregated to form a new group.) All four ports on S1 and on S2 are now aggregated in "on" mode, each set forming one aggregated port.

3.7.6 Troubleshooting

3.7.6.1 Port Channel Troubleshooting

If problems occur when configuring port aggregation, first check the following possible causes.

- ☞ Ensure that all ports in a port group have the same properties, i.e., they are in full-duplex mode, forced to the same speed, and have the same VLAN properties. Correct any inconsistency.
- ☞ Some commands cannot be used on a port in a port channel, such as arp, bandwidth, ip, and ip-forward.

3.7.6.2 Load Balance Troubleshooting

3.8 MTU

3.8.1 Introduction to MTU

There is currently no industry standard for Jumbo frames (including frame format and length). Frames between 1519 and 12000 bytes are normally considered jumbo frames. Networks using jumbo frames can increase overall network speed by 2% to 5%. Technically, a Jumbo frame is simply a longer frame sent and received by the switch. However, because of their length, Jumbo frames are not sent to the CPU; Jumbo frames sent to the CPU are discarded during packet reception.

3.8.2 MTU Configuration Task Sequence

1. Enable the MTU function

1. Enable the MTU function

Command	Explanation
Global Mode	
mtu [<mtu-value>] **no mtu enable**	Enables sending and receiving of MTU (jumbo) frames. The no command disables sending and receiving of MTU frames.

3.9 BPDU Tunnel

3.9.1 Introduction to BPDU Tunnel

BPDU Tunnel is a Layer 2 tunneling technology. It allows Layer 2 protocol packets from geographically dispersed private network users to be transmitted transparently over specific tunnels across a service provider network.

3.9.1.1 BPDU Tunnel Function

In MAN applications, multiple branches of a company may connect to each other through a service provider network. A VPN provided by the service provider enables the geographically dispersed networks to form a single LAN, so the service provider needs to provide a tunnel function: data generated by the user's network must reach the company's other networks through the service

provider network intact. To maintain the concept of a local network, not only the data within the user's private network but also the Layer 2 protocol packets within it must be transmitted across the tunnel.

3.9.1.2 Background of BPDU Tunnel

Dedicated lines are used in a service provider network to build user-specific Layer 2 networks. As a result, a user network is split into parts located on different sides of the service provider network. As shown in the figure below, User A has two devices (CE 1 and CE 2), both in the same VLAN. The user's network is divided into network 1 and network 2, which are connected by the service provider network. If Layer 2 protocol packets cannot pass through the service provider network, the user's networks cannot perform independent Layer 2 protocol calculations (for example, spanning tree calculation), and they affect each other.

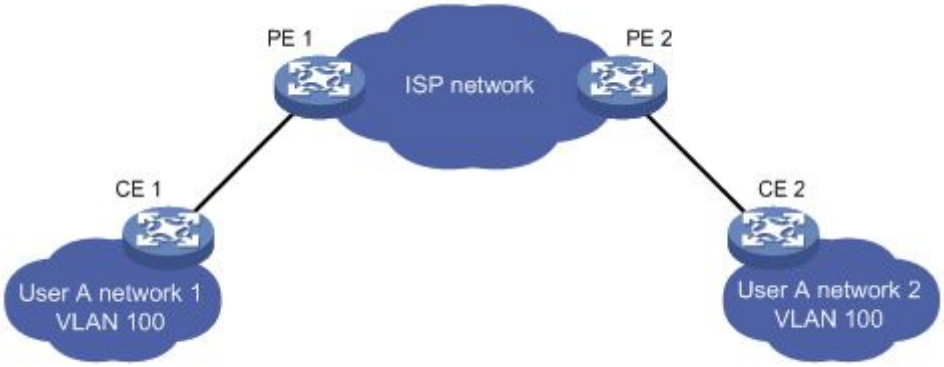


Figure 3-12 BPDU Tunnel application

3.9.2 BPDU Tunnel Configuration Task List

BPDU Tunnel configuration task list:

1. Configure the tunnel MAC address globally
2. Configure the port to support the tunnel

1. Configure the tunnel MAC address globally

Command	Explanation
Global Mode	
bpdu-tunnel dmac <mac> **no bpdu-tunnel dmac**	Configures or cancels the tunnel MAC address globally.

2. Configure the port to support the tunnel

Command	Explanation
Port Mode	
bpdu-tunnel {stp|gvrp|uldp|lacp|dot1x} **no bpdu-tunnel {stp|gvrp|uldp|lacp|dot1x}**	Enables the port to support the tunnel for the specified protocol. The no command disables the function.

3.9.3 BPDU Tunnel Example

Dedicated lines are used in a service provider network to build user-specific Layer 2 networks. As a result, a user network is split into parts located on different sides of the service provider network. As shown in the figure below, User A has two devices (CE 1 and CE 2), both in the same VLAN. The user's network is divided into network 1 and network 2, which are connected by the service provider network. If Layer 2 protocol packets cannot pass through the service provider network, the user's

networks cannot perform independent Layer 2 protocol calculations (for example, spanning tree calculation), and they affect each other.

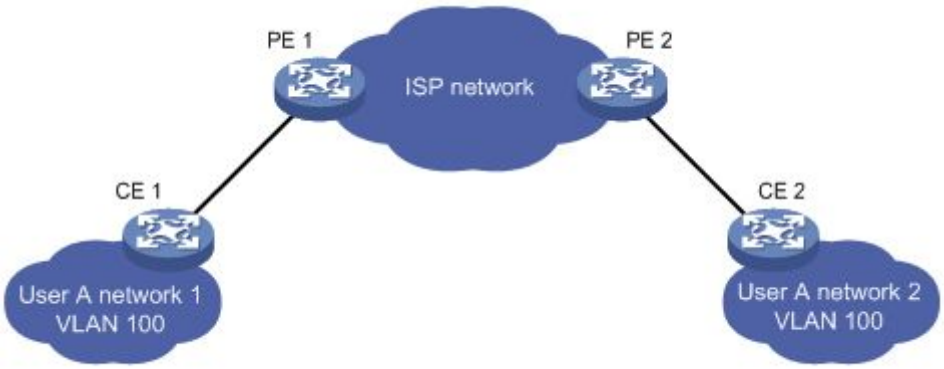


Figure 3-13 BPDU Tunnel application environment

With BPDU Tunnel, Layer 2 protocol packets from the user's networks can pass through the service provider network as follows:

1. After receiving a Layer 2 protocol packet from network 1 of user A, PE 1 in the service provider network encapsulates the packet, replaces its destination MAC address with a specific multicast MAC address, and then forwards the packet in the service provider network.

2. The encapsulated Layer 2 protocol packet (called a BPDU Tunnel packet) is forwarded to PE 2 at the other end of the service provider network, which decapsulates the packet, restores its original destination MAC address, and then sends the packet to network 2 of user A.

The BPDU Tunnel configuration of edge switches PE1 and PE2 is as follows:

PE1 configuration:

```
PE1(config)# bpdu-tunnel dmac 01-02-03-04-05-06
PE1(config-if-ethernet1/0/1)# bpdu-tunnel stp
PE1(config-if-etherne1/0/1)# bpdu-tunnel lacp
PE1(config-if-ethernet1/0/1)# bpdu-tunnel uldp
PE1(config-if-ethernet1/0/1)# bpdu-tunnel gvrp
PE1(config-if-ethernet1/0/1)# bpdu-tunnel dot1x
```

PE2 configuration:

```
PE2(config)# bpdu-tunnel dmac 01-02-03-04-05-06
PE2(config-if-ethernet1/0/1)# bpdu-tunnel stp
PE2(config-if-ethernet1/0/1)# bpdu-tunnel lacp
PE2(config-if-ethernet1/0/1)# bpdu-tunnel uldp
PE2(config-if-ethernet1/0/1)# bpdu-tunnel gvrp
PE2(config-if-ethernet1/0/1)# bpdu-tunnel dot1x
```

3.9.4 bpdu-tunnel Troubleshooting

BPDU Tunnel can be configured on a port only after STP, GVRP, ULDP, LACP, and dot1x are disabled on that port.

3.10 DDM

3.10.1 Introduction to DDM

3.10.1.1 Brief Introduction to DDM

DDM (Digital Diagnostic Monitoring) implements the detailed digital diagnostic function defined in the SFF-8472 MSA. Monitored parameter signals are digitized on the module's internal circuit board. The calibrated results, or the digitized measurement results together with the calibration parameters, are then stored in the standard memory map, where they can be conveniently read through the two-wire serial interface.

Intelligent fiber modules normally support the digital diagnostic function. Through the module's internal MCU, network management units can monitor the parameters of a fiber module (temperature, voltage, bias current, TX power, and RX power) to obtain their thresholds and the module's real-time state. This helps network management units locate faults on fiber links, reduces the maintenance workload, and improves system reliability.

DDM is used for the following applications:

1. Module lifetime forecast

Monitoring the bias current makes it possible to predict the laser lifetime. By monitoring the module's voltage and temperature, the administrator can detect potential problems:

(1) A high Vcc voltage can cause CMOS breakdown; a low Vcc voltage can cause abnormal operation.

(2) A high RX power can damage the receiving module; a low RX power can prevent the receiving module from working normally.

(3) A high temperature accelerates hardware aging.

(4) Monitoring the received optical power makes it possible to monitor the performance of the link and of the remote switch.

2. Fault location

On fiber links, fault location is important for restoring services quickly. Fault isolation helps the administrator quickly determine whether a link fault is located in a module (local or remote) or on the link itself, which reduces the system recovery time.

With the digital diagnostic function, analyzing the warning and alarm status of the real-time parameters (temperature, voltage, bias current, TX power, and RX power) allows faults to be located quickly. The Tx Fault and Rx LOS states are also important for fault analysis.

3. Compatibility verification

Compatibility verification checks whether the module's operating environment complies with the module's data sheet or the applicable standard, because the module's performance can only be guaranteed in a compliant environment. If environmental parameters exceed the limits of the data sheet or the applicable standard, module performance degrades, which can cause transmission errors.

An environment is not compatible with the module in the following cases:

(1) The voltage exceeds the specified range.

(2) The RX power is overloaded or is below the receiver sensitivity.

(3) The temperature exceeds the operating temperature range.

3.10.1.2 DDM Function

DDM provides the following functions:

1. Displaying the monitoring information of the transceiver

By checking the real-time parameters (TX power, RX power, temperature, voltage, and bias current) and querying the monitoring information (such as warnings, alarms, real-time state, and thresholds), the administrator can determine the current working state of the transceiver and find potential problems. Checking the fault information of a fiber module also helps the administrator locate link faults quickly and shortens the recovery time.

2. User-defined thresholds

The real-time parameters (TX power, RX power, temperature, voltage, and bias current) have fixed thresholds. Because user environments differ, users can define their own thresholds (high alarm, low alarm, high warning, and low warning) to monitor the working state of the transceiver flexibly and detect faults directly.

The thresholds configured by the user and by the manufacturer can be displayed at the same time. If a user-defined threshold is invalid, the user is prompted, and alarms or warnings are generated automatically according to the default threshold. (The user can restore all thresholds, or a single threshold, to the default.)

Threshold validity: the high and low warning thresholds must lie between the high and low alarm thresholds, and each high threshold must be greater than the corresponding low threshold, that is, high alarm >= high warn >= low warn >= low alarm.

For fiber modules, the calibration mode of the received power is either internal calibration or external calibration, as determined by the manufacturer. The real-time parameters and the default thresholds use the same calibration mode.

3. Transceiver monitoring

In addition to checking the real-time working state of the transceiver, users may need to monitor its detailed status history, such as the time and type of the last abnormality. Transceiver monitoring lets users find past abnormal states in the log and query the last abnormal state with commands. After handling the abnormal information of a fiber module, users can clear it and restart monitoring of the module.

3.10.2 DDM Configuration Task List

DDM configuration task list:

1. Display the real-time monitoring information of the transceiver

2. Configure the alarm or warning thresholds of each transceiver parameter

3. Configure transceiver monitoring

(1) Configure the transceiver monitoring interval

(2) Enable or disable transceiver monitoring

(3) Display the transceiver monitoring information

(4) Clear the transceiver monitoring information

1. Display the real-time monitoring information of the transceiver

Command	Explanation
User Mode, Admin Mode, and Global Mode	
show transceiver [interface ethernet <interface-list>][detail]	Displays the monitoring information of the transceiver.

2. Configure the alarm or warning thresholds of each transceiver parameter

Command	Explanation
Port Mode	

transceiver threshold {default | {temperature | voltage | bias | rx-power | tx-power} {high-alarm | low-alarm | high-warn | low-warn} {<value> | default}}	Sets the user-defined threshold.

3. Configure transceiver monitoring

(1) Configure the transceiver monitoring interval

Command	Explanation
Global Mode	
transceiver-monitoring interval <minutes> **no transceiver-monitoring interval**	Sets the transceiver monitoring interval. The no command restores the default interval of 15 minutes.

(2) Enable or disable transceiver monitoring

Command	Explanation
Port Mode	
transceiver-monitoring {enable | disable}	Enables or disables transceiver monitoring on the port. The system records abnormal states only on ports where transceiver monitoring is enabled. When the function is disabled on a port, its abnormal-state information is cleared.

(3) Display the transceiver monitoring information

Command	Explanation
Admin Mode and Global Mode	
show transceiver threshold-violation [interface ethernet <interface-list>]	Displays the transceiver monitoring information, including the last threshold violation, the current monitoring interval, and whether transceiver monitoring is enabled on the port.

(4) Clear the transceiver monitoring information

Command	Explanation
Admin Mode	
clear transceiver threshold-violation [interface ethernet <interface-list>]	Clears the threshold-violation information of transceiver monitoring.

3.10.3 Examples of DDM

Example 1:

Ethernet ports 21 and 23 have fiber modules with DDM installed, Ethernet port 24 has a fiber module without DDM, and Ethernet port 22 has no fiber module. Display the DDM information of the fiber modules.

a. Display the information of all interfaces whose real-time parameters can be read normally (interfaces with no fiber module or with an unsupported fiber module are not shown). For example:

```
Switch#show transceiver
Interface  Temp (℃)  Voltage (V)  Bias (mA)   RX Power (dBM)     TX
Power (dBM)
1/0/21     33         3.31         6.11        -30.54(A-)         -6.01
1/0/23     33         5.00 (W+)   6.11        -20.54(W-)         -6.02
```

b. Display the information of the specified interfaces (N/A means that no fiber module is installed or the module is not supported). For example:

```
Switch#show transceiver interface ethernet 1/0/21-22;23
Interface  Temp (℃)  Voltage (V)  Bias (mA)   RX Power (dBM)    TX Power (dBM)
1/0/21     33         3.31         6.11        -30.54(A-)         -6.01
1/0/22     N/A        N/A          N/A         N/A                N/A
1/0/23     33         5.00 (W+)    6.11        -20.54(W-)         -6.02
```

c. Display the detailed information, including basic information, real-time monitoring values, warnings, alarms, abnormal states, threshold information, and the serial number. For example:

```
Switch#show transceiver interface ethernet 1/0/21-22;24 detail
Ethernet 1/0/21 transceiver detail information:
Base information:
SFP found in this port, manufactured by company, on Sep 29 2010.
Type is 1000BASE-SX. Serial Number is 1108000001.
Link length is 550 m for 50um Multi-Mode Fiber.
Link length is 270 m for 62.5um Multi-Mode Fiber.
Nominal bit rate is 1300 Mb/s, Laser wavelength is 850 nm.
Brief alarm information:
RX loss of signal
Voltage high
RX power low
Detail diagnostic and threshold information:
                  Diagnostic                          Threshold
Realtime Value   High Alarm  Low Alarm    High Warn     Low Warn
--------------      -----------    -----------      ------------
---------
Temperature (℃)    33            70          0            70          0
Voltage (V)         7.31(A+)       5.00        0.00         5.00
0.00
Bias current (mA)   6.11(W+)       10.30       0.00         5.00
0.00
RX Power (dBM)     -30.54(A-)      9.00        -25.00       9.00          -
25.00
TX Power (dBM)     -6.01           9.00        -25.00       9.00          -
25.00
Ethernet 1/0/22 transceiver detail information: N/A
Ethernet 1/0/24 transceiver detail information:
Base information:
SFP found in this port, manufactured by company, on Sep 29 2010.
Type is 1000BASE-SX. Serial Number is 1108000001.
Link length is 550 m for 50um Multi-Mode Fiber.
Link length is 270 m for 62.5um Multi-Mode Fiber.
Nominal bit rate is 1300 Mb/s, Laser wavelength is 850 nm.
Brief alarm information: N/A
Detail diagnostic and threshold information: N/A
```

Explanation: If the serial number is 0, the output shows that it is not specified, as below:

```
SFP found in this port, manufactured by company, on Sep 29 2010.
Type is 1000BASE-SX. Serial Number is not specified.
Link length is 550 m for 50um Multi-Mode Fiber.
Link length is 270 m for 62.5um Multi-Mode Fiber.
Nominal bit rate is 1300 Mb/s, Laser wavelength is 850 nm.
```

Example 2:

Ethernet port 21 has a fiber module with DDM installed. Display the DDM information, and then configure the thresholds of the fiber module.

Step 1: Display the detailed DDM information.

```
Switch#show transceiver interface ethernet 1/0/21 detail
Ethernet 1/0/21 transceiver detail information:
Base information:
……
Brief alarm information:
```

```
RX loss of signal
Voltage high
RX power low
Detail diagnostic and threshold information:
                    Diagnostic                          Threshold
Realtime Value   High Alarm  Low Alarm   High Warn   Low Warn
--------------      ----------    ----------     -----------
---------
Temperature (℃)    33              70         0           70        0
Voltage (V)         7.31(A+)         5.00       0.00        5.00       0.00
Bias current (mA)   6.11(W+)         10.30      0.00        5.00       0.00
RX Power (dBM)    -30.54(A-)        9.00      -25.00       9.00       -25.00
TX Power (dBM)    -13.01            9.00      -25.00       9.00       -25.00
```

Step 2: Configure the TX power thresholds of the fiber module: set the low-warning threshold to -12 and the low-alarm threshold to -10.00.

```
Switch#config
Switch(config)#interface ethernet 1/0/21
Switch(config-if-ethernet1/0/21)#transceiver threshold tx-power low-warning -
12
Switch(config-if-ethernet1/0/21)#transceiver threshold tx-power low-alarm -
10.00
```

Step 3: Display the detailed DDM information of the fiber module. Alarms are now based on the user-configured thresholds; the manufacturer's thresholds are shown in parentheses. An "A-" alarm is raised because -13.01 is less than -12.00.

```
Switch#show transceiver interface ethernet 1/0/21 detail
Ethernet 1/0/21 transceiver detail information:
Base information:
……
Brief alarm information:
RX loss of signal
Voltage high
RX power low
TX power low
Detail diagnostic and threshold information:
                    Diagnostic                          Threshold
Realtime Value   High Alarm  Low Alarm   High Warn   Low Warn
--------------      ----------    ----------     ----------     ---------
Temperature (℃)   33             70          0           70          0
Voltage (V)        7.31(A+)        5.00        0.00        5.00        0.00
Bias current (mA)  6.11(W+)       10.30       0.00        5.00        0.00
RX Power (dBM)   -30.54(A-)      9.00       -25.00       9.00        -25.00
TX Power (dBM)   -13.01(A-)      9.00      -12.00(-25.00)  9.00    -10.00(-
25.00)
```

Example 3:

Ethernet port 21 has a fiber module with DDM installed. Display the transceiver monitoring information of the fiber module, and then enable transceiver monitoring on the port.

Step 1: Display the transceiver monitoring information of the fiber modules. Transceiver monitoring is disabled on both Ethernet ports 21 and 22, and the monitoring interval is set to 30 minutes.

```
Switch(config)#show transceiver threshold-violation interface ethernet
1/0/21-22
Ethernet 1/0/21 transceiver threshold-violation information:
Transceiver monitor is disabled. Monitor interval is set to 30 minutes.
The last threshold-violation doesn’t exist.
Ethernet 1/0/22 transceiver threshold-violation information:
Transceiver monitor is disabled. Monitor interval is set to 30 minutes.
The last threshold-violation doesn’t exist.
```

Step 2: Enable transceiver monitoring on Ethernet port 21.

```
Switch(config)#interface ethernet 1/0/21
Switch(config-if-ethernet1/0/21)#transceiver-monitoring enable
```

Step 3: Display the transceiver monitoring information of the fiber modules. In the output below, transceiver monitoring is enabled on Ethernet port 21, the last threshold violation occurred at Jan 02 11:00:50 2011, and the detailed DDM information that exceeded the thresholds is also shown.

```
Switch(config-if-ethernet1/0/21)#quit
Switch(config)#show transceiver threshold-violation interface ethernet
1/0/21-22
Ethernet 1/0/21 transceiver threshold-violation information:
Transceiver monitor is enabled. Monitor interval is set to 30 minutes.
The current time is Jan 02 12:30:50 2011.
The last threshold-violation time is Jan 02 11:00:50 2011.
Brief alarm information:
RX loss of signal
RX power low
Detail diagnostic and threshold information:
                    Diagnostic                              Threshold
Realtime Value  High Alarm      Low Alarm     High Warn      Low Warn
------------      -----------        -----------       ------------
---------
Temperature (℃)   33        70          0             70             0
Voltage (V)        7.31      10.00        0.00          5.00           0.00
Bias current (mA)  3.11      10.30        0.00          5.00           0.00
RX Power (dBM)  -30.54(A-)  9.00        -25.00(-34)     9.00          -25.00
TX Power (dBM)   -1.01      9.00        -12.05         9.00          -10.00
Ethernet 1/0/22 transceiver threshold-violation information:
Transceiver monitor is disabled. Monitor interval is set to 30 minutes.
The last threshold-violation doesn’t exist.
```

3.10.4 DDM Troubleshooting

If problems occur when configuring DDM, check whether they are caused by the following:

- Ensure that the fiber module transceiver is firmly inserted in the port; otherwise, the DDM information is not displayed.
- Ensure that the SNMP configuration is valid; otherwise, warning events cannot be reported to the network management system.
- Because only some boards and switches support SFP or XFP modules with DDM, ensure that the board and switch in use support this function.
- The **show transceiver** and **show transceiver detail** commands take a long time to run because the switch checks all ports. It is recommended to query the transceiver monitoring information of specific ports instead.
- Ensure that the user-defined thresholds are valid. If any threshold is invalid, the transceiver generates alarms automatically according to the default settings.

3.11 EFM OAM

3.11.1 Introduction to EFM OAM

Ethernet was originally designed for Local Area Networks, but as link length and network scope expanded rapidly, Ethernet also came to be applied to Metropolitan Area Networks and Wide Area Networks. Because it lacked an effective management mechanism for these larger scopes, implementing OAM on Ethernet became a necessary development trend.

There are four protocol standards for Ethernet OAM: 802.3ah (EFM OAM), 802.3ag (CFM), E-LMI, and Y.1731. EFM OAM and CFM are defined by the IEEE. EFM OAM works at the data link layer to

effectively discover and manage the status of the underlying data link. Using EFM OAM can effectively advance the management and maintenance of Ethernet to ensure stable network operation. CFM is used to monitor connectivity across the whole network and to locate faults in the access aggregation network layer. Compared with CFM, the Y.1731 standard defined by the ITU (International Telecommunications Union) is more powerful. The E-LMI standard defined by the MEF applies only to UNIs. These protocols can therefore be used for different network topologies and management needs, and they complement one another.

EFM OAM (Ethernet in the First Mile Operation, Administration and Maintenance) works at the data link layer of the OSI model to implement its functions through an OAM sublayer, as shown in the figure below:

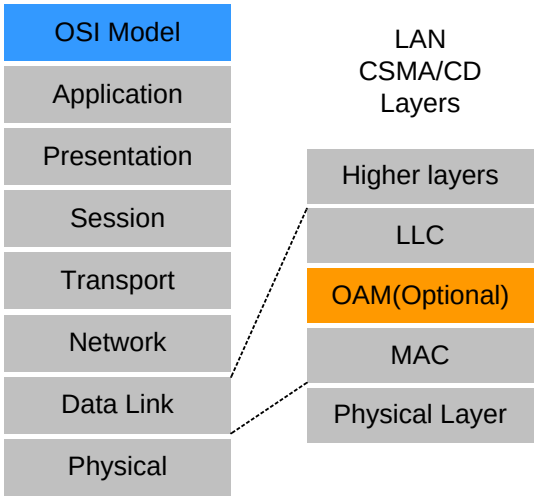


Figure 3-14 OAM Location in the OSI Model

OAM protocol data units (OAMPDUs) use the protocol destination MAC address 01-80-c2-00-00-02, and the maximum transmission rate is 10 Pkt/s.

EFM OAM is built on the basis of an OAM connection. It provides link operation and management mechanisms such as link monitoring, remote fault detection, and remote loopback testing, briefly introduced below:

1. Ethernet OAM connection establishment

An Ethernet OAM entity discovers remote OAM entities and establishes sessions with them by exchanging Information OAMPDUs. EFM OAM can operate in two modes: active mode and passive mode. A session can only be initiated by an OAM entity working in active mode; entities working in passive mode wait until they receive a connection request. After an Ethernet OAM connection is established, the Ethernet OAM entities on both sides continuously exchange Information OAMPDUs to keep the connection valid. If an Ethernet OAM entity receives no Information OAMPDU for five seconds, the Ethernet OAM connection is disconnected.

2. Link monitoring

Fault detection in Ethernet is difficult, especially when the physical connection in the network is not disconnected but network performance is gradually degrading. Link monitoring is used to detect and discover link faults in various environments. EFM OAM implements link monitoring through the exchange of Event Notification OAMPDUs. When a link error event is detected, the local OAM entity sends an Event Notification OAMPDU to notify the remote OAM entity. At the same time, it logs the information and sends an SNMP trap to the network management system. When the OAM entity on the other side receives the notification, it also logs and reports it. With this log information, network administrators can keep track of network status in a timely manner.

The link events monitored by EFM OAM refer to link error events, including the Errored Symbol Period event, Errored Frame event, Errored Frame Period event, and Errored Frame Seconds event.

Errored Symbol Period event: The number of errored symbols reaches or exceeds the low threshold. (Symbol: the minimum data transmission unit of the physical medium. It is specific to the coding system, so symbols may differ across physical mediums; the symbol rate is the number of electrical state changes per second.)

Errored Frame Period event: Given a period of N frames, the number of errored frames received within that period reaches or exceeds the low threshold. (Errored frame: a frame detected as errored by CRC.)

Errored Frame event: The number of errored frames detected over M seconds reaches or exceeds the low threshold.

Errored Frame Seconds event: The number of errored-frame seconds detected over M seconds reaches or exceeds the low threshold. (An errored-frame second is a one-second interval in which at least one errored frame was received.)

3. Remote Fault Detection

In a network where traffic is interrupted due to device failures or unavailability, the flag field defined in Ethernet OAMPDUs allows an Ethernet OAM entity to send fault information to its peer. Because Information OAMPDUs are exchanged continuously across established OAM connections, an Ethernet OAM entity can inform its OAM peer of link faults through Information OAMPDUs. This allows network administrators to keep track of link status in a timely manner through the log information and to troubleshoot promptly.

There are three kinds of link faults reported via Information OAMPDU: Critical Event, Dying Gasp, and Link Fault. Their exact definitions vary by manufacturer; here the definitions are as follows:

Critical Event: the EFM OAM function on the port is disabled.

Link Fault: the number of unidirectional-operation faults reaches or exceeds the high threshold locally. Unidirectional Operation means a unidirectional link cannot work normally on a full-duplex link without auto-negotiation. EFM OAM can detect the fault and inform remote OAM peers by sending an Information OAMPDU.

Dying Gasp: there is no standard definition for this. Although a device does not generate a Dying Gasp OAMPDU itself, it still receives and processes such an OAMPDU sent by its peer.

4. Remote loopback testing

Remote loopback testing is available only after an Ethernet OAM connection is established. With remote loopback enabled, the Ethernet OAM entity operating in active mode issues remote loopback requests and the peer responds to them. If the peer is operating in loopback mode, it returns all packets except Ethernet OAMPDUs to the sender along the original path. Performing remote loopback testing periodically helps detect network faults in a timely manner. Performing it segment by segment also helps locate network faults. Note: normal communication is not possible while in remote loopback mode.

A typical EFM OAM application topology is shown below; it is used for point-to-point links and emulated IEEE 802.3 point-to-point links. The device enables EFM OAM over a point-to-point connection to monitor link faults in the First Mile of Ethernet access. From the user's perspective, the connection between the user and the telecommunications provider is the "First Mile"; from the service provider's perspective, it is the "Last Mile".

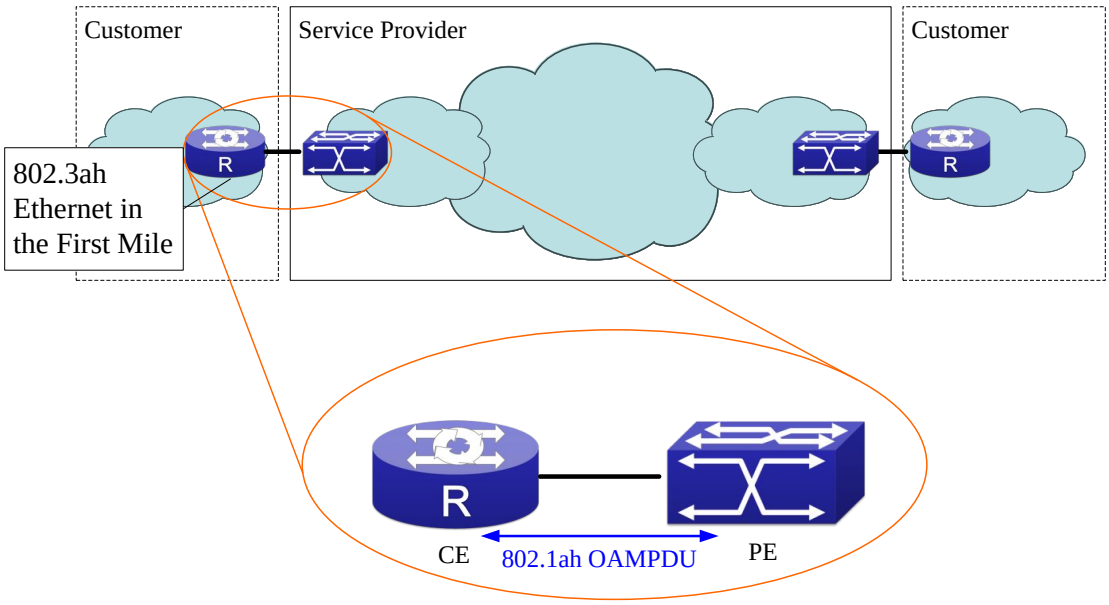


Figure 3-15 Typical OAM Application Topology

3.11.2 EFM OAM Configuration

EFM OAM configuration task list:

1. Enable the EFM OAM function on the port
2. Configure link monitoring
3. Configure remote fault detection
4. Enable EFM OAM loopback on the port

Note: OAM must be enabled first before configuring OAM parameters.

1. Enable the EFM OAM function on the port

Command	Explanation
Port Mode	
ethernet-oam mode {active | passive}	Configures the EFM OAM work mode. The default is active mode.
ethernet-oam **no ethernet-oam**	Enables EFM OAM on the port. The no command disables EFM OAM on the port.
ethernet-oam period <seconds> **no ethernet-oam period**	Configures the OAMPDU transmission period (optional). The no command restores the default value.
ethernet-oam timeout <seconds> **no ethernet-oam timeout**	Configures the EFM OAM connection timeout. The no command restores the default value.

2. Configure link monitoring

Command	Explanation
Port Mode	
ethernet-oam link-monitor **no ethernet-oam link-monitor**	Enables EFM OAM link monitoring. The no command disables link monitoring.

ethernet-oam errored-symbol-period {threshold low <low-symbols> | window <seconds>} **no ethernet-oam errored-symbol-period {threshold low | window }**	Configures the low threshold and window period of the errored symbol period event (optional). The no command restores the default value.
ethernet-oam errored-frame-period {threshold low <low-frames> | window <seconds>} **no ethernet-oam errored-frame-period {threshold low | window }**	Configures the low threshold and window period of the errored frame period event. The no command restores the default value.
ethernet-oam errored-frame {threshold low <low-frames> | window <seconds>} **no ethernet-oam errored-frame {threshold low | window }**	Configures the low threshold and window period of the errored frame event (optional). The no command restores the default value.
ethernet-oam errored-frame-seconds {threshold low <low-frame-seconds> | window <seconds>} **no ethernet-oam errored-frame-seconds {threshold low | window }**	Configures the low threshold and window period of the errored frame seconds event (optional). The no command restores the default value.

3. Configure remote fault detection

Command	Explanation
Port Mode	
ethernet-oam remote-failure **no ethernet-oam remote-failure**	Enables EFM OAM remote failure detection (failure here means a local critical-event or link-fault event) (optional). The no command disables the function.
ethernet-oam errored-symbol-period threshold high {high-symbols | none} **no ethernet-oam errored-symbol-period threshold high**	Configures the high threshold of the errored symbol period event (optional). The no command restores the default value.
ethernet-oam errored-frame-period threshold high {high-frames | none} **no ethernet-oam errored-frame-period threshold high**	Configures the high threshold of the errored frame period event (optional). The no command restores the default value.
ethernet-oam errored-frame threshold high {high-frames | none} **no ethernet-oam errored-frame threshold high**	Configures the high threshold of the errored frame event (optional). The no command restores the default value.
ethernet-oam errored-frame-seconds threshold high {high-frame-seconds | none} **no ethernet-oam errored-frame-seconds threshold high**	Configures the high threshold of the errored frame seconds event (optional). The no command restores the default value.

4. Enable EFM OAM loopback on the port

Command	Explanation
Port Mode	
ethernet-oam remote-loopback **no ethernet-oam remote-loopback**	Enables the remote EFM OAM entity to enter OAM loopback mode (the peer must be configured to support OAM loopback). The no command cancels remote OAM loopback.
ethernet-oam remote-loopback supported **no ethernet-oam remote-loopback supported**	Enables remote loopback support on the port. The no command cancels remote loopback support on the port.

3.11.3 EFM OAM Example

Example:

The CE and PE devices on a point-to-point link enable EFM OAM to monitor the performance of the "First Mile" link. When a fault event occurs, they report log information to the network management system, and remote loopback can be used to test the link when necessary.

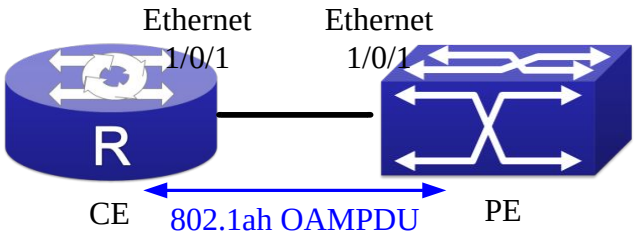


Figure 3-16 Typical OAM Application Topology

Configuration procedure (SNMP and log configuration are omitted below):

```
Configuration on CE:
 CE(config)#interface ethernet1/0/1
CE (config-if-ethernet1/0/1)#ethernet-oam mode passive
CE (config-if-ethernet1/0/1)#ethernet-oam
CE (config-if-ethernet1/0/1)#ethernet-oam remote-loopback supported
```

All other parameters use the default configuration.

```
Configuration on PE:
PE(config)#interface ethernet 1/0/1
PE (config-if-ethernet1/0/1)#ethernet-oam
```

All other parameters use the default configuration.

Run the following command to start remote loopback:

```
PE(config-if-ethernet1/0/1)#ethernet-oam remote-loopback
```

Run the following command to make the OAM peer exit OAM loopback after the test is complete:

```
PE(config-if-ethernet1/0/1)# no ethernet-oam remote-loopback
```

Run the following command to disable remote loopback support:

```
CE(config-if-ethernet1/0/1)#no ethernet-oam remote-loopback supported
```

3.11.4 EFM OAM Troubleshooting

If problems occur when using EFM OAM, check whether they are caused by the following:

- ☞ Check whether the OAM entities at both ends of the link are in passive mode. If so, an EFM OAM connection cannot be established between them.
- ☞ Ensure that the SNMP configuration is correct; otherwise, errored events cannot be reported to the network management system.
- ☞ The link cannot communicate normally in OAM loopback mode. Cancel remote loopback promptly after testing the link performance.
- ☞ Ensure that the board in use supports the remote loopback function.
- ☞ After OAM loopback is enabled on a port, do not configure STP, MRPP, ULPP, flow control, or loopback detection on that port, because OAM remote loopback and these functions are mutually exclusive.
- ☞ When OAM is enabled, auto-negotiation on the port is disabled automatically. Auto-negotiation must therefore also be disabled on the peer port; otherwise, the link cannot come up. When OAM is disabled, auto-negotiation on the port is restored. To ensure a normal link connection, the negotiation settings at both ends of the link must match.

☞ After OAM is enabled and negotiation succeeds at both ends of the link, the link state is up. If the fiber on the peer's RX side is then pulled out, the peer's TX and the RX of the OAM-enabled port remain normal, so the OAM-enabled port stays up.

3.12 Port Security

3.12.1 Introduction to Port Security

Port security is a MAC address-based security mechanism for network access control. It extends the existing 802.1X authentication and MAC authentication. It prevents unauthorized devices from accessing the network by checking the source MAC address of received frames, and prevents access to unauthorized devices by checking the destination MAC address of sent frames. With port security, you can define port security modes so that the device learns only legal source MAC addresses, implementing the corresponding network security management. After port security is enabled, when the device detects an illegal frame, it triggers the corresponding port security feature and takes a predefined action automatically. This reduces the maintenance workload and greatly enhances system security.

3.12.2 Port Security Configuration Task List

1. Basic port security configuration

Command	Explanation
Port Mode	
switchport port-security **no switchport port-security**	Enables port security on the interface. The no command disables port security.
switchport port-security mac-address <*mac-address*> [vlan <*vlan-id*>] **no switchport port-security mac-address <*mac-address*> [vlan <*vlan-id*>]**	Configures a static secure MAC address on the interface. The no command deletes the static secure MAC address.
switchport port-security maximum <*value*> [vlan <*vlan-list*>] **no switchport port-security maximum <*value*> [vlan <*vlan-list*>]**	Configures the maximum number of secure MAC addresses allowed on the interface.
switchport port-security violation {protect | recovery | restrict | shutdown} **no switchport port-security violation**	Configures the action taken when a port security violation occurs. A violation occurs when the number of MAC addresses exceeds the configured maximum, when a MAC address accessing the interface belongs to another interface in the MAC address table, or when a MAC address is configured on several interfaces in the same VLAN.
switchport port-security aging {static | time <*value*> | type {absolute | inactivity}} **no switchport port-security violation aging {static | time | type}**	Enables aging of port-security entries on the interface and specifies the aging time or aging type.
Admin Mode	
clear port-security {all | configured | dynamic | sticky} [[address <*mac-addr*> | interface <*interface-id*>] [vlan <*vlan-id*>]]	Clears secure MAC entries on the interface.
show port-security [interface <*interface-id*>]	Displays the port-security

[address | vlan]	configuration.

3.12.3 Port Security Example

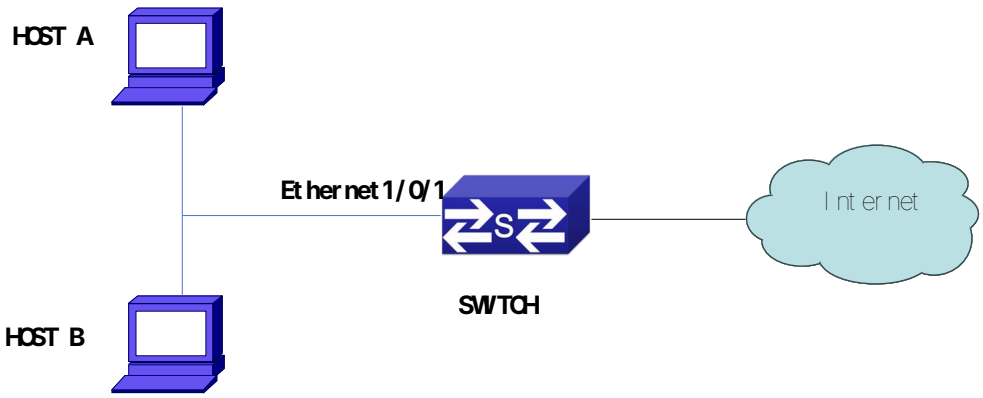


Figure 3-17 Typical Port Security Topology

When port security is enabled on an interface and the maximum number of secure MAC addresses allowed on the interface is set to 10, at most 10 users can access the Internet through the interface. When the maximum is reached, new users cannot access the Internet. This both limits the number of users and secures Internet access. If the maximum number of secure MAC addresses is set to 1, only HOST A or HOST B can access the Internet.

Configuration process:

Configure the switch.

```
Switch(config)#interface Ethernet 1/0/1
Switch(config-if-ethernet1/0/1)#switchport port-security
Switch(config-if- ethernet1/0/1)#switchport port-security maximum 10
Switch(config-if- ethernet1/0/1)#exit
Switch(config)#
```

3.12.4 Port Security Troubleshooting

If problems occur when configuring port security, check whether they are caused by the following:

- ☞ Check whether port security is enabled correctly.
- ☞ Check whether a valid maximum number of MAC addresses is configured.

3.13 VLAN

3.13.1 Introduction to VLAN

VLAN (Virtual Local Area Network) is a technology that divides the devices in a network into separate logical network segments based on function, application, or management requirements. This allows virtual workgroups to be formed regardless of the physical location of the devices. The IEEE published the IEEE 802.1Q protocol to standardize VLAN implementation, and the VLAN function of the switch follows IEEE 802.1Q.

The key idea of VLAN technology is that a large LAN can be dynamically partitioned into many separate broadcast domains as required.

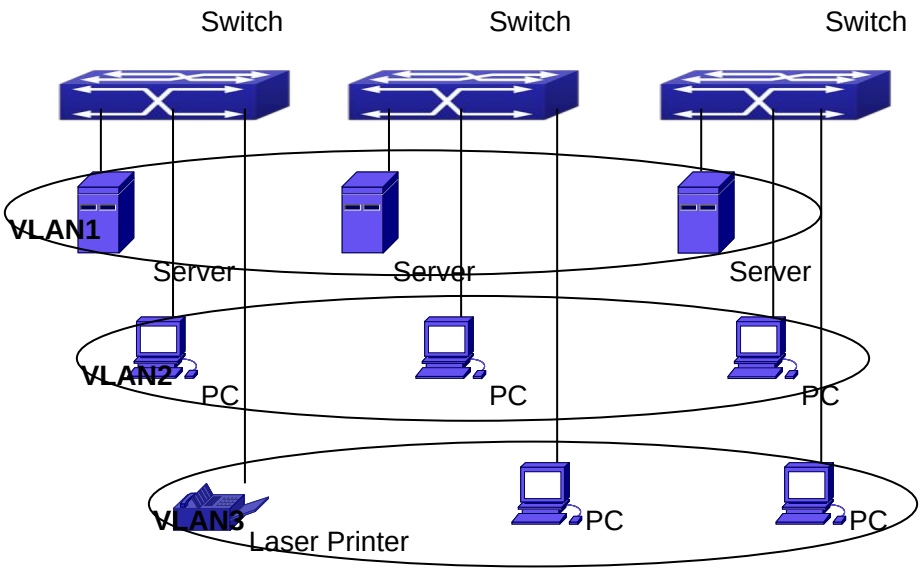


Figure 3-18 Logically Defined VLAN Network

Each broadcast domain is a VLAN. VLANs have the same properties as physical LANs, except that a VLAN is a logical rather than a physical partition. Therefore, VLANs can be partitioned regardless of physical location, and the broadcast, multicast, and unicast traffic within a VLAN is separated from other VLANs.

With these features, VLAN technology provides the following benefits:

- Improved network performance
- Saved network resources
- Simplified network management
- Lower network cost
- Enhanced network security

Switch Ethernet ports can work in three modes: Access, Hybrid, and Trunk. Each mode handles tagged and untagged packets differently when forwarding.

An Access port belongs to only one VLAN and is usually used to connect to a computer.

A Trunk port allows multiple VLANs to pass and can receive and send packets of multiple VLANs. It is usually used to connect switches.

A Hybrid port allows multiple VLANs to pass and can receive and send packets of multiple VLANs. It can be used to connect switches or to connect to a user's computer.

Hybrid ports and Trunk ports receive data in the same way but send data differently: a Hybrid port can send packets of multiple VLANs without a VLAN tag, while a Trunk port sends packets of all VLANs with a VLAN tag, except packets of the port's native VLAN.

The switch implements VLAN and GVRP (GARP VLAN Registration Protocol) as defined in 802.1Q. This chapter describes the use and configuration of VLAN and GVRP in detail.

3.13.2 VLAN Configuration Task List

1. Create or delete a VLAN
2. Set or delete a VLAN name
3. Assign switch ports to a VLAN
4. Set the switch port type

5. Configure a Trunk port
6. Configure an Access port
7. Configure a Hybrid port
8. Enable or disable VLAN ingress rules on ports
9. Configure a private VLAN
10. Configure private VLAN association
11. Specify the internal VLAN ID

1. Create or delete a VLAN

Command	Explanation
Global Mode	
vlan WORD **no vlan WORD**	Creates or deletes a VLAN, or enters VLAN Mode.

2. Set or delete a VLAN name

Command	Explanation
VLAN Mode	
name <vlan-name> **no name**	Sets or deletes the VLAN name.

3. Assign switch ports to a VLAN

Command	Explanation
VLAN Mode	
switchport interface <*interface-list*> **no switchport interface <*interface-list*>**	Assigns switch ports to the VLAN. The no command removes the ports from the VLAN.

4. Set the switch port type

Command	Explanation
Port Mode	
switchport mode {trunk | access | **hybrid}**	Sets the current port as a Trunk, Access, or Hybrid port.

5. Configure a Trunk port

Command	Explanation
Port Mode	
switchport trunk allowed vlan {WORD | all | add WORD | except WORD | remove WORD} **no switchport trunk allowed vlan**	Sets or deletes the VLANs allowed to pass through the Trunk port. The no command restores the default setting.
switchport trunk native vlan <*vlan-id*> **no switchport trunk native vlan**	Sets or deletes the PVID of the Trunk port.

6. Configure an Access port

Command	Explanation
Port Mode	
switchport access vlan <*vlan-id*> **no switchport access vlan**	Adds the current port to the specified VLAN. The no command restores the default setting.

7. Configure a Hybrid port

Command	Explanation
Port Mode	

switchport hybrid allowed vlan {WORD | all | add WORD | except WORD | remove WORD} {tag | untag} **no switchport hybrid allowed vlan**	Sets or deletes the VLANs allowed to pass through the Hybrid port in tagged or untagged mode.
switchport hybrid native vlan <*vlan-id*> **no switchport hybrid native vlan**	Sets or deletes the PVID of the port.

8. Enable or disable VLAN ingress rules

Command	Explanation
Port Mode	
vlan ingress enable **no vlan ingress enable**	Enables VLAN ingress rules. The no command disables them.

9. Configure a private VLAN

Command	Explanation
VLAN Mode	
private-vlan {primary | isolated | community} **no private-vlan**	Configures the current VLAN as a private VLAN. The no command deletes the private VLAN.

10. Configure private VLAN association

Command	Explanation
VLAN Mode	
private-vlan association <secondary-vlan-list> **no private-vlan association**	Sets or deletes the private VLAN association.

11. Specify the internal VLAN ID

Command	Explanation
Global Mode	
vlan <2-4094> internal	Specifies the internal VLAN ID.

3.13.3 Typical VLAN Application

Scenario:

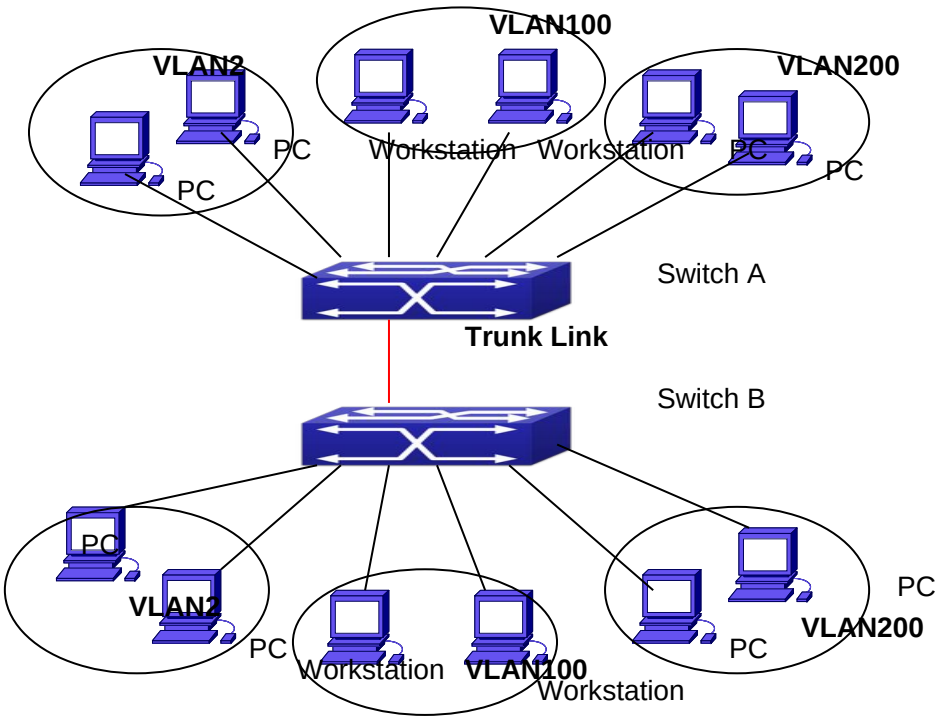


Figure 3-19 Typical VLAN Application Topology

Due to security and application requirements, the existing LAN must be partitioned into three VLANs: VLAN2, VLAN100, and VLAN200. These VLANs span two different sites, A and B. One switch is placed at each site, and the cross-site requirement is met by transferring VLAN traffic between the two switches.

Configuration Item	Configuration Description
VLAN2	Ports 2-4 of the switches at site A and site B.
VLAN100	Ports 5-7 of the switches at site A and site B.
VLAN200	Ports 8-10 of the switches at site A and site B.
Trunk port	Port 11 of the switches at site A and site B.

Connect the Trunk ports of the two switches to form a Trunk link that carries the cross-switch VLAN traffic, and connect all network devices to the other ports of the corresponding VLANs.

In this example, ports 1 and 12 are unused and can be used as management ports or for other purposes.

The configuration steps are as follows:

```
Switch A:
Switch(config)#vlan 2
Switch(Config-Vlan2)#switchport interface ethernet 1/0/2-4
Switch (Config-Vlan2)#exit
Switch (config)#vlan 100
Switch (Config-Vlan100)#switchport interface ethernet 1/0/5-7
Switch (Config-Vlan100)#exit
Switch (config)#vlan 200
Switch (Config-Vlan200)#switchport interface ethernet 1/0/8-10
Switch (Config-Vlan200)#exit
Switch (config)#interface ethernet 1/0/11
Switch (Config-If-Ethernet1/0/11)#switchport mode trunk
Switch(Config-If-Ethernet1/0/11)#exit
Switch(config)#
Switch B:
Switch(config)#vlan 2
Switch(Config-Vlan2)#switchport interface ethernet 1/0/2-4
```

```
Switch (Config-Vlan2)#exit
Switch (config)#vlan 100
Switch (Config-Vlan100)#switchport interface ethernet 1/0/5-7
Switch (Config-Vlan100)#exit
Switch (config)#vlan 200
Switch (Config-Vlan200)#switchport interface ethernet 1/0/8-10
Switch (Config-Vlan200)#exit
Switch (config)#interface ethernet 1/0/11
Switch (Config-If-Ethernet1/0/11)#switchport mode trunk
Switch (Config-If-Ethernet1/0/11)#exit
```

3.13.4 Typical Hybrid Port Application

Scenario:

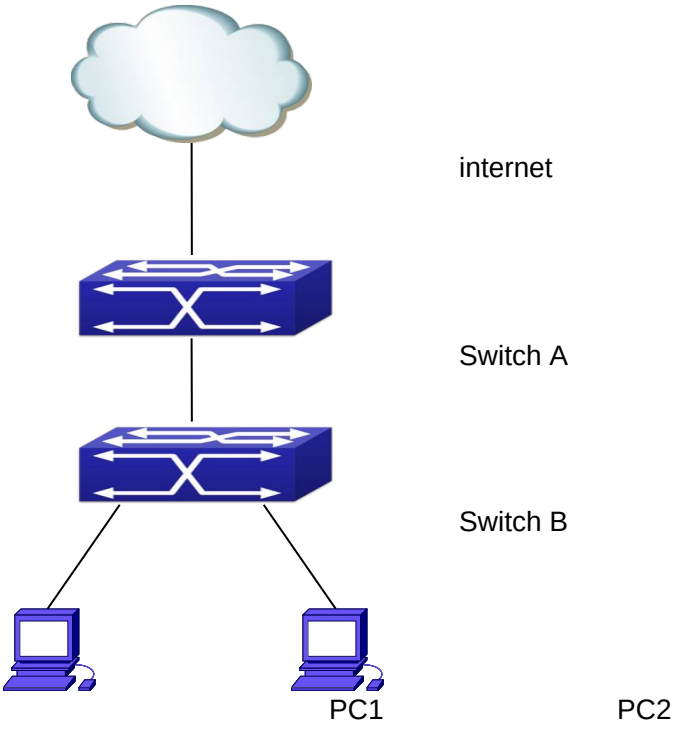


Figure 3-13 Typical Hybrid Port Application

PC1 connects to interface Ethernet 1/0/7 of SwitchB, PC2 connects to interface Ethernet 1/0/9 of SwitchB, and Ethernet 1/0/10 of SwitchA connects to Ethernet 1/0/10 of SwitchB.

For security reasons, PC1 and PC2 must not be able to access each other, but both must be able to access other network resources through the gateway SwitchA. This can be achieved with Hybrid ports.

The configuration items are as follows:

Port	Type	PVID	Allowed VLANs
Port 1/0/10 of Switch A	Access	10	Allows packets of VLAN 10 to pass untagged.
Port 1/0/10 of Switch B	Hybrid	10	Allows packets of VLANs 7, 9, and 10 to pass untagged.
Port 1/0/7 of Switch B	Hybrid	7	Allows packets of VLANs 7 and 10 to pass untagged.
Port 1/0/9 of Switch B	Hybrid	9	Allows packets of VLANs 9 and 10 to pass untagged.

The configuration steps are as follows:

```
Switch A:
Switch(config)#vlan 10
Switch(Config-Vlan10)#switchport interface ethernet 1/0/10
Switch B:
Switch(config)#vlan 7;9;10
Switch(config)#interface ethernet 1/0/7
Switch(Config-If-Ethernet1/0/7)#switchport mode hybrid
Switch(Config-If-Ethernet1/0/7)#switchport hybrid native vlan 7
Switch(Config-If-Ethernet1/0/7)#switchport hybrid allowed vlan 7;10 untag
Switch(Config-If-Ethernet1/0/7)#exit
Switch(Config)#interface Ethernet 1/0/9
Switch(Config-If-Ethernet1/0/9)#switchport mode hybrid
Switch(Config-If-Ethernet1/0/9)#switchport hybrid native vlan 9
Switch(Config-If-Ethernet1/0/9)#switchport hybrid allowed vlan 9;10 untag
Switch(Config-If-Ethernet1/0/9)#exit
Switch(Config)#interface Ethernet 1/0/10
Switch(Config-If-Ethernet1/0/10)#switchport mode hybrid
Switch(Config-If-Ethernet1/0/10)#switchport hybrid native vlan 10
Switch(Config-If-Ethernet1/0/10)#switchport hybrid allowed vlan 7;9;10 untag
Switch(Config-If-Ethernet1/0/10)#exit
```

3.14 GVRP

3.14.1 Introduction to GVRP

GVRP (GARP VLAN Registration Protocol) is an application of GARP (Generic Attribute Registration Protocol). GARP mainly provides an attribute transmission mechanism that allows protocol entities to register and deregister attributes. Depending on the attributes transmitted, GARP has several application protocols, such as GMRP and GVRP. GVRP is the protocol that propagates VLAN attributes throughout the layer-2 network through GARP.

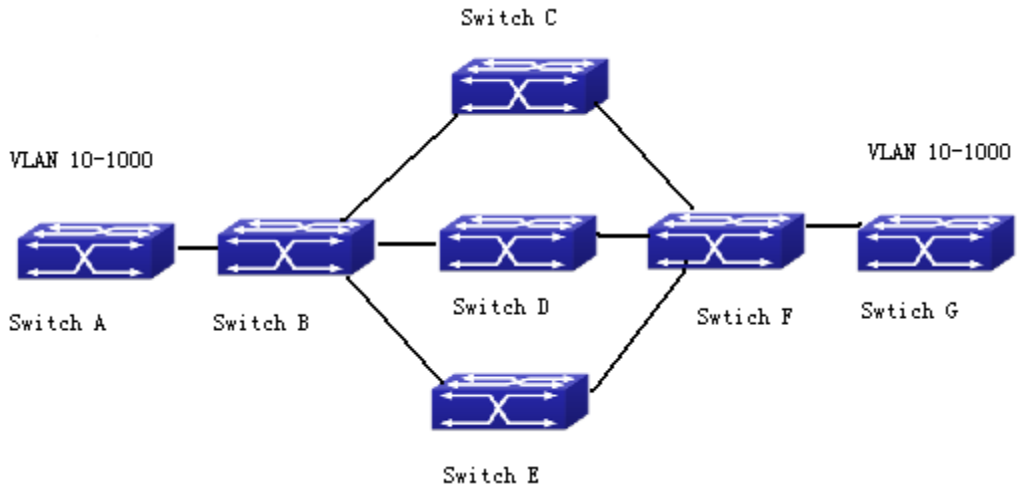


Figure 3-14 A Typical Application Scenario

Switches A and G are not directly connected in the layer-2 network; switches B, C, D, E, and F are intermediate switches connecting A and G. VLANs 100-1000 are configured manually on switches A and G, but not on switches B through F. When GVRP is not enabled, A and G cannot communicate with each other because the intermediate switches do not have the relevant VLANs. After GVRP is enabled on all switches, its VLAN attribute transmission mechanism allows the intermediate switches to register the VLANs dynamically, so that VLANs 100-1000 on A and G can communicate with each other. When VLANs 100-1000 are manually deleted on switches A and G, the VLANs dynamically registered on the intermediate switches are deregistered. GVRP thus allows the same VLAN on two non-adjacent switches to communicate without configuring each intermediate switch manually, which simplifies VLAN configuration.

3.14.2 GVRP Configuration Task List

GVRP configuration task list:

1. Configure the GVRP timers
2. Configure the port type
3. Enable the GVRP function

1. Configure the GVRP timers

Command	Explanation
Global Mode	
garp timer join <200-500> **garp timer leave <500-1200>** **garp timer leaveall <5000-60000>** **no garp timer (join | leave | leaveAll)**	Configures the GVRP leaveall, join, and leave timers. The no command restores the default value of the specified timer.

2. Configure the port type

Command	Explanation
Port Mode	
gvrp **no gvrp**	Enables GVRP on the port. The no command disables GVRP on the port.

3. Enable the GVRP function

Command	Explanation
Global Mode	
gvrp **no gvrp**	Enables GVRP globally. The no command disables GVRP globally.

3.14.3 GVRP Example

GVRP application:

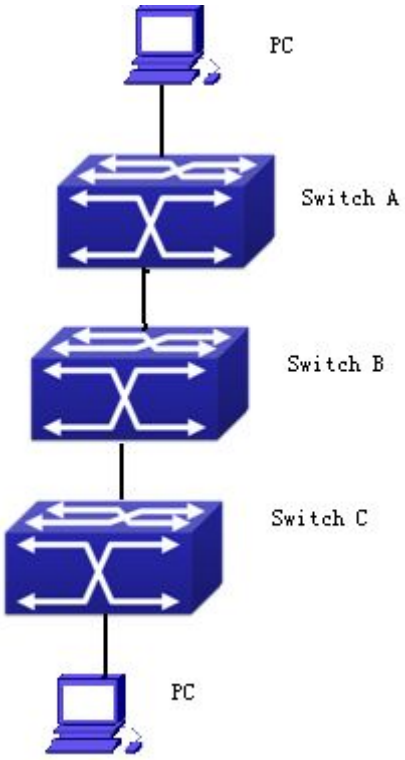


Figure 3-15 Typical GVRP Application Topology

To enable dynamic registration and updating of VLAN information among switches, configure GVRP on the switches. Configure GVRP on switches A, B, and C so that switch B learns VLAN100 dynamically. The two workstations connected to VLAN100 on switches A and C can then communicate with each other through switch B without static VLAN100 entries on switch B.

Configuration Item	Configuration Description
VLAN100	Ports 2-6 of switches A and C.
Trunk port	Port 11 of switches A and C; ports 10 and 11 of switch B.
Global GVRP	Switches A, B, and C.
Port GVRP	Port 11 of switches A and C; ports 10 and 11 of switch B.

Connect the two workstations to the VLAN100 ports on switches A and C, connect port 11 of switch A to port 10 of switch B, and connect port 11 of switch B to port 11 of switch C.

The configuration steps are as follows:

```
Switch A:
Switch(config)# gvrp
Switch(config)#vlan 100
Switch(Config-Vlan100)#switchport interface ethernet 1/0/2-6
Switch(Config-Vlan100)#exit
Switch(config)#interface ethernet 1/0/11
Switch(Config-If-Ethernet1/0/11)#switchport mode trunk
Switch(Config-If-Ethernet1/0/11)# gvrp
Switch(Config-If-Ethernet1/0/11)#exit
Switch B:
Switch(config)#gvrp
Switch(config)#interface ethernet 1/0/10
Switch(Config-If-Ethernet1/0/10)#switchport mode trunk
Switch(Config-If-Ethernet1/0/10)# gvrp
Switch(Config-If-Ethernet1/0/10)#exit
Switch(config)#interface ethernet 1/0/11
Switch(Config-If-Ethernet1/0/11)#switchport mode trunk
Switch(Config-If-Ethernet1/0/11)# gvrp
```

```
Switch(Config-If-Ethernet1/0/11)#exit
Switch C:
Switch(config)# gvrp
Switch(config)#vlan 100
Switch(Config-Vlan100)#switchport interface ethernet 1/0/2-6
Switch(Config-Vlan100)#exit
Switch(config)#interface ethernet 1/0/11
Switch(Config-If-Ethernet1/0/11)#switchport mode trunk
Switch(Config-If-Ethernet1/0/11)# gvrp
Switch(Config-If-Ethernet1/0/11)#exit
```

3.14.4 GVRP Troubleshooting

The GARP timer settings of the Trunk ports at both ends of a Trunk link must be the same; otherwise, GVRP does not work normally. Avoid enabling GVRP and RSTP on the switch at the same time. If GVRP must be enabled, disable RSTP on the ports first.

3.15 Dot1q-tunnel

3.15.1 Introduction to Dot1q-tunnel

Dot1q-tunnel, also called QinQ (802.1Q-in-802.1Q), is an extension of 802.1Q. Its main idea is to encapsulate the customer VLAN tag (CVLAN tag) in the service provider VLAN tag (SPVLAN tag). Carrying both VLAN tags, the packet is transmitted through the ISP's backbone network, providing a simple layer-2 tunnel for users. Dot1q-tunnel is simple and easy to manage, requires only static configuration, and is especially suitable for small office networks or small metropolitan area networks that use layer-3 switches as backbone equipment.

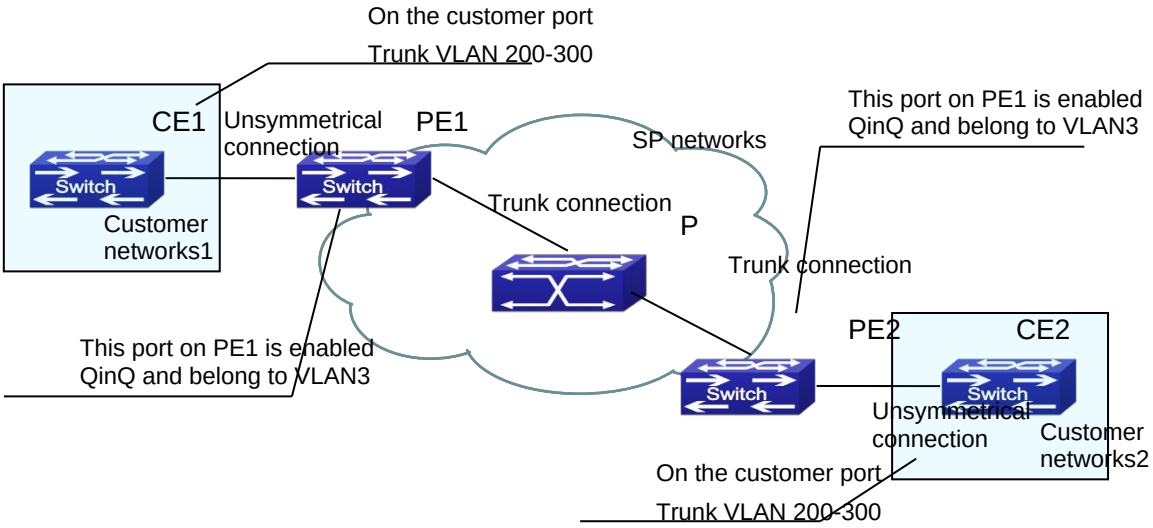


Figure 3-16 Dot1q-tunnel-Based Internetworking Mode

As shown above, after dot1q-tunnel is enabled on the user port, it assigns each user an SPVLAN identifier (SPVID). Here the user's identifier is 3. The same SPVID must be assigned to the same network user on different PEs. When a packet from CE1 reaches PE1, it carries a VLAN tag in the range 200-300 of the user's internal network. Because dot1q-tunnel is enabled, the user port on PE1 adds another VLAN tag to the packet, whose ID is the SPVID assigned to the user. The packet is then transmitted only in VLAN3 while traveling through the ISP network, carrying two VLAN tags (the inner tag is the one it carries when entering PE1, and the outer tag is the SPVID), and the VLAN information of the user network is transparent to the provider network. When the packet reaches PE2, the outer VLAN tag is removed before the packet is forwarded from the client port on PE2 to

CE2, so the packet CE2 receives is identical to the one sent by CE1. For the user, the operator network between PE1 and PE2 simply provides a reliable layer-2 link.

Dot1q-tunnel technology allows the ISP network to support many client VLANs with only one of its own VLANs. Both the ISP network and the clients can configure their own VLANs independently.

The dot1q-tunnel function has the following characteristics:

- ☝ It requires only simple static configuration, with no complex configuration or maintenance.

☞ Operators only need to assign one SPVID to each user, which increases the number of users that can be supported concurrently, while users have complete freedom to select and manage their VLAN IDs (any value within 1-4096).

- ☝ User networks are highly independent. When the ISP upgrades its network, user networks do not need to change their original configuration.

This section describes the application and configuration of dot1q-tunnel in detail.

3.15.2 Dot1q-tunnel Configuration

Dot1q-tunnel configuration task sequence:

1. Configure the dot1q-tunnel function on the port

2. Configure the protocol type (TPID) on the port

1. Configure the dot1q-tunnel function on the port

Command	Explanation
Port Mode	
dot1q-tunnel enable **no dot1q-tunnel enable**	Enables dot1q-tunnel on the port. The no command disables it.

2. Configure the protocol type (TPID) on the port

Command	Explanation
Port Mode	
dot1q-tunnel tpid {0x8100|0x9100|0x9200|<1-65535>}	Configures the protocol type (TPID) on a Trunk port.

3.15.3 Typical Dot1q-tunnel Application

Scenario:

Edge switches PE1 and PE2 of the ISP network use VLAN3 to forward VLAN200-300 data between CE1 and CE2 of the client network. Port 1 of PE1 is connected to CE1 and port 10 to the public network; the TPID of the connected equipment is 9100. Port 1 of PE2 is connected to CE2 and port 10 to the public network.

Configuration Item	**Configuration Explanation**
VLAN3	Port 1 of PE1 and PE2.
dot1q-tunnel	Port 1 of PE1 and PE2.
tpid	9100

The configuration procedure is as follows:

PE1:

```
Switch(config)#vlan 3
Switch(Config-Vlan3)#switchport interface ethernet 1/0/1
Switch(Config-Vlan3)#exit
Switch(Config)#interface ethernet 1/0/1
Switch(Config-Ethernet1/0/1)# dot1q-tunnel enable
Switch(Config-Ethernet1/0/1)# exit
```

```
Switch(Config)#interface ethernet 1/0/10
Switch(Config-Ethernet1/0/10)#switchport mode trunk
Switch(Config-Ethernet1/0/10)#dot1q-tunnel tpid 0x9100
Switch(Config-Ethernet1/0/10)#exit
Switch(Config)#
```

PE2:

```
Switch(config)#vlan 3
Switch(Config-Vlan3)#switchport interface ethernet 1/0/1
Switch(Config-Vlan3)#exit
Switch(Config)#interface ethernet 1/0/1
Switch(Config-Ethernet1/0/1)# dot1q-tunnel enable
Switch(Config-Ethernet1/0/1)# exit
Switch(Config)#interface ethernet 1/0/10
Switch(Config-Ethernet1/0/10)#switchport mode trunk
Switch(Config-Ethernet1/0/10)#dot1q-tunnel tpid 0x9100
Switch(Config-Ethernet1/0/10)#exit
Switch(Config)#
```

3.15.4 Dot1q-tunnel Troubleshooting

- ☞ Enabling dot1q-tunnel on a Trunk port makes the tags of data packets unpredictable, which is not desired in this application. Therefore, it is not recommended to enable dot1q-tunnel on Trunk ports.
- ☞ Dot1q-tunnel cannot be enabled together with STP/MSTP.
- ☞ Dot1q-tunnel cannot be enabled together with PVLAN.

3.16 VLAN-translation

3.16.1 Introduction to VLAN-translation

As its name suggests, VLAN translation translates the original VLAN ID to a new VLAN ID according to user requirements, so that data can be exchanged across different VLANs. VLAN translation is divided into ingress translation and egress translation; this switch supports only ingress VLAN ID translation.

This section describes the application and configuration of VLAN translation in detail.

Access ports of the switch do not support this function.

3.16.2 VLAN-translation Configuration

VLAN-translation configuration task sequence:

1. Configure the VLAN-translation function on the port
2. Configure the VLAN-translation relations on the port
3. Configure whether packets are dropped when the VLAN-translation check fails
4. Display the VLAN-translation configuration

1. Configure the VLAN-translation function on the port

Command	Explanation
Port Mode	
vlan-translation enable **no vlan-translation enable**	Enables VLAN-translation on the port. The no command disables it.

2. Configure the VLAN-translation relations on the port

Command	Explanation

Port Mode	
vlan-translation <*old-vlan-id*> to <*new-vlan-id*> in **no vlan-translation old-vlan-id in**	Adds or deletes a VLAN-translation relation.

3. Configure whether packets are dropped when the VLAN-translation check fails

Command	Explanation
Port Mode	
vlan-translation miss drop in **no vlan-translation miss drop in**	Configures the port to drop packets when VLAN-translation fails. The no command disables dropping.

4. Display the VLAN-translation configuration

Command	Explanation
Admin Mode	
show vlan-translation	Displays the VLAN-translation configuration.

3.16.3 Typical VLAN-translation Application

Scenario:

Edge switches PE1 and PE2 of the ISP network use VLAN3 to carry VLAN20 data between CE1 and CE2 of the client network. Port 1/0/1 of PE1 is connected to CE1 and port 1/0/10 to the public network; port 1/0/1 of PE2 is connected to CE2 and port 1/0/10 to the public network.

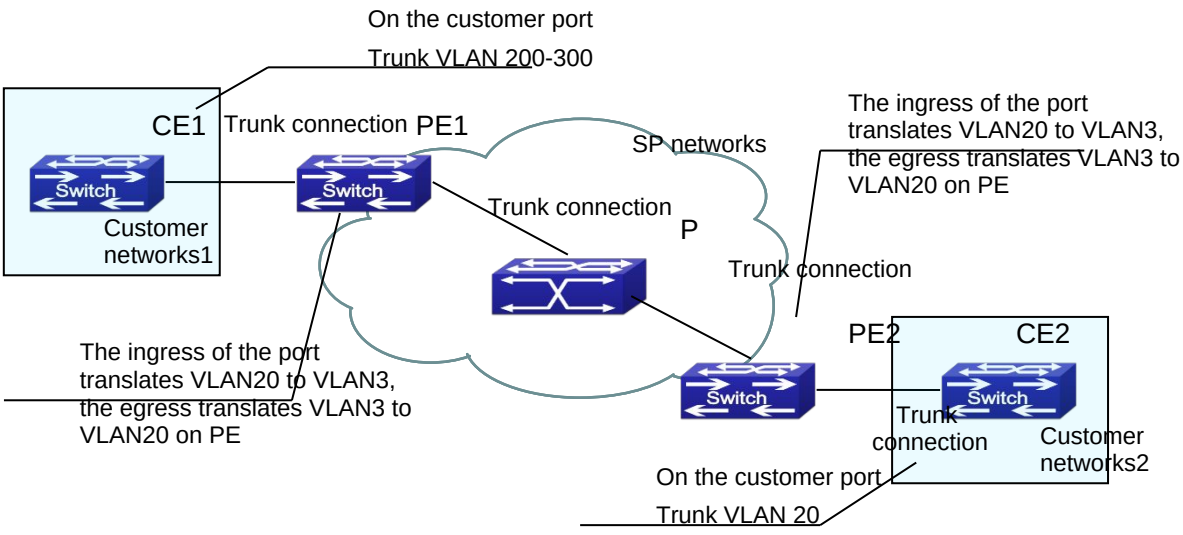


Figure 3-17 VLAN Translation Topology

Configuration Item	Configuration Explanation
VLAN-translation	Port 1/0/1 of PE1 and PE2.
Trunk port	Ports 1/0/1 and 1/0/10 of PE1 and PE2.

The configuration procedure is as follows:

PE1, PE2:

```
switch(Config)#interface ethernet 1/0/1
switch(Config-Ethernet1/0/1)#switchport mode trunk
switch(Config-Ethernet1/0/1)# vlan-translation enable
switch(Config-Ethernet1/0/1)# vlan-translation 20 to 3 in
switch(Config-Ethernet1/0/1)# vlan-translation 3 to 20 out
switch(Config-Ethernet1/0/1)# exit
```

```
switch(Config)#interface ethernet 1/0/10
switch(Config-Ethernet1/0/10)#switchport mode trunk
switch(Config-Ethernet1/0/10)#exit
switch(Config)#
```

3.16.4 VLAN-translation Troubleshooting

- ☞ VLAN-translation is normally applied on Trunk ports.

The processing priority of VLAN translation and VLAN ingress filtering is: VLAN translation > VLAN ingress filtering.

3.17 Dynamic VLAN

3.17.1 Introduction to Dynamic VLAN

Dynamic VLANs are so named in contrast to static VLANs (port-based VLANs). The switch supports the following dynamic VLAN types: MAC-based VLAN, IP-subnet-based VLAN, and protocol-based VLAN. They are described as follows:

A MAC-based VLAN assigns hosts to VLANs based on the MAC address of each host: every host, identified by its MAC address, is assigned to a specific VLAN. As a result, users keep their VLAN membership when they move from one physical location to another. The greatest advantage of this method is that the VLAN does not need to be reconfigured when a user's physical location changes (for example, when the user moves from one switch to another), because membership is user-based rather than switch-port-based.

An IP-subnet-based VLAN assigns hosts to VLANs based on the source IP address and subnet mask of each host. The switch assigns a VLAN ID to each packet according to its subnet and forwards the packet in that VLAN. Its advantage is the same as that of the MAC-based VLAN: users do not need to change their configuration when they relocate.

A protocol-based VLAN assigns traffic to VLANs based on the network-layer protocol, so different protocols can be assigned to different VLANs. This is attractive to network administrators who want to organize users by application and service. Users can also move freely within the network while keeping their VLAN membership, so they can change physical location without changing their VLAN configuration, while administrators can still divide VLANs by protocol type. In addition, this method does not require an added frame tag to identify the VLAN, which reduces network traffic.

Note: Dynamic VLANs work only together with the Hybrid port attribute, so any port that may be added to a dynamic VLAN must be configured as a Hybrid port.

3.17.2 Dynamic VLAN Configuration

Dynamic VLAN configuration task sequence:

1. Enable the MAC-based VLAN function on the port
2. Set the VLAN as a MAC VLAN
3. Configure the mapping between the MAC address and the VLAN
4. Enable the IP-subnet-based VLAN function on the port
5. Configure the mapping between the IP subnet and the VLAN
6. Configure the mapping between the protocol and the VLAN
7. Adjust the priority of the dynamic VLAN

1. Enable the MAC-based VLAN function on the port

Command	Explanation

Port Mode	
switchport mac-vlan enable **no switchport mac-vlan enable**	Enables the MAC-based VLAN function on the port. The no command disables it.

2. Set the VLAN as a MAC VLAN

Command	Explanation
Global Mode	
mac-vlan vlan <*vlan-id*> **no mac-vlan**	Configures the specified VLAN as a MAC VLAN. The **no mac-vlan** command cancels the MAC VLAN configuration of this VLAN.

3. Configure the mapping between the MAC address and the VLAN

Command	Explanation
Global Mode	
mac-vlan mac <mac-addrss> <mac-mask> vlan <vlan-id> priority <priority-id> **no mac-vlan {mac <mac-addrss> <mac-mask>|all}**	Adds a mapping between a MAC address and a VLAN, so that the specified MAC address joins the specified VLAN. The no command deletes the mapping, so that the MAC address leaves the VLAN.

4. Enable the IP-subnet-based VLAN function on the port

Command	Explanation
Port Mode	
switchport subnet-vlan enable **no switchport subnet-vlan enable**	Enables the IP-subnet-based VLAN function on the port. The no command disables it.

5. Configure the mapping between the IP subnet and the VLAN

Command	Explanation
Global Mode	
subnet-vlan ip-address <*ipv4-addrss*> mask <*subnet-mask*> vlan <*vlan-id*> priority <*priority-id*> **no subnet-vlan {ip-address <*ipv4-addrss*> mask <*subnet-mask*>|all}**	Adds a mapping between an IP subnet and a VLAN, so that the specified IP subnet joins the specified VLAN. The no command deletes the mapping, so that the subnet leaves the VLAN.

6. Configure the mapping between the protocol and the VLAN

Command	Explanation
Global Mode	
protocol-vlan mode {ethernetii etype <*etype-id*>|llc {dsap <*dsap-id*> ssap <*ssap-id*>}|snap etype <*etype-id*>} vlan <*vlan-id*> priority <*priority-id*> **no protocol-vlan {mode {ethernetii etype <*etype-id*>|llc {dsap <*dsap-id*> ssap <*ssap-id*>}|snap etype <*etype-id*>}|all}**	Adds a mapping between a protocol and a VLAN, so that the specified protocol joins the specified VLAN. The no command deletes the mapping, so that the protocol leaves the VLAN.

7. Adjust the priority of the dynamic VLAN

Command	Explanation
Global Mode	
dynamic-vlan mac-vlan prefer **dynamic-vlan subnet-vlan prefer**	Configures the priority of the dynamic VLAN types.

3.17.3 Typical Application of the Dynamic VLAN

Scenario:

In an office network, Department A belongs to VLAN 100. Several members of this department often move around the office network, and they must still be able to access the resources of the other department members in VLAN 100. Assume one member is M, whose PC has the MAC address 00-03-0f-11-22-33. When M moves to VLAN 200 or VLAN 300, the port connecting M is configured in Hybrid mode and joins VLAN 100 as untagged. VLAN 100 data is then forwarded to the port connecting M, so M can communicate in VLAN 100.

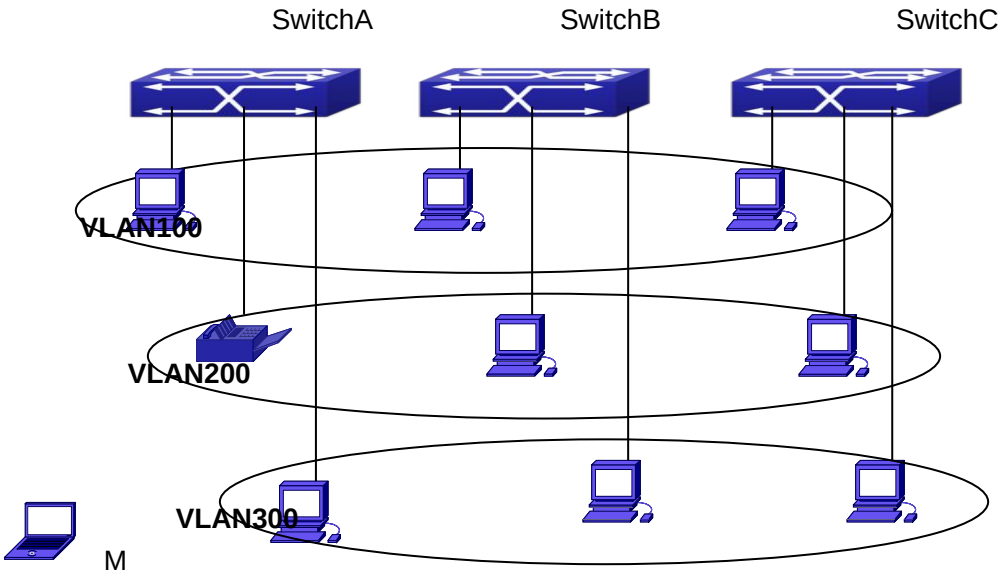


Figure 3-18 Typical dynamic VLAN application topology

Configuration Items	Configuration Explanation
MAC-based VLAN	Global configuration on Switch A, Switch B, and Switch C.

For example, if M is connected to E1/0/1 of Switch A, the configuration procedure is as follows:

```
Switch A, Switch B, Switch C:
SwitchA (Config)#mac-vlan mac 00-03 -0f-11-22-33 vlan 100 priority 0
SwitchA (Config)#interface ethernet 1/0/1
SwitchA (Config-Ethernet1/0/1)# swportport mode hybrid
SwitchA (Config-Ethernet1/0/1)# swportport hybrid allowed vlan 100 untagged
SwitchB (Config)#mac-vlan mac 00-03-0f-11-22-33 vlan 100 priority 0
SwitchB (Config)#exit
SwitchB#
SwitchC (Config)#mac-vlan mac 00-03-0f-11-22-33 vlan 100 priority 0
SwitchC (Config)#exit
SwitchC#
```

3.17.4 Dynamic VLAN Troubleshooting

☞ On a switch configured with a dynamic VLAN, if two connected devices (for example, PCs) both belong to the same dynamic VLAN, the first communication between them may fail. To solve this, have both devices actively send packets to the switch (for example, with ping) so that the switch learns their source MAC addresses. The two devices can then communicate freely within the dynamic VLAN.

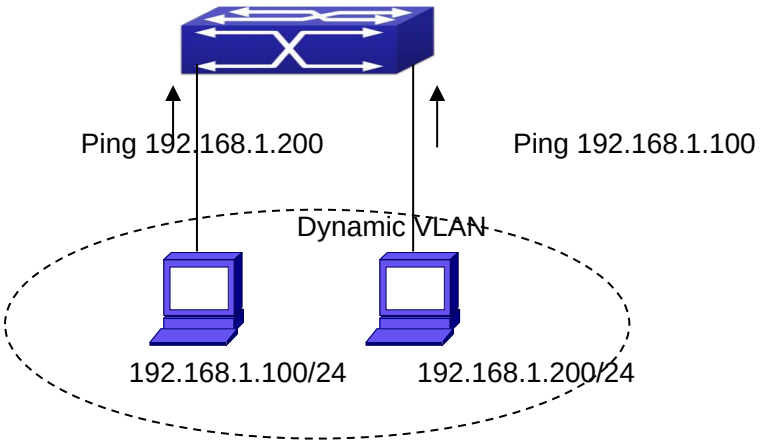


Figure 3-19 Dynamic VLAN troubleshooting

☞ When the switch processes packets, dynamic VLAN takes priority over VLAN ingress filtering (dynamic VLAN > VLAN ingress filtering).

3.18 Voice VLAN

3.18.1 Introduction to Voice VLAN

A Voice VLAN is configured specifically for user voice traffic. By creating a Voice VLAN and adding the ports connected to voice devices to it, you can configure QoS (Quality of Service) for voice data and raise the transmission priority of voice traffic to ensure call quality.

The switch determines whether traffic is voice traffic from a specified device based on the source MAC address of the packets entering the port. Packets whose source MAC address matches a voice device OUI (Organizationally Unique Identifier) defined in the system are treated as voice traffic and forwarded in the Voice VLAN.

The configuration is MAC-address-based, relying on the fact that every voice device transmitting over the network has a unique MAC address. The VLAN tracks the address of the specified MAC, so a voice device always belongs to the Voice VLAN even when it is physically relocated. The greatest advantage of this VLAN is that devices are placed into the Voice VLAN automatically according to their voice traffic, which is then transmitted at the specified priority. When a voice device is physically relocated, it still belongs to the Voice VLAN without any configuration changes, because membership is based on the voice device rather than the switch port.

Note: The Voice VLAN works only together with the Hybrid port attribute, so any port that may be added to the Voice VLAN must be configured as a Hybrid port.

3.18.2 Voice VLAN Configuration

Voice VLAN configuration task sequence:

1. Set the VLAN as the Voice VLAN
2. Add a voice device to the Voice VLAN
3. Enable the Voice VLAN on the port

1. Set the VLAN as the Voice VLAN

Command	Explanation
Global Mode	

voice-vlan vlan <*vlan-id*> **no voice-vlan**	Sets the VLAN as the Voice VLAN. The no command cancels the Voice VLAN setting.

2. Add a voice device to the Voice VLAN

Command	Explanation
Global Mode	
voice-vlan mac <*mac-address*> mask <*mac-mask*> priority <*priority-id*> [name <*voice-name*>] **no voice-vlan {mac <*mac-address*> mask <*mac-mask*>|name <*voice-name*> |all}**	Specifies a voice device to join the Voice VLAN. The no command removes the device from the Voice VLAN.

3. Enable the Voice VLAN on the port

Command	Explanation
Port Mode	
switchport voice-vlan enable **no switchport voice-vlan enable**	Enables the Voice VLAN function on the port. The no command disables it.

3.18.3 Typical Applications of the Voice VLAN

Scenario:

A company implements voice communication by configuring a Voice VLAN. IP-phone1 and IP-phone2 can be connected to any port of the switch for normal communication and connect to other switches through the uplink port. IP-phone1 has MAC address 00-03-0f-11-22-33 and connects to port 1/0/1 of the switch; IP-phone2 has MAC address 00-03-0f-11-22-55 and connects to port 1/0/2 of the switch.

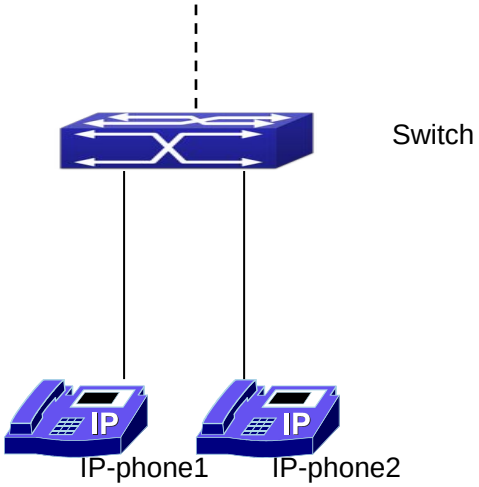


Figure 3-27 Typical Voice VLAN application topology

Configuration Items	Configuration Explanation
Voice VLAN	Global configuration on the Switch.

Configuration procedure:

```
Switch 1:
Switch(config)#vlan 100
Switch(Config-Vlan100)#exit
Switch(config)#voice-vlan vlan 100
Switch(config)#voice-vlan mac 00-03-0f-11-22-33 mask 255 priority 5 name
company
```

```
Switch(config)#voice-vlan mac 00-03-0f-11-22-55 mask 255 priority 5 name
company
Switch(config)#interface ethernet 1/0/10
Switch(Config-If-Ethernet1/0/10)#switchport mode trunk
Switch(Config-If-Ethernet1/0/10)#exit
switch(Config)#interface ethernet 1/0/1
switch(Config-If-Ethernet1/0/1)#switchport mode hybrid
switch(Config-If-Ethernet1/0/1)#switchport hybrid allowed vlan 100 untag
switch(Config-If-Ethernet1/0/1)#exit
switch(Config)#interface ethernet 1/0/2
switch(Config-If-Ethernet1/0/2)#switchport mode hybrid
switch(Config-If-Ethernet1/0/2)#switchport hybrid allowed vlan 100 untag
switch(Config-If-Ethernet1/0/2)#exit
```

3.18.4 Voice VLAN Troubleshooting

☞ The Voice VLAN cannot be used together with the MAC-based VLAN.

3.19 Multi-to-One VLAN Translation

3.19.1 Introduction to Multi-to-One VLAN Translation

Multi-to-One VLAN translation translates the original VLAN IDs of uplink traffic into a new VLAN ID as required, and restores the original VLAN IDs for downlink traffic.

This section describes the application and configuration of Multi-to-One VLAN translation in detail.

Access ports of the switch do not support this function.

3.19.2 Multi-to-One VLAN Translation Configuration

Multi-to-One VLAN translation configuration task list:

1. Configure Multi-to-One VLAN translation on the port
2. Display the Multi-to-One VLAN translation configuration

1. Configure Multi-to-One VLAN translation on the port

Command	Explanation
Port Mode	
vlan-translation n-to-1 <*WORD*> to <*new-vlan-id*> **no vlan-translation n-to-1 <*WORD*>**	Configures Multi-to-One VLAN translation. The no command deletes it.

2. Display the Multi-to-One VLAN translation configuration

Command	Explanation
Admin Mode	
show vlan-translation n-to-1	Displays the Multi-to-One VLAN translation configuration.

3.19.3 Typical Application of Multi-to-One VLAN Translation

Scenario:

UserA, userB, and userC belong to VLAN 1, VLAN 2, and VLAN 3, respectively. Before entering the network layer, the traffic of userA, userB, and userC is translated into VLAN 100 by Ethernet1/0/1 of edge switch1. In the reverse direction, traffic from the network layer to userA, userB, and userC is translated back into VLAN 1, VLAN 2, and VLAN 3, respectively, by Ethernet1/0/1 of edge switch1. In

the same way, Multi-to-One translation is implemented for userD, userE, and userF on Ethernet1/0/1 of edge switch2.

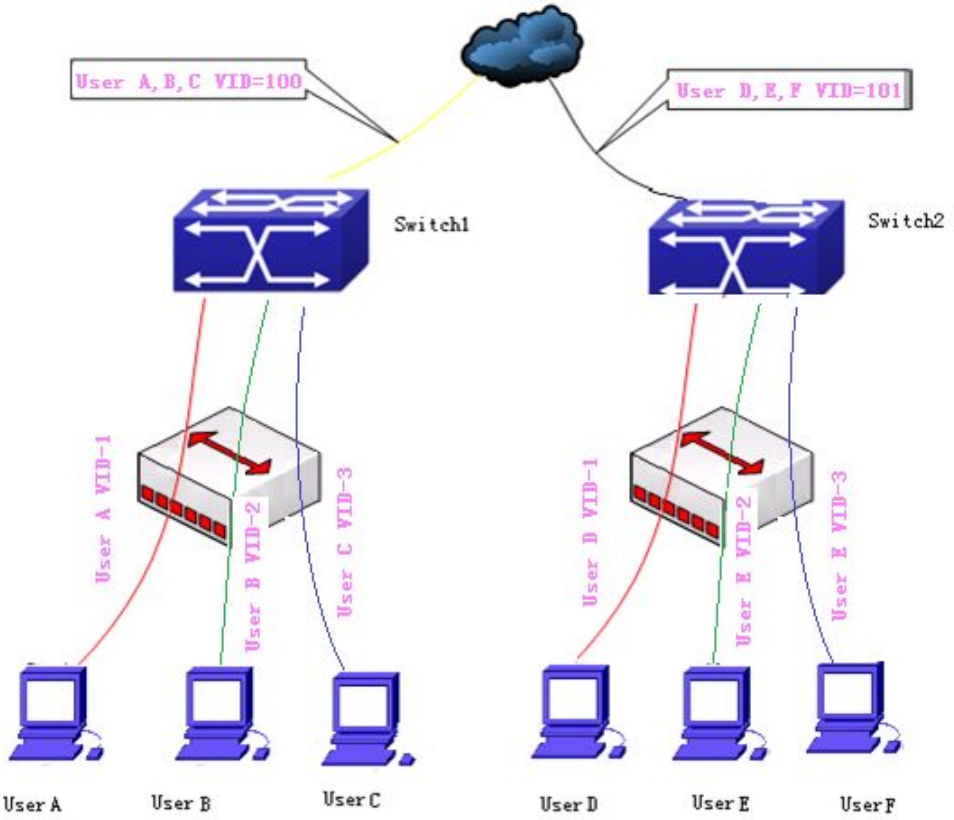


Figure 3-28 Typical VLAN translation application

Configuration Item	Configuration Explanation
VLAN	Switch1, Switch2
Trunk Port	Downlink port 1/0/1 and uplink port 1/0/5 of Switch1 and Switch2
Multi-to-One VLAN-translation	Downlink port 1/0/1 of Switch1 and Switch2

The configuration procedure is as follows:

```
Switch1、Switch2:
switch(Config)# vlan 1-3;100
switch(Config-Ethernet1/0/1)#switchport mode trunk
switch(Config-Ethernet1/0/1)# vlan-translation n-to-1 1-3 to 100
switch(Config)#interface ethernet 1/0/5
switch(Config-Ethernet1/0/5)#switchport mode trunk
switch(Config-Ethernet1/0/5)#exit
```

3.19.4 Multi-to-One VLAN Translation Troubleshooting

- Do not use this function together with Dot1q-tunnel.
- Do not use this function together with VLAN translation.
- The same MAC address must not exist in both the original VLAN and the translated VLAN.
- Check whether the chip's hardware resources are sufficient for all clients to work normally.
- Limiting MAC address learning may affect Multi-to-One VLAN translation.
- Multi-to-One VLAN translation must be enabled after MAC learning.

3.20 Super VLAN

3.20.1 Introduction to Super VLAN

Super VLAN, also called VLAN aggregation, introduces the concepts of super VLAN and sub VLAN. One or more sub VLANs belong to a super VLAN, and a sub VLAN no longer occupies a separate subnet. Within the same super VLAN, regardless of which sub VLAN a host belongs to, its IP address is within the subnet associated with the super VLAN.

VLANs are widely used in LAN switching networks because they provide flexible control of broadcast domains and are easy to deploy. However, a typical layer-3 switch implements communication between broadcast domains by mapping each VLAN to its own layer-3 interface, which wastes IP addresses.

For example, a device's VLAN division is shown in the figure below.

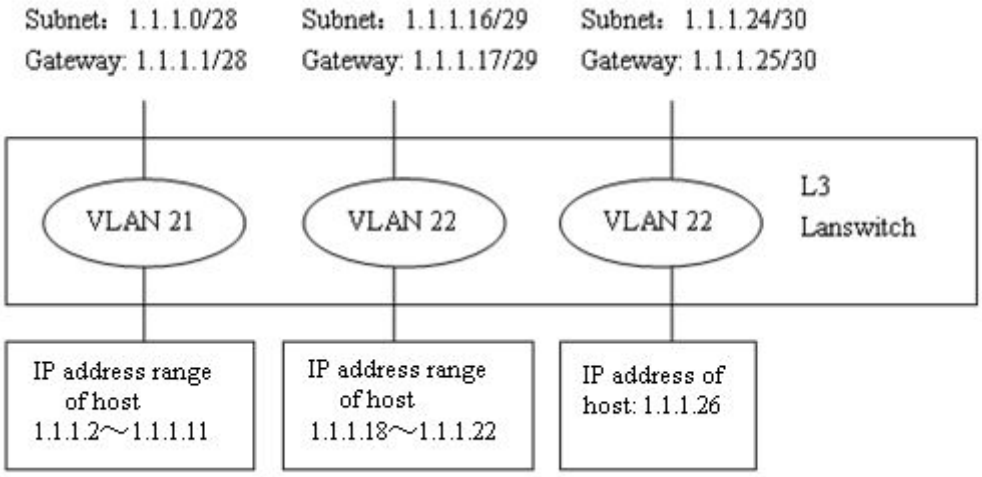


Figure 2-29 Typical VLAN network

VLAN	IP Subnet	Gateway Address	Usable Hosts	Customer Hosts	Needed Hosts
21	1.1.1.0/28	1.1.1.1	14	13	10
22	1.1.1.16/29	1.1.1.17	6	5	5
23	1.1.1.24/30	1.1.1.25	2	1	1

Tab 1-1 Host address division in a typical VLAN network

In the table above, VLAN 21 needs 10 host addresses and is assigned a subnet with a 28-bit mask, 1.1.1.0/28. However, the subnet address 1.1.1.0, the subnet broadcast address 1.1.1.15, and the default gateway address 1.1.1.1 cannot be used as host addresses; only 1.1.1.2 to 1.1.1.14 can. The total number of usable host addresses is therefore 13 (2^(32-28) - 3 = 13), while VLAN 21 needs only 10.

Similarly, VLAN 22 needs 5 host addresses, so it must be assigned a subnet with a 29-bit mask (1.1.1.16/29). VLAN 23 needs only 1 host address, but it occupies the subnet 1.1.1.24/30.

The three VLANs need 16 addresses in total (10 + 5 + 1), but even the optimal division in a typical VLAN network occupies 28 addresses (2^(32-28) + 2^(32-29) + 2^(32-30) = 28), so many addresses are wasted. In addition, if VLAN 21 later needs only 3 hosts instead of 10 but has already been assigned a subnet with a 28-bit mask, the surplus addresses are wasted because no other VLAN can use them.

This division also makes network expansion difficult. If the VLAN 23 customer needs 2 more hosts and does not want to change the assigned IP addresses, but the addresses after 1.1.1.24 have already been assigned to others, a new subnet with a 29-bit mask and a new VLAN must be assigned to this customer. The VLAN 23 customer then has only 3 hosts but two subnets (in different VLANs), which makes management difficult.

As shown above, the number of IP addresses wasted (subnet addresses, broadcast addresses, default gateway addresses) is considerable, and addressing flexibility is greatly reduced. Super VLAN was developed to solve this problem.

The advantages of Super VLAN are as follows:

Reduces the number of IP addresses used for subnet addresses, default gateway addresses, and broadcast addresses

Allows different broadcast domains to use addresses in the same subnet

Improves addressing flexibility

Reduces address waste

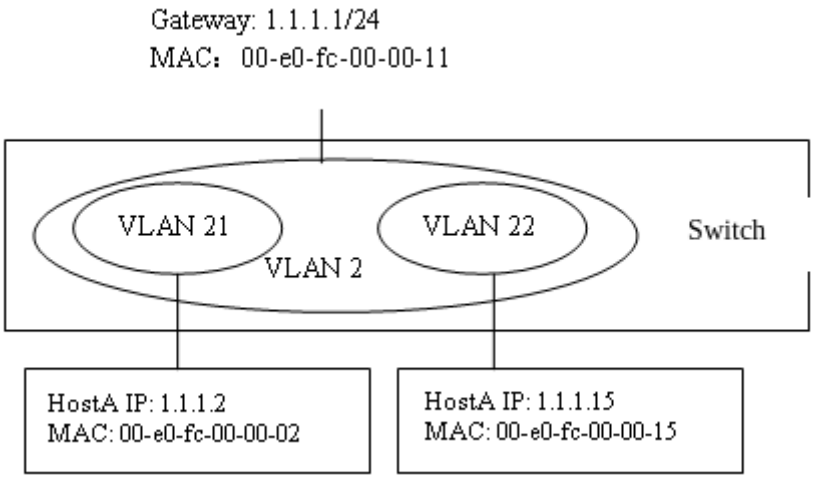


Figure 2-30 Super VLAN function

A super VLAN differs from a typical VLAN. A super VLAN only creates a layer-3 interface and does not contain any ports; it is a layer-3 concept.

The layer-3 interface of a super VLAN is up as long as at least one physical port in its sub VLANs is up.

3.20.2 Super VLAN Configuration

1. Create or delete a super VLAN
2. Specify or delete a sub VLAN
3. Enable or disable the ARP proxy function of a sub VLAN
4. Specify or delete the IP address range of an interface
5. Specify or delete the IP address range of a sub VLAN

1. Create or delete a super VLAN

Command	Explanation
VLAN Configuration Mode	
supervlan **no supervlan**	Creates a super VLAN. The no command deletes it.

2. Specify or delete a sub VLAN

Command	Explanation
VLAN Configuration Mode	
subvlan WORD **no subvlan {WORD | all}**	Specifies sub VLANs. The no command deletes the specified sub VLANs or all sub VLANs.

3. Enable or disable the ARP proxy function of a sub VLAN

Command	Explanation
VLAN Interface Mode	
arp-proxy subvlan {WORD | all} **no arp-proxy subvlan {WORD | all}**	Enables the ARP proxy function of the sub VLAN. The no command disables it.

4. Specify or delete the IP address range of an interface

Command	Explanation
VLAN Interface Mode	
ip-addr-range <*ipv4-addrss*> to <*ipv4-addrss*> **no ip-addr-range**	Specifies the IP address range of the interface. The no command deletes it.

5. Specify or delete the IP address range of a sub VLAN

Command	Explanation
VLAN Interface Mode	
ip-addr-range subvlan <*vlan-id*> <*ipv4-addrss*> to <*ipv4-addrss*> **no ip-addr-range subvlan <*vlan-id*>**	Specifies the IP address range of a sub VLAN. The no command deletes it.

3.20.3 Typical Application of Super VLAN

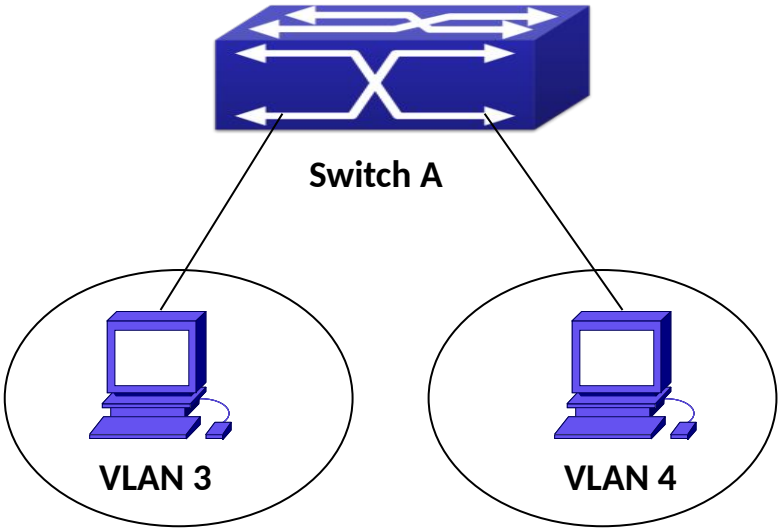


Figure 3-31 Typical super VLAN topology

Because of LAN application requirements, the terminals in two VLANs must be configured with addresses in the same subnet. Layer-2 traffic must be isolated, but layer-3 traffic between the terminals must be forwarded normally. The address range of VLAN 3 is 1.1.1.1 to 1.1.1.10, and the address range of VLAN 4 is 1.1.1.20 to 1.1.1.30; only layer-3 traffic from terminals within these two address ranges is forwarded. To meet this requirement, configure a super VLAN on the switch.

Configuration Items	**Configuration Explanation**
VLAN2	Super VLAN

VLAN3	Port 1 of Switch A
VLAN4	Port 2 of Switch A

Configuration procedure:

```
Switch A:
switch(Config)#vlan 2-4
switch(Config)#vlan 2
switch(Config-Vlan2)#supervlan
switch(Config-Vlan2)#subvlan 3;4
switch(Config-Vlan2)#exit
switch(Config)#interface vlan 2
switch(config-if-vlan2)#ip address 1.1.1.254 255.255.255.0
switch(config-if-vlan2)#arp-proxy subvlan all
switch(config-if-vlan2)#ip-addr-range subvlan 3 1.1.1.1 to 1.1.1.10
switch(config-if-vlan2)#ip-addr-range subvlan 4 1.1.1.20 to 1.1.1.30
switch(config-if-vlan2)#exit
```

3.20.4 Super VLAN Troubleshooting

- Super VLAN is mutually exclusive with other functions (such as VRRP, dynamic VLAN, private VLAN, and multicast VLAN), so do not use them at the same time.
- The ARP proxy of a sub VLAN takes effect on that sub VLAN only. Traffic received by a VLAN with ARP proxy enabled can be forwarded to other VLANs. When two devices in different sub VLANs send traffic to each other, enable the ARP proxy function on both sub VLANs.
- A layer-3 interface cannot be configured on a sub VLAN.
- When a super VLAN is created or deleted, the VLAN must not have a layer-3 interface; otherwise, an error occurs.
- If an IP address range is specified for the super VLAN interface but not for the sub VLANs, the address range of the interface applies. If IP address ranges are specified for both the interface and a sub VLAN, the switch first checks whether packets are within the sub VLAN address range, and then whether they are within the interface address range, before processing the packets further.
- A VLAN can be set as a super VLAN or sub VLAN only if it already exists.
- When a port is set to trunk mode, the super VLAN is automatically filtered out of its allowed VLANs.

3.21 MAC Address Table

3.21.1 Introduction to the MAC Table

The MAC table maps destination MAC addresses to switch ports. MAC addresses are classified as static or dynamic. Static MAC addresses are configured manually by the user, have the highest priority, and are permanently effective (they are not overwritten by dynamic MAC addresses). Dynamic MAC addresses are entries the switch learns while forwarding data frames, and they are effective only for a limited period. When the switch receives a data frame to be forwarded, it stores the source MAC address of the frame and maps it to the receiving port. The switch then looks up the destination MAC address in the MAC table; if there is a match, the frame is forwarded through the associated port; otherwise, the switch forwards the frame to its broadcast domain. If a dynamic MAC address is not learned again from forwarded frames for a long time, its entry is deleted from the MAC table.

There are two MAC table operations:

1. Obtain MAC addresses.
2. Forward or filter data frames according to the MAC table.

3.21.1.1 Obtaining the MAC Table

The MAC table can be built statically or dynamically. Static configuration sets up mappings between MAC addresses and ports manually; dynamic learning is the process in which the switch learns mappings between MAC addresses and ports and updates the MAC table regularly. This section focuses on the dynamic learning process.

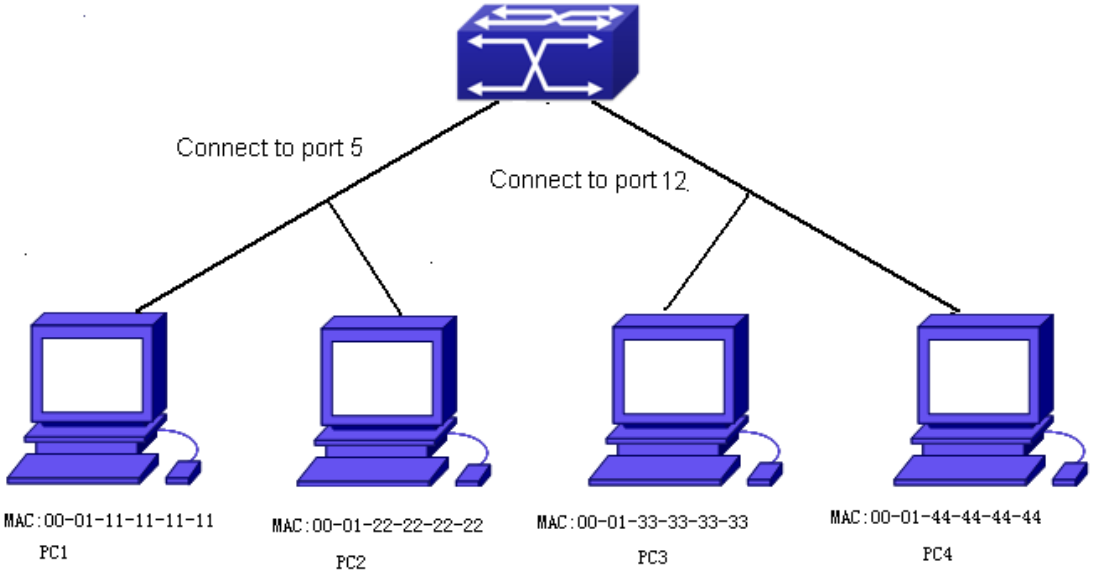


Figure 3-32 Dynamic MAC table learning

In the topology above, four PCs are connected to the switch. PC1 and PC2 belong to the same physical segment (the same collision domain), which connects to port 1/0/5 of the switch; PC3 and PC4 belong to the same physical segment, which connects to port 1/0/12 of the switch.

Initially, the MAC table contains no address mapping entries. Taking communication between PC1 and PC3 as an example, the MAC address learning process is as follows:

1. When PC1 sends a message to PC3, the switch reads the source MAC address 00-01-11-11-11-11 from the message and adds a mapping entry for 00-01-11-11-11-11 and port 1/0/5 to the MAC table.
2. At the same time, the switch learns that the message is destined for 00-01-33-33-33-33. Because the MAC table contains only the entry for MAC address 00-01-11-11-11-11 and port 1/0/5, and no port mapping exists for 00-01-33-33-33-33, the switch broadcasts the message to all ports (assuming all ports belong to the default VLAN 1).
3. PC3 and PC4 on port 1/0/12 both receive the message sent by PC1. Because the destination MAC address is 00-01-33-33-33-33, PC4 does not reply; only PC3 replies to PC1. When port 1/0/12 receives the reply from PC3, a mapping entry for MAC address 00-01-33-33-33-33 and port 1/0/12 is added to the MAC table.
4. The MAC table now has two dynamic entries: MAC address 00-01-11-11-11-11 to port 1/0/5, and 00-01-33-33-33-33 to port 1/0/12.
5. After the communication between PC1 and PC3 ends, if the switch receives no further messages from PC1 or PC3, the corresponding entries are deleted from the MAC table within 300 to 2 × 300 seconds (that is, within one to two aging times). 300 seconds is the default aging time for MAC address entries on the switch; the aging time can be modified.

3.21.1.2 Forward or Filter

The switch forwards or filters received data frames according to the MAC table. Using the figure above as an example, assume the switch has learned the MAC addresses of PC1 and PC3, and the

user has manually configured the port mappings for PC2 and PC4. The MAC table of the switch is then as follows:

MAC Address	Port Number	Entry Added By
00-01-11-11-11-11	1/0/5	Dynamic learning
00-01-22-22-22-22	1/0/5	Static configuration
00-01-33-33-33-33	1/0/12	Dynamic learning
00-01-44-44-44-44	1/0/12	Static configuration

1．Forward data according to the MAC table

If PC1 sends a message to PC3, the switch forwards the data received on port 1/0/5 through port 1/0/12.

2．Filter data according to the MAC table

If PC1 sends a message to PC2, the switch checks the MAC table, finds that PC2 and PC1 are in the same physical segment, and filters (drops) the message.

The switch can forward three types of frames:

- ☞ Broadcast frame
- ☞ Multicast frame
- ☞ Unicast frame

The switch handles the three types of frames as follows:

1．Broadcast frame: The switch can separate collision domains but not broadcast domains. If no VLAN is configured, all devices connected to the switch are in the same broadcast domain, and when the switch receives a broadcast frame, it forwards the frame through all ports. When VLANs are configured on the switch, the MAC table includes VLAN information. In this case, the switch does not forward received broadcast frames through all ports, but only through the ports in the same VLAN.

2．Multicast frame: The switch broadcasts unknown multicast frames within the same VLAN. If IGMP Snooping or a static multicast group is configured, the switch forwards multicast frames only to the ports of the multicast group.

3．Unicast frame: When no VLAN is configured, if the destination MAC address is in the MAC table, the switch forwards the frame directly to the associated port; if the destination MAC address is not found in the MAC table, the switch broadcasts the unicast frame. When VLANs are configured, the switch forwards unicast frames only within the same VLAN. If the destination MAC address is found in the MAC table but belongs to a different VLAN, the switch can only broadcast the unicast frame within its own VLAN.

3.21.2 MAC Address Table Configuration Task List

1．Configure the MAC address aging time

2．Configure static MAC forwarding or filter entries

3．Clear the dynamic address table

4．Configure MAC learning through CPU control

5．Configure the bucket size for the MAC address table

1. **Configure the MAC aging-time**

Command	Explanation
Global Mode	
mac-address-table aging-time <*0|aging-time*> **no mac-address-table aging-time**	Configures the MAC address aging time.

2. **Configure static MAC forwarding or filter entry**

Command	Explanation
Global Mode	
mac-address-table {static | static-multicast | blackhole} address <*mac-addr*> vlan <*vlan-id* > [interface [ethernet | portchannel] <*interface-name*>] | [source| destination|both] **no mac-address-table {static | static-multicast | blackhole | dynamic} [address <*mac-addr*>] [vlan <*vlan-id*>] [interface [ethernet | portchannel] <*interface-name*>]**	Configures static MAC entries, static multicast MAC entries, and filter (blackhole) address entries.

3. **Clear dynamic address table**

Command	Explanation
Admin Mode	
clear mac-address-table dynamic [address <*mac-addr*>] [vlan <*vlan-id*>] [interface [ethernet | portchannel] <*interface-name*>]	Clears the dynamic address table.

4. **Configure MAC learning through CPU control**

Command	Explanation
Global Mode	
mac-address-learning cpu-control **no mac-address-learning cpu-control**	Enables MAC learning through CPU control. The no command restores automatic MAC address learning by the chip.
show collision-mac-address-table	Displays the hash collision MAC table.
Admin Mode	
clear collision-mac-address-table	Clears the hash collision MAC table.

5. **Configure bucket size for MAC address table**

Command	Explanation
Global Mode	
mac-address-table bucket size <1-4>	Sets the bucket size for the MAC address table to improve learning performance for random MAC addresses.

3.21.3 Typical Configuration Examples

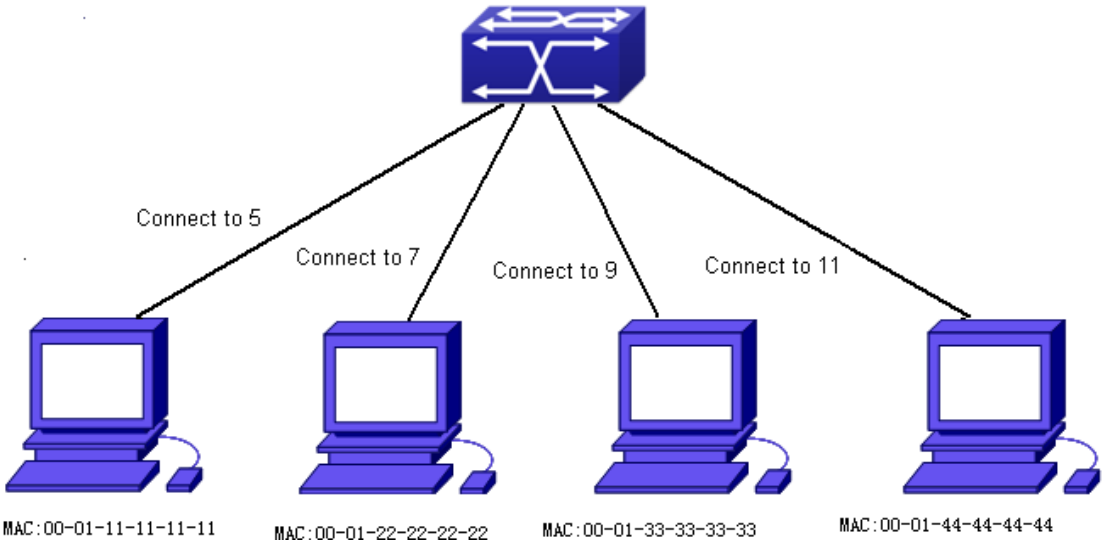


Figure 3-33 Typical MAC table configuration example

Scenario:

As shown in the figure above, four PCs connect to ports 1/0/5, 1/0/7, 1/0/9, and 1/0/11 of the switch, and all four PCs belong to the default VLAN 1. As required by the network environment, dynamic learning is enabled. PC1 holds sensitive data and must not be accessed by any PC in another physical segment. PC2 and PC3 have static mappings to port 1/0/7 and port 1/0/9, respectively.

The configuration steps are as follows:

1. Set the MAC address 00-01-11-11-11-11 of PC1 as a filter address.

```
Switch(config)#mac-address-table static 00-01-11-11-11-11 discard vlan 1.
```

2. Set the static mappings for PC2 and PC3 to port 1/0/7 and port 1/0/9, respectively.

```
Switch(config)#mac-address-table static address 00-01-22-22-22-22 vlan 1
interface ethernet 1/0/7
Switch(config)#mac-address-table static address 00-01-33-33-33-33 vlan 1
interface ethernet 1/0/9
```

3.21.4 MAC Table Troubleshooting

The show mac-address-table command shows that a port has failed to learn the MAC address of a connected device. Possible reasons:

- The connected cable is broken.
- Spanning Tree is enabled and the port is in "discarding" state, or the device has just been connected and Spanning Tree is still calculating. Wait until the Spanning Tree calculation finishes, and the port will learn the MAC address.
- If the cause is not one of the problems above, check the switch port. If the problem persists, contact EnGenius technical support.

3.22 CFM-OAM Configuration

3.22.1 Overview

Since Ethernet was introduced, its simplicity and low cost have made it the dominant technology in local area networks. In recent years, Gigabit and 10-Gigabit Ethernet have been introduced one after

the other, prompting network providers, equipment manufacturers, and standards bodies to extend Ethernet technology to metropolitan and wide area networks.

However, Ethernet is fundamentally a LAN technology. Remote monitoring and SLA (Service Level Agreement) testing are not pressing needs in a LAN environment, so traditional Ethernet lacks the OAM functions that network providers require. In addition, the link failure recovery time of traditional Ethernet (10 seconds, or 1 second at best) is not acceptable to network providers. Achieving a failure recovery time of less than 50 milliseconds is therefore also a major challenge for metro Ethernet.

Since several Ethernet OAM standards (for example, IEEE 802.3ah, IEEE 802.1ag, and ITU Y.1731) have been published, OAM is no longer a weakness of Ethernet. IEEE 802.1ag, also known as the Connectivity Fault Management (CFM) standard, provides end-to-end network monitoring and operation tools. It can perform tasks such as MAC Ping and L2 Traceroute in large Layer 2 Ethernet networks, which simplifies troubleshooting and SLA monitoring in Ethernet operation. In addition, the CCM (Continuity Check Message) defined in the Ethernet OAM standards can support protection switching in less than 50 milliseconds. The shortest CCM transmission interval in the standard is 3.3 milliseconds; losing three consecutive CCMs causes the primary link to be declared failed and replaced by the backup link. The effectiveness of CFM depends on proper network planning and configuration.

3.22.1.1 Ethernet OAM Protocol Standards

There are four Ethernet OAM protocol standards, defined by different organizations: 802.3ah (Ethernet in the First Mile, EFM), 802.1ag (Connectivity Fault Management, CFM), E-LMI (Ethernet Local Management Interface), and Y.1731 (fault and performance monitoring). They are as follows:

IEEE 802.3ah: Ethernet Link OAM (EFM OAM)

IEEE 802.1ag: Connectivity Fault Management (CFM)

ITU-T Y.1731: OAM functions and mechanisms for Ethernet-based networks

MEF E-LMI: Ethernet Local Management Interface

EFM OAM and CFM are defined by the IEEE. EFM OAM works at the data link layer and can effectively discover and manage the underlying data links. EFM OAM improves Ethernet management and maintenance capabilities and helps keep network operation stable. CFM is a network-level Ethernet OAM technology, mainly used in the access aggregation layer to monitor network connectivity and locate connectivity faults. Y.1731, defined by the ITU (International Telecommunication Union), is much more powerful than CFM; the functions of CFM can be regarded as a subset of Y.1731. E-LMI, defined by the MEF, applies only to the UNI (the user-network interface between the user and the provider edge facing the user).

3.22.2 CFM OAM Basic Concepts

802.1ag divides the whole network (customer, provider, operator) into different MDs (Maintenance Domains). Within each maintenance domain, the services provided by the MD are maintained and managed, and these services involve many nodes. The core idea of 802.1ag is to check some or all of the nodes involved so that faulty parts can be found. A point that takes part in maintenance checking is called an MP (Maintenance Point); MPs are divided into MEPs (maintenance end points) and MIPs (maintenance intermediate points), and are configured on bridge ports.

3.22.2.1 Maintenance Domain

A network can be logically divided into different layers from inside to outside, called MDs (Maintenance Domains). Maintenance domains can be nested but cannot overlap. Each maintainer can see only its own maintenance domain. A lower-level domain can provide services to the adjacent upper level. This clearly separates maintainers, such as operators and users, and makes network problems easier to locate.

A network can contain multiple maintenance domains, and each maintenance domain is at a specific level; there are 8 levels in total. The higher the level, the larger the maintenance scope. Domains of different levels can be nested. In practice, CFM is typically applied as shown below, with the network divided into a customer domain, a service provider domain, an operator domain, and so on.

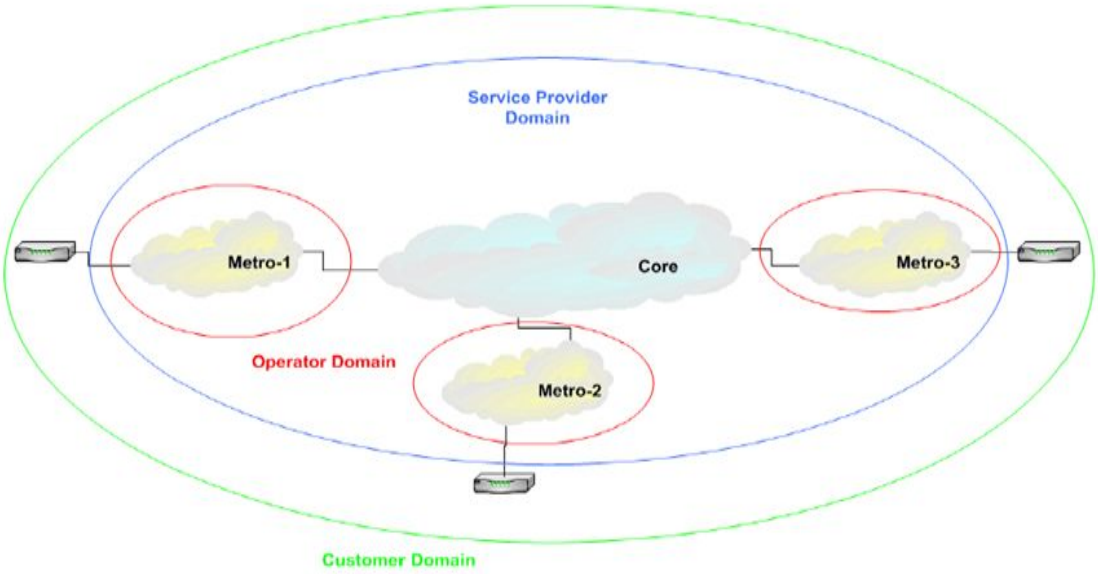


Fig 2-34 Maintenance domain

In addition to its level, each MD has a globally unique MD name that identifies it.

3.22.2.2 Maintenance Association

Each service instance in a maintenance domain is an MA (Maintenance Association). An MD usually provides several service instances, and 802.1ag manages, maintains, and detects faults on each MA individually. Each MA has a unique name within its MD. In a network, a service instance is usually identified by a VLAN ID, and an MA corresponds to a service instance, so MAs are linked to VLAN IDs. One MA corresponds to one VLAN ID, although the protocol also allows multiple VLAN IDs to be managed and maintained by one MA; one of these VLANs is the primary VLAN. The case in which one VLAN ID corresponds to multiple MAs in an MD is undefined and is not considered here. In short, the MD and the primary VLAN ID uniquely identify an MA.

3.22.2.3 Maintenance End Point (MEP)

A MEP belongs to a specific maintenance service and marks the boundary of that service; it is configured on a port. MEPs initiate all CFM messages (CCM, LTM, LBM), and most protocol behavior and states occur on MEPs. MEPs are divided into Up MEPs and Down MEPs. On a bridge, a MEP that sends and receives the CFM messages of its MA through the LAN is a Down MEP; a MEP that sends and receives the CFM messages of its MA through the Bridge Relay is an Up MEP, as shown in the following figure:

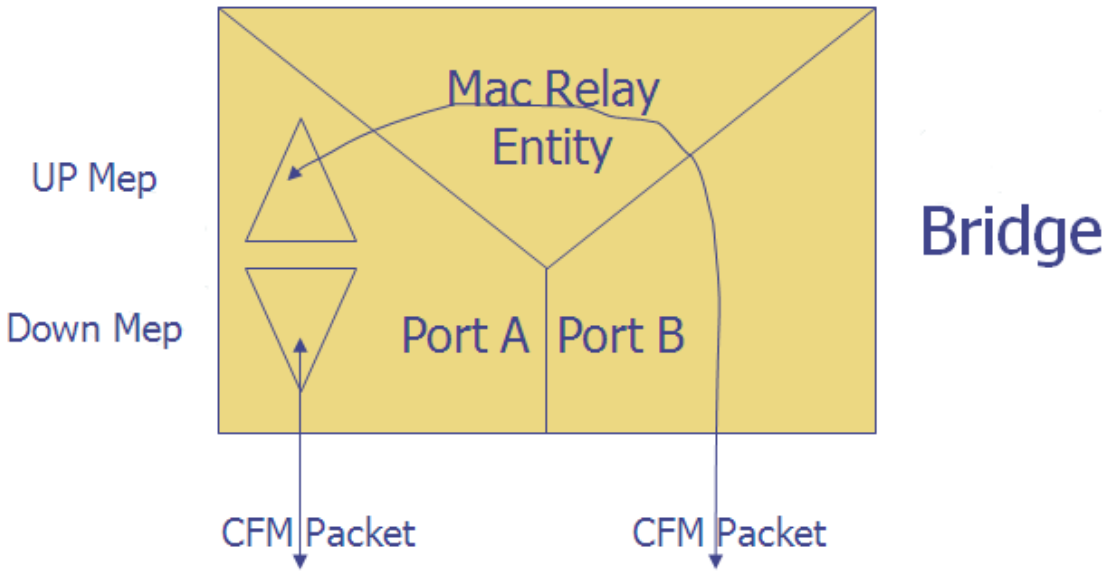


Fig 2-35 Difference between Down MEP and Up MEP

Put simply, a MEP that sends and receives CFM messages through its local port is a Down MEP; otherwise, it is an Up MEP. A MEP inherits the attributes of the MD and MA in which it is located, namely the MD level and the VLAN ID. Each MEP has a unique identifier within its MA, called the MEPID.

3.22.2.4 Maintenance Intermediate Point (MIP)

A MIP belongs to a specific maintenance service, is an intermediate point of that service, and is configured on a port. A MIP cannot send CFM messages actively; it can only respond to received messages by sending replies and forwarding them. It is not responsible for detecting and reporting faults, but it assists MEPs in fault detection. A MIP inherits the attributes of the MD and MA in which it is located, namely the MD level and the VLAN ID.

MIPs are not configured directly; the system creates them according to certain rules. On a port, each maintenance level can have only one MIP. The MIP creation rules are as follows:

none: No MIP is created.

default: If there is no higher-level MEP on the port and no lower-level MIP exists, a MIP is created on the port at this level.

explicit: If there is no higher-level MEP on the port, a lower-level MEP exists, and no lower-level MIP exists, a MIP is created on the port at this level.

defer: Whether a MIP is created is determined by the MIP creation rule configured for the MD of the MA.

3.22.3 Introduction to CFM OAM Functions

802.1ag provides five functions: fault detection, fault verification and isolation, path discovery, fault notification, and fault recovery. Fault recovery must be performed together with other protocols.

3.22.3.1 Fault Detection

A MEP periodically sends CCMs to the remote MEPs in the same MA and receives CCMs from the remote MEPs. If it does not receive a CCM within 3 CCM intervals, it considers the link faulty and reports the fault. The process is shown in the following figure:

CCMs can be sent to the other MEPs in the same MA as multicast messages. The last 3 bits of the multicast address represent the maintenance domain level, which makes hardware processing

easier. When a lower-level MEP receives a higher-level message, the hardware can forward it directly based on the multicast MAC address and VLAN.

Fault detection through periodic CCMs works as follows:

- If a MEP does not receive a CCM on time, a fault has occurred in the MA.
- By checking received CCMs, a MEP can detect a mismatch in the transmission interval.
- If a MEP receives a CCM with an incorrect MEPID or MAID, there is a configuration error within the MA or a cross-connection error.
- If a MEP receives a lower-level CCM, there is a configuration error within the MA or a cross-connection error.
- If a MEP receives a CCM carrying MAC status information, it can detect faults outside the MA.

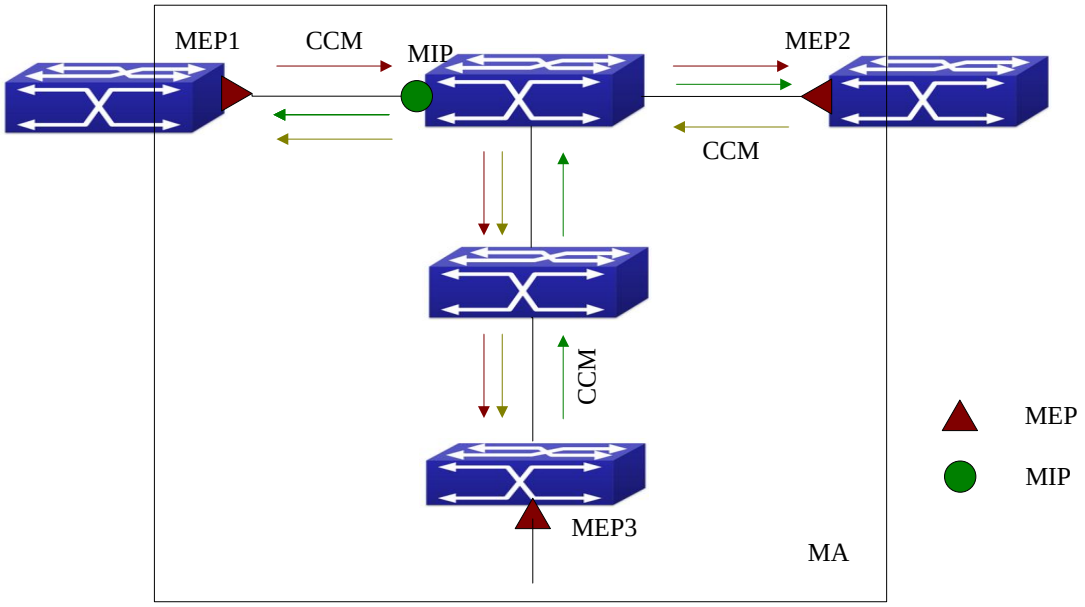


Fig 2-36 Continuity check diagram

3.22.3.2 Path Discovery

Path discovery is performed by MEPs and MIPs through Linktrace. The 802.1ag Linktrace function is similar to IP Traceroute: it sends test messages and receives reply messages to check the path to a destination device or locate a fault point. The process is as follows: the MEP sends an LTM to the target MP (MEP or MIP). Each MIP that receives the LTM sends an LTR back to the source MEP and then forwards the LTM, until the LTM reaches the destination MP or can no longer be forwarded. Based on the returned LTRs, the source MEP determines the link status and obtains the path to the target MAC address, as shown in the following figure.

The destination MAC address of an LTM is a multicast address. The last 3 bits of the multicast address represent the maintenance domain level. An LTR is a unicast message whose destination address is the source MAC address of the LTM.

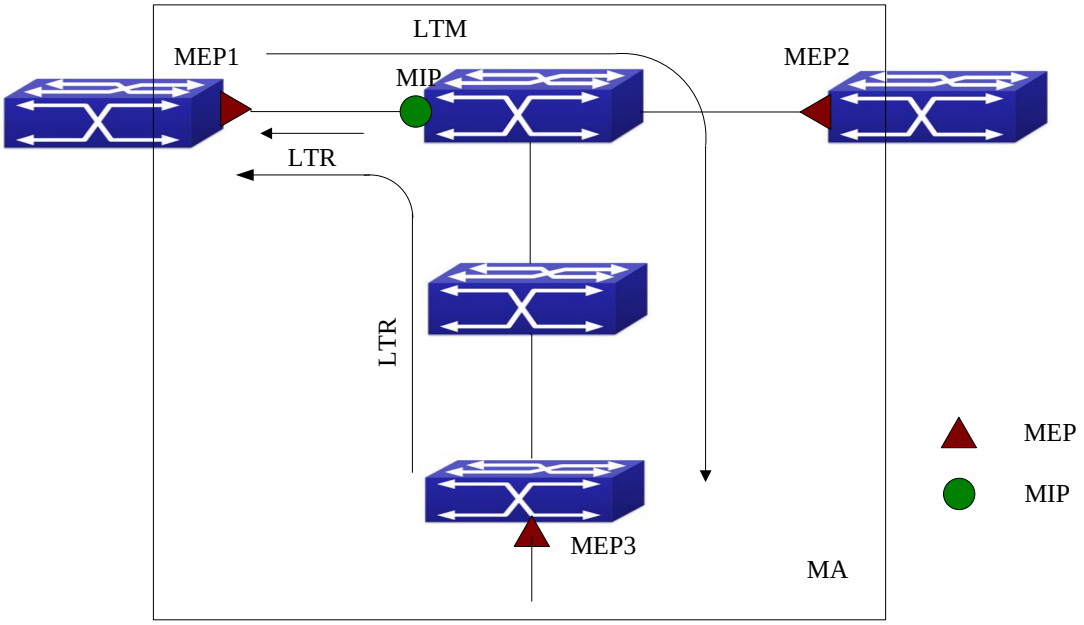


Fig 2-37 Path discovery diagram

3.22.3.3 Fault Verification and Location

Fault verification and location can be performed with the 802.1ag Linktrace function described above, or with the 802.1ag Loopback function. The Loopback function is similar to IP Ping: it sends test messages and receives reply messages to check whether the destination device is reachable. The process is as follows: the MEP sends a unicast loopback message (LBM) whose destination address is the remote MP. Intermediate devices forward the LBM, and the remote MP sends a loopback reply (LBR) to the source MEP after receiving the LBM. Based on this, the source MEP determines whether the remote MP is reachable, as shown in the following figure.

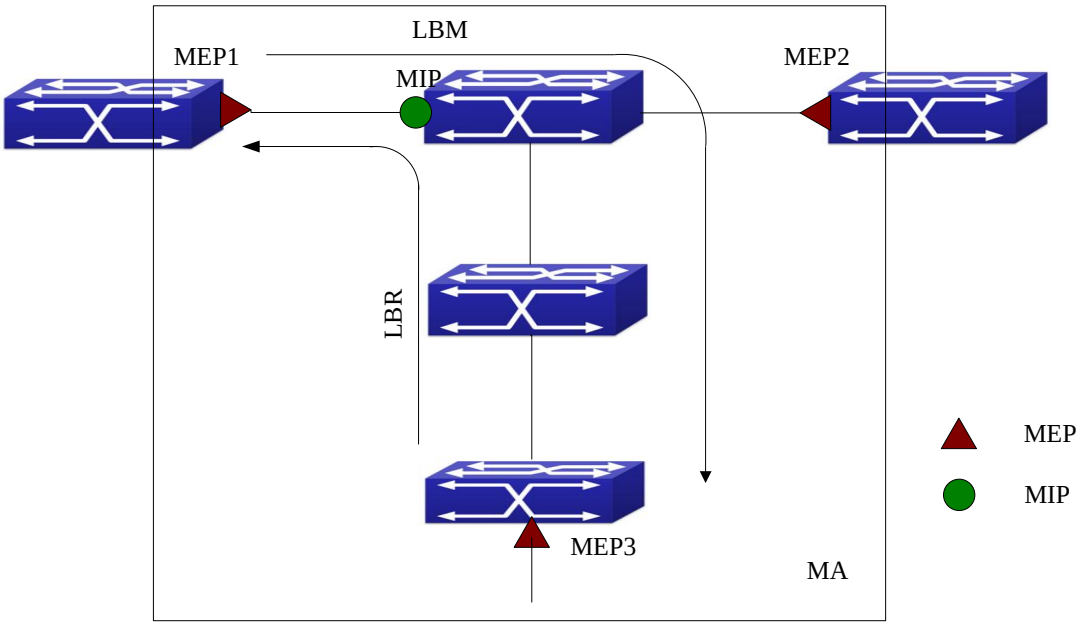


Fig 2-38 Loopback function

3.22.3.4 Fault Notification

After CFM detects a link fault, it can handle the fault in several ways:

The MEP sends an SNMP trap to the management station to report the fault.

The MEP writes a record to the device log, and administrators can find the problem by checking the log.

CFM works with other automatic protection protocols, such as APS, to recover from the fault. CFM notifies these protocols of the fault, and they negotiate an automatic link switchover.

3.22.4 CFM OAM Basic Function Configuration

3.22.4.1 CFM Management Topology Design

Before using the CFM OAM function, plan the network as follows:

1. Divide the maintenance domains of the whole network into levels and determine the boundaries of each domain at each level.
2. Determine the name of each maintenance domain; all devices in the same maintenance domain use the same name.
3. Determine the maintenance services in each maintenance domain according to the VLANs to be monitored.
4. Determine the name of each maintenance service; all devices in the same maintenance service of a domain use the same name.
5. Ensure that all devices in the same maintenance service of the same domain use the same MEP list.
6. Configure MEPs on ports at the boundaries of maintenance domains and services, and configure MIPs on non-boundary devices or ports.

The following data is therefore required:

Serial Number	Data
1	MD name and level
2	MA name, MA related VLAN ID
3	MEP ID, name of the port on which the MEP is configured, and MEP type
4	RMEP ID
5	MIP creation rules
6	Interval at which MEPs in the MA send and detect CCMs

3.22.4.2 CFM OAM Configuration Task List

1. Select the CFM OAM function mode
2. Enable the CFM OAM function globally
3. Enable the Y.1731 function globally (optional)
4. Create an MD
5. Create an MA
6. Create a MEP
7. Configure an RMEP
8. Create a MIP (optional)
9. Enable the fault alarm function (optional)
10. Configure CC sending and detection
11. Check the CFM configuration

1. **Select the CFM OAM function mode**

Command	Explanation
Global Mode	

ethernet cfm mode {hw|sw|auto} **no ethernet cfm mode**	Selects the mode in which CFM OAM is enabled. This command can be used only before the CFM OAM function is enabled. The no command restores the default mode (auto).

2. **Enable the CFM OAM function globally**

Command	Explanation
Global Mode	
ethernet cfm global **no ethernet cfm global**	Enables the CFM OAM function globally. The no command disables it.

3. **Enable the Y.1731 function globally (optional)**

Command	Explanation
Global Mode	
ethernet cfm y1731 global **no ethernet cfm y1731 global**	Enables the Y.1731 function. After this function is enabled, the switch enters Y.1731 mode, and messages are sent and decoded in Y.1731 format. **Notice:** The **ethernet cfm global** command must be used before this command; otherwise, this command does not take effect. The no command disables the function.

4. **Create an MD**

Command	Explanation
Global Mode	
ethernet cfm domain < *domain-name* > level < *level-id* > ***no* ethernet cfm domain < *domain-name* >**	Creates an MD and enters MD Configuration Mode. After the MD is created successfully, its level cannot be modified. The no command deletes the created MD.
MD Configuration Mode	
id {mac-address XX-XX-XX-XX-XX-XX domain-number < *domain-number* > | dns < *dns-name* > | null } **no id**	Configures the MDID. The maintenance domain name configured with the ***ethernet cfm domain*** command is not included in the message. The MDID and the MA name together form the MAID; the total length of the MAID is 44, and this length cannot be exceeded; otherwise, an error occurs. The no command deletes the configured ID.

5. **Create an MA**

Command	Explanation
MD Configuration Mode	
service { < *ma-name* > | number < *ma-num* > | pvlan < *vlan-id* > }{ port | pvlan < *vlan-id* > } [vlan < *WORD* >] [direction down] **no service { < ma-name > | number < ma-num > | pvlan < vlan-id > }**	Creates an MA, configures its Up/Down property, and enters MA Configuration Mode. A service can be associated with one or more VLANs. After the MA is created successfully, its associated VLANs and Up/Down property cannot be modified; to modify them, delete the MA and create it again. A switch supports up to 512 MAs. The no command deletes the created MA.

6. **Create a MEP**

Command	Explanation
MA Configuration Mode	
mep mepid < *WORD* > **no mep mepid [< *WORD* >]**	Creates the list of MEPs permitted in the MA. The no command deletes the created MEPs.
Port Mode	
ethernet cfm mep < *mepid* > domain < *domain-name* > service { < *ma-name* > | number < *ma-num* > | pvlan < *vlan-id* > } **no ethernet cfm mep < mepid > domain < domain-name > service { < ma-name > | number < ma-num > | pvlan < vlan-id > }**	Creates a MEP on the port. The MEP property is determined when the MA is created: if the MA is Up (or Down), all MEPs in this MA are Up (or Down). The no command deletes the created MEP.

7. **Configure an RMEP**

Command	Explanation
MA Configuration Mode	
continuity-check receive rmep <*mep-id*> [active time < *time* >] **no continuity-check receive rmep <*mep-id*>**	Enables CCM reception and creates an RMEP in the MA. A MEPID that has been configured as a MEP in an MA cannot be configured as an RMEP. The no command deletes the configured RMEP.

8. **Create a MIP (optional)**

Command	Explanation
MD Configuration Mode; MA Configuration Mode	
mip auto-create [lower-mep-only | none] **no mip auto-create**	Configures automatic MIP creation in the domain of the MA. By default, no MIP creation rule is configured, and the sender ID is not carried. The no command deletes the MIP.
Global Mode	
ethernet cfm mip auto-create level < *level-id* > vlan < *WORD* > [lower-mep-only] [sender-id chassis] **no mip auto-create**	Configures MIP creation at a level that is not associated with an MA. By default, no MIP creation rule is configured, and the sender ID is not carried. The no command deletes the MIP.

9. **Enable the fault alarm function (optional)**

Command	Explanation
Global Mode	
ethernet cfm alarm {delay < *mseconds* > | notification { all | error-xcon | mac-remote-error-xcon | none | remote-error-xcon | xcon} | reset < *mseconds* >} **no ethernet cfm alarm { delay | notification { all | error-xcon | mac-remote-error-xcon | none | remote-error-xcon | xcon} | reset }**	Configures the fault alarm function. The no command restores the default.
ethernet cfm logging **no ethernet cfm logging**	Enables the logging function. When an alarm occurs, it is recorded in the log. The no command disables this function.

ethernet cfm snmp-server enable traps **no ethernet cfm snmp-server enable traps**	Enables SNMP notifications for alarms. When this is configured, an SNMP notification is sent when an alarm occurs. The no command disables this function.

10. **Configure CC sending and detection**

Command	Explanation
MA Configuration Mode	
continuity-check enable **no continuity-check enable**	Enables CCM sending and receiving on the maintenance point. The no command disables local CCM sending and detection.
continuity-check interval < *interval-value* > **no continuity-check interval**	Configures the interval at which the MEP sends CCMs. In software mode, the minimum interval is 100 ms (interval value 3); in hardware mode, the minimum interval is 3.3 ms (interval value 1). The no command restores the default of 1 s (interval value 4).

11. **Check the CFM configuration**

Command	Explanation
Admin Mode	
show ethernet cfm domain { < domain_name > | brief }	Displays the configuration of maintenance domains.
show ethernet cfm service [domain < *domain-name* > [service { ma-name | number < *ma-num* > | pvlan < *vlan-id* > }]]	Displays the configuration of maintenance associations.
show ethernet cfm maintenance-points local [detail] [mep | mip] [domain < *domain-name* > | interface { ethernet | } <*IFNAME*>]	Displays the attributes and operating information of local maintenance points.
show ethernet cfm maintenance-points remote detail (mac XX-XX-XX-XX-XX-XX | domain WORD (service ((WORD)| (number <*0-65535*>)|(pvlan <*1-4094*>))) mepid <*1-4094*>)	Displays detailed attributes and operating information of remote MEPs.
show ethernet cfm maintenance-points remote (domain WORD (service (WORD|number <*0-65535*>|pvlan <*1-4094*>) (mepid <*1-4094*>|)|)|)	Displays the attributes and operating information of remote MEPs.
show ethernet cfm statistic [domain < *domain-name* > [service { ma-name | number < *ma-num* > | pvlan < *vlan-id* > }]]	Displays CFM message statistics on the device.

3.22.5 CFM OAM Fault Verification

3.22.5.1 Management Topology Confirmation

Before performing fault verification, make sure the CFM OAM function has been configured.

The following data is required for manual fault detection:

Serial Number	Data
1	MD name
2	MA name
3	Destination MEP ID or MAC address

4	Number, size, and timeout of the messages to be sent by the loopback function
5	TTL value of the linktrace messages to be sent

3.22.5.2 Implementing the Loopback Function

Run the following command in Admin Mode:

Command	Explanation
Admin Mode	
ping ethernet [target-mep < *mepid* > | target-mac < *mac-address* >] {domain < *domain-name* > service { < *ma-name* > | number < *ma-num* > | pvlan < *vlan-id* > }} [number < *number* >] [packetsize < *size* >] [timeout < *timeout* >]	Enables the loopback function: sends LBMs from the specified maintenance point to another point and receives LBRs. By default, this function is disabled. If a target MEP ID is entered, the switch looks up the corresponding MAC address; if it cannot find it, an error is displayed. If a MAC address is entered, loopback is performed to that address. If only the domain is specified, the Y.1731 function must be enabled, and multicast LBMs are sent.

3.22.6 CFM OAM Fault Location

3.22.6.1 Management Topology Confirmation

Before performing fault location, make sure the CFM OAM function has been configured.

The following data is required for manual fault location:

Serial Number	Data
1	MD name
2	MA name
3	Destination MEP ID or MAC address
4 (optional)	Maximum hop count (TTL)
5 (optional)	Whether to query only the FDB, or both the FDB and the MIP database, for fault location
6 (optional)	MEP ID that initiates LTM

3.22.6.2 CFM OAM Fault Location Task List

1. **Run the linktrace function**

Command	Explanation
Admin Mode	
traceroute ethernet { target-mep < *target-mep-id* > | target-mac < *mac-address* > } {domain < *domain-name* > service { < *ma-name* > | number < *ma-num* > | pvlan < *vlan-id* > }} [fdb-only | source < *mepid* >]] [ttl < *ttl-value* >]	Checks the path from the specified maintenance point to the target point. By default, the TTL is 64, and both the FDB and the MIP database are queried.

2. **Configure the auto-linktrace function (optional)**

Command	Explanation
MA Configuration Mode	

traceroute ethernet auto **no traceroute ethernet auto**	Enables automatic sending of linktrace packets. By default, this function is disabled. **Notice:** After this function is enabled, if the maintenance point does not receive CCM packets from the remote point within 3.5 CCM intervals, it considers the connection to the remote point faulty and sends an LTM packet (the target of the LTM is the remote maintenance point, and the TTL field is the maximum value of 255) to locate the fault based on the LTR packets received in response. The no command disables this function.
Global Mode	
ethernet cfm auto-traceroute cache { size < *size-value* > | hold-time <*minutes*>} **no ethernet cfm auto-traceroute cache { size | hold-time }**	Configures the size of the auto-linktrace result cache or the hold time of the results. By default, the cache stores the 5 most recent auto-linktrace results, and the hold time is 100 minutes. The no command restores the default.

3. **View the auto-linktrace results**

Command	Explanation
Admin Mode	
show ethernet cfm traceroute-reply auto [domain < *domain_name* > [service { ma-name | number < *ma-num* > | pvlan < *vlan-id* > }]]	Displays the auto-linktrace results. If no domain is specified, all auto-linktrace results on the device are displayed. If no MA is specified, the auto-linktrace results of the specified domain on the device are displayed.

3.22.7 ULPP Linkage (Optional)

3.22.7.1 ULPP Linkage Task List

1. **Configure ULPP linkage**

- ☞ Configure the topology and make sure the CC function is running normally.
- ☞ Configure the ULPP function first.
- ☞ ULPP linkage works only with Down MEPs.
- ☞ The VLAN associated with the linkage MEP must be in the protected VLANs of the relevant ULPP group.

Command	Explanation
Port Mode	
switchport ulpp group <*group-id*> track cfm cc level <*level-value*>	Associates the ULPP group member port with CFM CC detection. When the ULPP group member port receives a matching CFM event (timeout or recovery), it performs the linkage action.

2. **Check the ULPP linkage configuration**

Command	Explanation
Admin Mode	
show ulpp group <*word*>	Displays the ULPP linkage configuration. If the configuration is successful, the MD level is shown in the Track-cfm-level column.

3.22.7.2 Example of ULPP Linkage Application

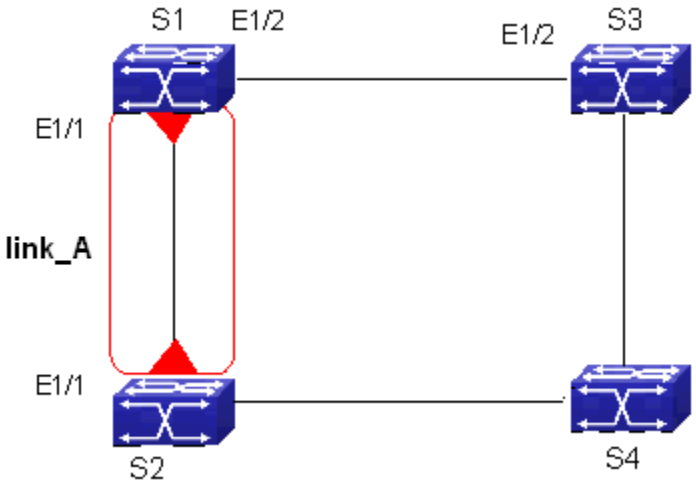


Fig 2-39 ULPP linkage application

1. **Configuration approach:**

Configure the ULPP linkage function as follows:

Create the VLANs and add the related ports to them.

Create the MD link_A, with level 4, on S1 and S2.

Create and configure MA1 in customer_A (MA1 is associated with VLAN 1).

Configure the corresponding MEPs and RMEPs on S1 and S2.

Create ULPP group 1 on S1 and configure the master and slave ports.

Configure the corresponding flush-receiving ports on S2 and S3.

Configure the linkage port on S1.

2. **Configuration Steps**

(1) Create the VLANs and add the related ports to them.

(2) Create the MD link_A and the corresponding MEPs and RMEPs on S1 and S2.

Create link_A on S1 and configure the corresponding MEP and RMEP.

```
Switch(config)#ethernet  cfm domain link_A level 4
Switch(config-ecfm)#service MA1 pvlan 1 direction down
Switch(config-ecfm-srv)#mep mepid 1-2
Switch(config-ecfm-srv)#continuity-check enable
Switch(config-ecfm-srv)#continuity-check receive rmep 2
Switch(config-ecfm-srv)#exit
Switch(config-ecfm)exit
Switch(config)#interface ethernet 1/1
Switch(config-if-ethernet1/1)#ethernet cfm mep 1 domain link_A service MA1
```

Use the same method to create link_A on S2 and configure the corresponding MEP and RMEP.

(3) Create ULPP group 1 on S1 to protect VLAN 1.

Create ULPP group 1 on S1 and enable preemption mode to protect the VLAN.

```
Switch(config)#spanning-tree mst configuration
Switch(config-mstp-region)#instance 1 vlan 1
Switch(config-mstp-region)#exit
Switch(config)#ulpp group 1
Switch(ulpp-group-1)#protect vlan-reference-instance 1
Switch(ulpp-group-1)#flush enable mac-vlan
Switch(ulpp-group-1)#preemption mode
```

Configure port 1/1 on S1 as the master port and port 1/2 as the slave port.

```
Switch(config)#interface ethernet 1/1
Switch(config-if-ethernet1/1)#ulpp group 1 master
Switch(config)#interface ethernet 1/2
Switch(config-if-ethernet1/2)#ulpp group 1 master
```

Configure receiving port 1/1 on S2 to receive flush messages.

```
Switch(config-if-ethernet1/1)#ulpp flush enable mac-vlan
```

Use the same method to configure receiving port 1/2 on S3 to receive flush messages.

(4) Configure ULPP linkage on S1.

Configure ULPP linkage on port 1/1 of S1.

```
Switch(config-if-ethernet1/1)#switchport ulpp group 1 track cfm level 4
```

3. **Checking the configuration**

Check the ULPP linkage configuration on S1.

```
Switch(config)#show ulpp group 1
ULPP group 1 information:
Description:
Preemption mode: ON
Preemption delay: 30s
Control VLAN: 1
Flush packet: MAC MAC-VLAN  ARP
Protected VLAN: Reference Instance 1
Member          Role        State        Track-cfm-level
-------------------------------------------------------------------------------
-----------
Ethernet1/1     MASTER      FORWARD             4
Ethernet1/2     SLAVE        STANDBY              -
-------------------------------------------------------------------------------
-----------
```

If CFM CC detection times out, CFM notifies ULPP to switch over the master and slave ports:

```
%Jan 01 00:12:19 2012 ULPP: ULPP group 1:Master port Ethernet1/3 receives cfm
event type:CFM_ALARM_RMEP_CCM vlan:1 level:4.
%Jan 01 00:12:19 2006 ULPP: ULPP group 1:Master port Ethernet1/3 change state
to down,Slave port Ethernet1/1 change state to forwarding.
%Jan 01 00:12:21 2006 CFM:A CFM_ALARM_RMEP_CCM of Interface Ethernet1/3 is
detected.
```

3.22.8 Configuration Application Example

The following figure illustrates a CFM configuration application. To monitor the link status, configure the switches as described below.

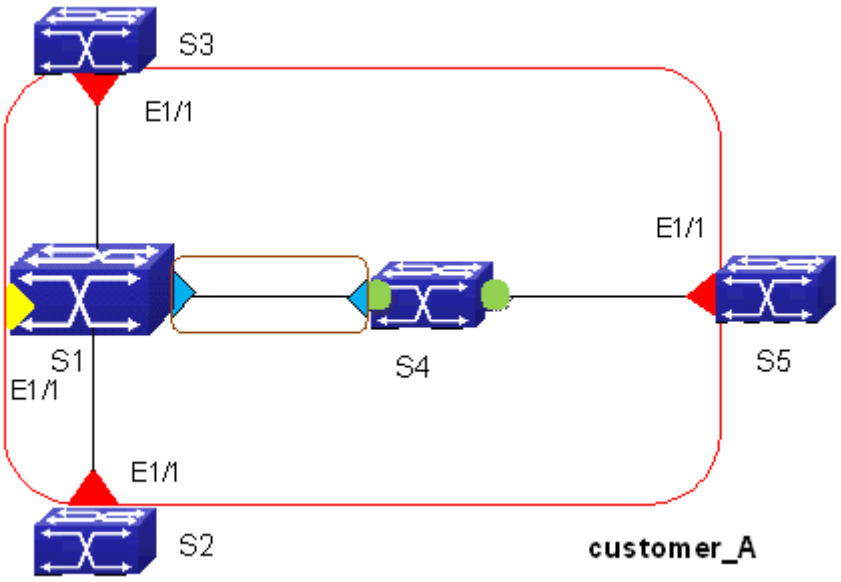


Fig 2-40 MD, MA, MEP, and MIP configuration

1. **Configuration approach**

Configure the basic Ethernet CFM function as follows:

Create the VLANs and add the related ports to them.

Create customer_A on devices S1, S2, S3, S4, and S5; the level of customer_A is 6.

Create and configure MA1 in customer_A (MA1 is associated with VLAN 1).

Create operator_A on devices S1 and S5; the level of operator_A is 3.

Create and configure MA2 in operator_A (MA2 is associated with VLAN 2).

Create local and remote MEPs in MA1 of customer_A on S1, S2, S3, and S4.

Create a MIP in customer_A on S4.

Configure local and remote MEPs in MA2 of customer_A on S1 and S5.

Enable CCM sending and receiving.

2. **Configuration Steps**

(1) Create the VLANs and add the related ports to them.

(2) Enable CFM globally, create customer_A, and configure MA1.

Create customer_A on S1 and configure MA1 in the Up direction.

```
Switch(config)#ethernet cfm global
Switch(config)#ethernet cfm domain customer_A level 6
Switch(config-ecfm)#service MA1 pvlan 1
```

(3) Create customer_A and configure MA1 on S2, S3, S4, and S5.

Create customer_A on S2 and configure MA2 in the Down direction.

```
Switch(config)#ethernet cfm global
Switch(config)#ethernet cfm domain customer_A level 6
Switch(config-ecfm)#service MA1 pvlan 1 direction down
```

Use the same method to create the MD and MA on the other devices.

(4) Create local and remote MEPs in MA1 of customer_A on S1, S2, S3, and S4, and create the MIP on S5.

On S1, create MEP list 1-4 in MA1, configure RMEPs 2-4, and create MEP 1 on Ethernet1/1.

```
Switch(config-ecfm-srv)#mep mepid 1-4
Switch(config-ecfm-srv)#continuity-check receive rmep 2-4
```

```
Switch(config-ecfm-srv)exit
Switch(config-ecfm)exit
Switch(config)#interface ethernet 1/1
Switch(config-if-ethernet1/1)#ethernet cfm mep 1 domain customer_A service
MA1
```

On S2, create MEP list 1-4 in MA1, configure RMEPs 1 and 3-4, and create MEP 2 on Ethernet1/1.

```
Switch(config-ecfm-srv)#mep mepid 1-4
Switch(config-ecfm-srv)#continuity-check receive rmep 1；3-4
Switch(config-ecfm-srv)exit
Switch(config-ecfm)exit
Switch(config)#interface ethernet 1/1
Switch(config-if-ethernet1/1)#ethernet cfm mep 2 domain customer_A service
MA1
```

On S3, create MEP list 1-4 in MA1, configure RMEPs 1, 2, and 4, and create MEP 3 on Ethernet1/1.

```
Switch(config-ecfm-srv)#mep mepid 1-4
Switch(config-ecfm-srv)#continuity-check receive rmep 1；2；4
Switch(config-ecfm-srv)exit
Switch(config-ecfm)exit
Switch(config)#interface ethernet 1/1
Switch(config-if-ethernet1/1)#ethernet cfm mep 3 domain customer_A service
MA1
```

On S4, create MEP list 1-4 in MA1, configure RMEPs 1-3, and create MEP 3 on Ethernet1/1.

```
Switch(config-ecfm-srv)#mep mepid 1-4
Switch(config-ecfm-srv)#continuity-check receive rmep 1-3
Switch(config-ecfm-srv)exit
Switch(config-ecfm)exit
Switch(config)#interface ethernet 1/1
Switch(config-if-ethernet1/1)#ethernet cfm mep 4 domain customer_A service
MA1
```

Use the default rule to create a MIP in MA1 on S5.

```
Switch(config-ecfm-srv)#mip auto-create
```

(5) Create operator_A and configure MA2 on S1 and S5.

Create operator_A on S1 and configure MA2 with the port type.

```
Switch(config)#ethernet cfm domain operator_A  level 3
Switch(config-ecfm)#service MA2 port direction down
```

Use the same method to create operator_A and MA2 on S5.

(6) Create local and remote MEPs in MA2 of operator_A on S1 and S5.

On S1, create MEP list 1-2, configure RMEP 2, and create MEP 1 on Ethernet1/2.

```
Switch(config-ecfm-srv)#mep mepid 1-2
Switch(config-ecfm-srv)#continuity-check receive rmep 2
Switch(config-ecfm-srv)exit
Switch(config-ecfm)exit
Switch(config)#interface ethernet 1/2
Switch(config-if-ethernet1/1)#ethernet cfm mep 1 domain customer_A service
MA2
```

On S2, create MEP list 1-2, configure RMEP 1, and create MEP 2 on Ethernet1/2.

```
Switch(config-ecfm-srv)#mep mepid 1-2
Switch(config-ecfm-srv)#continuity-check receive rmep 1
Switch(config-ecfm-srv)exit
Switch(config-ecfm)exit
Switch(config)#interface ethernet 1/2
Switch(config-if-ethernet1/1)#ethernet cfm mep 2 domain customer_A service
MA2
```

(7) Enable CCM sending and receiving in MA1 on S1, S2, S3, and S4.

Enable CCM sending and receiving in MA1 on S1.

```
Switch(config-ecfm-srv)#continuity-check receive enable
```

Enable the CC function on the other devices in the same way.

Enable CCM sending and receiving in MA on S1 and S5.

```
Switch(config-ecfm-srv)#continuity-check receive enable
```

(8) Check the MEP configuration of MA1 in customer_A on S1.

```
Switch#show ethernet cfm maintenance-points local detail mep domain
customer_A
Mepid:1
Port:Ethernet1/1                    Active:1
Domain Name: customer_A
Service Name:MA1
Level:6
Vlan:1                                  Direction:Up
------------------------------------------------------------------------------
--------------------------------
CCM:
CC Send:Enable
CC Received:Enable                      Interval:1(s)
```

3.22.9 CFM Troubleshooting

When CFM-OAM is configured, errors may occur and it may not operate normally because of physical connection problems or configuration errors.

☞ Make sure the entire link is connected normally and the VLANs associated with the MA exist.

1. **Configuration Failure**

☞ Make sure the global OAM function is enabled on the system; otherwise, no OAM-related command can be configured.

☞ Invalid MA/MD name:

MD name: a string of 1-43 characters. It can contain letters, digits, and underscores, and the first and last characters cannot be underscores.

MA name: a string of 1-43 characters. It can contain letters, digits, and underscores, and the first and last characters cannot be underscores. The combined length of the MA name and the domain name cannot exceed 44 characters.

☞ At the same level, a primary VLAN can be associated with only one MA.

☞ The MD level, the primary VLAN associated with the MA, and the Up/Down attribute cannot be changed after creation.

☞ The MEP ID must be created before the MEP and RMEP are configured.

☞ After a MEP ID in an MA is configured as a MEP, it cannot be configured as an RMEP.

☞ A Down MA on a device allows only one MEP; an Up MA allows multiple MEPs, but multiple MEPs of the same MA cannot be configured on the same port.

2. **Cannot establish the CC connection**

☞ Use the ***show ethernet cfm*** commands to make sure that the level, MD name, MA name, and MA-associated primary VLAN are the same on both ports.

☞ Make sure the RMEP configured on this port matches the MEP ID configured on the peer port.

☞ Make sure CC sending and detection are enabled on both this port and the peer port.

☞ Make sure the CCM sending interval is the same on both ports.

- ☞ If a Down MEP is configured on a port, the MEP receives messages from this port. If an Up MEP is configured, the MEP receives messages from other ports. Make sure the Up MEP is configured on a non-receiving port.
- ☞ A port blocked by STP cannot receive, send, or reply to CFM messages. However, if the port is configured as a MEP, it can still send and receive CCMs even when blocked by STP. Only Layer 2 Ethernet ports support the CFM function.
- ☞ MEPs and MIPs can be configured on a port channel, but MEPs and MIPs configured on the port channel do not take effect on its member ports. If a port with CFM MEPs and MIPs is added to a port channel, that port's MEPs and MIPs also stop taking effect. To restore the MEPs and MIPs on the port, remove the port from the port channel.

3. **Cannot create the MIP**

- ☞ Make sure the MIP creation rule is correct.

Default rule: If there is no higher-level MEP on the port and no lower-level MIP exists, a MIP is created on the port at this level.

Explicit rule: If there is no higher-level MEP on the port and no lower-level MIP exists, whether a MIP is created depends on whether a lower-level MEP exists.

- ☞ A MIP is created only when the port is up; only one MIP is created per VLAN on a port, and lower-level MIPs have higher creation priority.
- ☞ A Down MA needs to be configured only on the port. If a MEP is configured on the port, no MIP can be created; if the port is added to a port channel, the MEP stops taking effect. A MIP cannot be created in the MA.

If the CFM OAM problem cannot be solved after these checks, use the ***debug ethernet cfm*** commands, collect the debug information (3 minutes), and send it to EnGenius technical support.

Chapter 4 IP Services Configuration

4.1 Layer 3 Interface

4.1.1 Introduction to Layer 3 Interfaces

Layer 3 interfaces can be created on the switch. A Layer 3 interface is not a physical interface but a virtual interface built on a VLAN. A Layer 3 interface can contain one or more Layer 2 ports that belong to the same VLAN, or no Layer 2 ports at all. A Layer 3 interface is up only when at least one of its Layer 2 ports is up; otherwise, it is down. By default, all Layer 3 interfaces on the switch use the same MAC address, which is selected from the reserved MAC addresses when the Layer 3 interface is created. Layer 3 interfaces are the basis of Layer 3 protocols. The switch uses the IP addresses configured on its Layer 3 interfaces to communicate with other devices over IP, and it can forward IP packets between different Layer 3 interfaces. A loopback interface is also a Layer 3 interface.

4.1.2 Layer 3 Interface Configuration Task List

Layer 3 Interface Configuration Task List:

1. Create a Layer 3 interface
2. Configure the bandwidth of a Layer 3 interface
3. Configure a VLAN interface description
4. Enable or disable a VLAN interface
5. Configure VRF

(1) Create a VRF instance and enter VRF Mode

(2) Configure the RD of the VRF instance (optional)

(3) Configure the RT of the VRF instance (optional)

(4) Associate the VRF instance with an interface

1. Create a Layer 3 interface

Command	Explanation
Global Mode	
interface vlan <*vlan-id*> **no interface vlan <*vlan-id*>**	Creates a VLAN interface (a Layer 3 interface). The no command deletes the VLAN interface (Layer 3 interface) from the switch.
interface loopback <*loopback-id*> **no interface loopback <*loopback-id*>**	Creates a loopback interface and enters Loopback Interface Mode. The no command deletes the loopback interface from the switch.

2. Configure the bandwidth of a Layer 3 interface

Command	Explanation
VLAN Interface Mode	
bandwidth <*bandwidth*> **no bandwidth**	Configures the bandwidth of a Layer 3 interface. The no command restores the default value.

3. Configure a VLAN interface description

Command	Explanation
VLAN Interface Mode	
description <*text*>	Configures a description for the VLAN interface. The no command deletes the

no description	description of the VLAN interface.

4. Enable or disable a VLAN interface

Command	Explanation
VLAN Interface Mode	
shutdown **no shutdown**	Disables or enables the VLAN interface. The no command enables the VLAN interface.

5. Configure VRF

(1) Create a VRF instance and enter VRF Mode

(2) Configure the RD of the VRF instance (optional)

(3) Configure the RT of the VRF instance (optional)

(4) Associate the VRF instance with an interface

Command	Explanation
Global Mode	
ip vrf <vrf-name> **no ip vrf <vrf-name>**	Creates a VRF instance. No VRF instance exists by default. The no command deletes the VRF instance.
VRF Mode	
rd <ASN:nn_or_IP-address:nn>	Configures the RD of the VRF instance. No RD is configured by default.
route-target {import | export | both} <rt-value> **no route-target {import | export | both} <rt-value>**	Configures the RT of the VRF instance. The no command deletes the RT.
Interface Mode	
ip vrf forwarding <vrf-name> **no ip vrf forwarding <vrf-name>**	Associates the VRF instance with the interface. The no command removes the association.
ip address <ip-address> <mask> **no ip address <ip-address> <mask>**	Configures the private IP address of the directly connected interface. The no command deletes the IP address.

4.2 Network Management Port

4.2.1 Introduction to the IP Network Management Port

The network management port is located above the Console port on the master management cards and is labeled "Ethernet"; its name in the software configuration is "Ethernet0". Use the command "interface Ethernet 0" to enter the network management port configuration mode. Programs such as Telnet, Web management, and FTP can manage the switch through an Ethernet cable connected to the network management port.

4.2.2 Network Management Port Configuration

Network Management Port Configuration Task List:

1. Enter the network management port configuration mode
2. Configure the properties of the network management port
   （1） Enable or disable the port
   （2） Configure the port IP address

1. Enter the network management port configuration mode

Command	Explanation
Global Mode	
interface ethernet <*num*>	Enters the network management port configuration mode.

2. Configure the properties of the network management port

Command	Explanation
Network Management Port Mode	
shutdown **no shutdown**	Disables or enables the network management port. The no command enables the port.
ip address <*ip-address*> <*mask*> **no ip address [<*ip-address*> <*mask*>]**	Configures the IP address of the network management port. The no command deletes the IP address.

4.3 IP Configuration

4.3.1 Introduction to IPv4 and IPv6

IPv4 is the current version of the universal Internet Protocol. Practice has proved that IPv4 is simple, flexible, open, stable, robust, and easy to implement, and that it works well with the various protocols of the upper and lower layers. Although IPv4 has hardly changed since it was established in the 1980s, it has grown to the current global scale along with the expansion of the Internet. However, as Internet infrastructure and application services continue to grow, IPv4 has shown its limitations in the face of the present scale and complexity of the Internet.

IPv6 is the sixth version of the Internet Protocol, the next-generation Internet Protocol designed by the IETF to replace IPv4. IPv6 was developed specifically to overcome the shortage of IPv4 addresses so that the Internet can continue to develop.

The most important problem that IPv6 solves is increasing the number of IP addresses. IPv4 addresses have nearly run out, while the number of Internet users has been increasing geometrically. With the continuous and rapid growth of Internet services and of devices that require IP addresses (home and small office networks, IP phones, and wireless information terminals that use the Internet), the supply of IP addresses is becoming increasingly tight. The IPv4 address shortage has long been addressed by technologies that extend the lifespan of the existing IPv4 infrastructure, such as Network Address Translation (NAT) and Classless Inter-Domain Routing (CIDR).

Although the combination of CIDR, NAT, and private addressing has temporarily mitigated the IPv4 address shortage, NAT breaks the end-to-end model that was the original intention of the IP design: routers acting as intermediate network nodes must maintain the state of every connection, which greatly increases network delay and reduces network performance. Moreover, translating packet addresses interferes with end-to-end network security checks; the IPSec authentication header is one example.

Therefore, to solve the various problems of IPv4 comprehensively, IPv6, the next-generation Internet Protocol designed by the IETF, is currently the only feasible solution.

First, the 128-bit addressing scheme of IPv6 guarantees enough globally unique IP addresses for IP network nodes worldwide, now and in the future. In addition to increasing the address space, IPv6 also improves many other essential aspects of the IPv4 design.

The hierarchical addressing scheme facilitates route aggregation, effectively reduces routing table entries, and improves the efficiency and scalability of routing and packet processing.

The IPv6 header is more efficient than the IPv4 header. It has fewer fields and no header checksum, which speeds up processing of the basic IPv6 header. In IPv6, the fragment field is an optional

extension header, so packets are not fragmented during router forwarding; the Path MTU Discovery mechanism works with the packet source to improve router processing efficiency.

Stateless address autoconfiguration and plug-and-play are supported. Through IPv6 address autoconfiguration, large numbers of hosts can easily discover network routers and automatically obtain globally unique IPv6 addresses, which makes IPv6 devices plug-and-play. Address autoconfiguration also makes renumbering existing networks easier and more convenient, and makes it easier for network operators to migrate from one provider to another.

IPSec is supported. IPSec is optional in IPv4 but required in IPv6. IPv6 provides a security extension header that offers end-to-end security services such as access control, confidentiality, and data integrity, making encryption, authentication, and Virtual Private Networks easier to implement.

Support for Mobile IP and mobile computing devices is enhanced. The Mobile IP protocol defined in IETF standards lets mobile devices move without breaking existing connections, which is an increasingly important network function. Unlike IPv4, IPv6 mobility obtains the transmission address (Care-of Address) through built-in autoconfiguration, so no Foreign Agent is needed. Furthermore, this binding process enables a Correspondent Node to communicate with a Mobile Node directly, which avoids the extra system overhead caused by the triangle routing required in IPv4.

Network Address Translation is no longer needed. NAT was introduced to share and reuse the same address space among different network segments. It temporarily mitigates the IPv4 address shortage, but it adds address translation overhead to network devices and applications. Because the IPv6 address space is so much larger, address translation becomes unnecessary, and the problems and overhead of NAT deployment disappear.

Widely deployed routing protocols are supported. IPv6 retains and extends support for existing Interior Gateway Protocols (IGP) and Exterior Gateway Protocols (EGP), for example through the IPv6 routing protocols RIPng, OSPFv3, IS-ISv6, and MBGP4+.

Multicast addresses are increased and multicast support is enhanced. By taking over IPv4 broadcast functions such as Router Discovery and Router Query, IPv6 multicast completely replaces IPv4 broadcast in terms of function. Multicast not only saves network bandwidth but also improves network efficiency.

4.3.2 IP Configuration

A Layer 3 interface can be configured as an IPv4 interface or an IPv6 interface.

4.3.2.1 IPv4 Address Configuration

IPv4 Address Configuration Task List:

1. Configure the IPv4 address of a Layer 3 interface

1. Configure the IPv4 address of a Layer 3 interface

Command	Explanation
VLAN Interface Mode	
ip address <*ip-address*> <*mask*> [secondary] **no ip address [<*ip-address*> <*mask*>]**	Configures the IP address of the VLAN interface. The **no ip address [<*ip-address*> <*mask*>]** command deletes the IP address of the VLAN interface.

4.3.2.2 IPv6 Address Configuration

IPv6 Configuration Task List:

1. Perform basic IPv6 configuration

(1) Configure an interface IPv6 address

(2) Configure IPv6 static routes

2. Configure IPv6 Neighbor Discovery

(1) Configure the number of DAD neighbor solicitation messages

(2) Configure the neighbor solicitation message interval

(3) Enable or disable router advertisement

(4) Configure the router lifetime

(5) Configure the minimum router advertisement interval

(6) Configure the maximum router advertisement interval

(7) Configure prefix advertisement parameters

(8) Configure static IPv6 neighbor entries

(9) Delete all entries in the IPv6 neighbor table

(10) Set the hop limit of sent router advertisements

(11) Set the MTU of sent router advertisements

(12) Set the reachable time of sent router advertisements

(13) Set the retransmission timer of sent router advertisements

(14) Set the flag that indicates whether information other than addresses is obtained through DHCPv6

(15) Set the flag that indicates whether addresses are obtained through DHCPv6

3. Configure IPv6 tunnels

(1) Create or delete a tunnel

(2) Configure a tunnel description

(3) Configure the tunnel source

(4) Configure the tunnel destination

(5) Configure the tunnel next hop

(6) Configure the tunnel mode

(7) Configure tunnel routes

1. Perform basic IPv6 configuration

(1) Configure an interface IPv6 address

Command	Explanation
Interface Mode	
ipv6 address <*ipv6-address*/*prefix-length*> [eui-64] **no ipv6 address <*ipv6-address/prefix-length*>**	Configures an IPv6 address, including aggregatable global unicast addresses, site-local addresses, and link-local addresses. The **no ipv6 address <*ipv6-address/prefix-length*>** command deletes the IPv6 address.

(2) Configure IPv6 static routes

Command	Explanation
Global Mode	

ipv6 route <ipv6-prefix/prefix-length> {<nexthop-ipv6-address>| <interface-type interface-number> | {<nexthop-ipv6-address> <interface-type interface-number>}} [distance] **no ipv6 route <ipv6-prefix/prefix-length> {<nexthop-ipv6-address>| <interface-type interface-number> |{<nexthop-ipv6-address> <interface-type interface-number>}} [distance]**	Configures an IPv6 static route. The no command deletes the IPv6 static route.

2. Configure IPv6 Neighbor Discovery

(1) Configure the number of DAD neighbor solicitation messages

Command	Explanation
Interface Mode	
ipv6 nd dad attempts <value> **no ipv6 nd dad attempts**	Sets the number of consecutive neighbor solicitation messages the interface sends during duplicate address detection. The no command restores the default value (1).

(2) Configure the neighbor solicitation message interval

Command	Explanation
Interface Mode	
ipv6 nd ns-interval <*seconds*> **no ipv6 nd ns-interval**	Sets the interval at which the interface sends neighbor solicitation messages. The no command restores the default value (1 second).

(3) Enable or disable router advertisement

Command	Explanation
Interface Mode	
ipv6 nd suppress-ra **no ipv6 nd suppress-ra**	Disables IPv6 router advertisement. The no command enables IPv6 router advertisement.

(4) Configure the router lifetime

Command	Explanation
Interface Mode	
ipv6 nd ra-lifetime <*seconds*> **no ipv6 nd ra-lifetime**	Configures the router lifetime in router advertisements. The no command restores the default value (1800 seconds).

(5) Configure the minimum router advertisement interval

Command	Explanation
Interface Mode	
ipv6 nd min-ra-interval <seconds> **no ipv6 nd min-ra-interval**	Configures the minimum interval between router advertisements. The no command restores the default value (200 seconds).

(6) Configure the maximum router advertisement interval

Command	Explanation
Interface Mode	
ipv6 nd max-ra-interval <*seconds*> **no ipv6 nd max-ra-interval**	Configures the maximum interval between router advertisements. The no command restores the default value (600 seconds).

(7) Configure prefix advertisement parameters

Command	Explanation
Interface Mode	
ipv6 nd prefix *<ipv6-address/prefix-length>* *<valid-lifetime>* *<preferred-lifetime>* [off-link] [no-autoconfig] **no ipv6 nd prefix *<ipv6-address/prefix-length>* *<valid-lifetime>* *<preferred-lifetime>* [off-link] [no-autoconfig]**	Configures the address prefix and advertisement parameters of the router. The no command deletes the address prefix from router advertisements.

(8) Configure static IPv6 neighbor entries

Command	Explanation
Interface Mode	
ipv6 neighbor *<ipv6-address>* *<hardware-address>* interface *<interface-type interface-name>*	Sets a static neighbor table entry, including the neighbor IPv6 address, MAC address, and Layer 2 port.
no ipv6 neighbor *<ipv6-address>*	Deletes the neighbor table entry.

(9) Delete all entries in the IPv6 neighbor table

Command	Explanation
Admin Mode	
clear ipv6 neighbors	Clears all static neighbor table entries.

(10) Set the hop limit of sent router advertisements

Command	Explanation
Interface Mode	
ipv6 nd ra-hoplimit <value>	Sets the hop limit of sent router advertisements.

(11) Set the MTU of sent router advertisements

Command	Explanation
Interface Mode	
ipv6 nd ra-mtu <value>	Sets the MTU of sent router advertisements.

(12) Set the reachable time of sent router advertisements

Command	Explanation
Interface Mode	
ipv6 nd reachable-time <seconds>	Sets the reachable time of sent router advertisements.

(13) Set the retransmission timer of sent router advertisements

Command	Explanation
Interface Mode	
ipv6 nd retrans-timer <seconds>	Sets the retransmission timer of sent router advertisements.

(14) Set the flag that indicates whether information other than addresses is obtained through DHCPv6

Command	Explanation
Interface Mode	
ipv6 nd other-config-flag	Sets the flag that indicates whether information other than addresses is obtained through DHCPv6.

(15) Set the flag that indicates whether addresses are obtained through DHCPv6

Command	Explanation

Interface Mode	
ipv6 nd managed-config-flag	Sets the flag that indicates whether addresses are obtained through DHCPv6.

3. Configure IPv6 tunnels

(1) Create or delete a tunnel

Command	Explanation
Global Mode	
interface tunnel <*tnl-id*> **no interface tunnel <*tnl-id*>**	Creates a tunnel. The no command deletes the tunnel.

(2) Configure a tunnel description

Command	Explanation
Tunnel Interface Mode	
description <*desc*> **no description**	Configures the tunnel description. The no command deletes the tunnel description.

(3) Configure the tunnel source

Command	Explanation
Tunnel Interface Mode	
tunnel source { <*ipv4-address*> | **<*ipv6-address*>| <*interface-name*> }** **no tunnel source**	Configures the IPv4/IPv6 address of the tunnel source. The no command deletes the IPv4/IPv6 address of the tunnel source.

(4) Configure the tunnel destination

Command	Explanation
Tunnel Interface Mode	
tunnel destination {<ipv4-address> | <ipv6-address>} **no tunnel destination**	Configures the IPv4/IPv6 address of the tunnel destination. The no command deletes the IPv4/IPv6 address of the tunnel destination.

(5) Configure the tunnel next hop

Command	Explanation
Tunnel Interface Mode	
tunnel nexthop <ipv4-address> **no tunnel nexthop**	Configures the IPv4 address of the tunnel next hop. The no command deletes the IPv4 address of the tunnel next hop.

(6) Configure the tunnel mode

Command	Explanation
Tunnel Interface Mode	
tunnel mode [[gre] | ipv6ip [6to4 | isatap]] **no tunnel mode**	Configures the tunnel mode. The no command clears the tunnel mode.

(7) Configure tunnel routes

Command	Explanation
Global Mode	
ipv6 route <ipv6-address/prefix-length> {<interface-type interface-number> | tunnel <tnl-id>} **no ipv6 route <ipv6-address/prefix-length> {<interface-type interface-number> | tunnel <tnl-id>}**	Configures a tunnel route. The no command deletes the tunnel route.

4.3.3 IP Configuration Examples

4.3.3.1 IPv4 Configuration Example

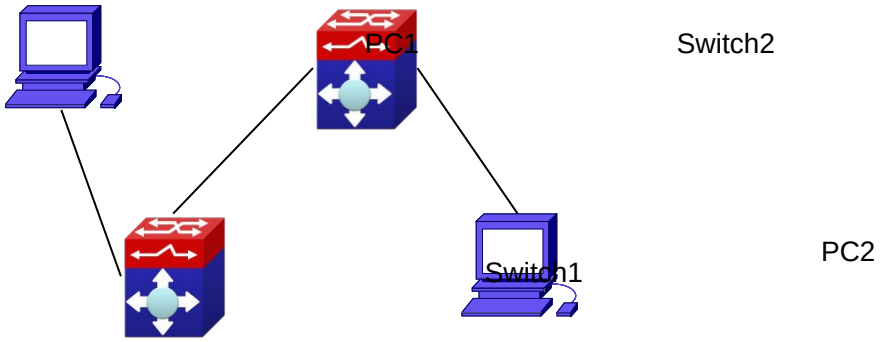


Figure 4-1 IPv4 configuration example

Configuration requirements: Configure IP addresses in different network segments on Switch1 and Switch2, configure static routes, and verify reachability with ping.

Configuration description:

1. Configure two VLANs on Switch1: VLAN1 and VLAN2.
2. Configure IPv4 address 192.168.1.1 255.255.255.0 on VLAN1 of Switch1, and IPv4 address 192.168.2.1 255.255.255.0 on VLAN2.
3. Configure two VLANs on Switch2: VLAN2 and VLAN3.
4. Configure IPv4 address 192.168.2.2 255.255.255.0 on VLAN2 of Switch2, and IPv4 address 192.168.3.1 255.255.255.0 on VLAN3.
5. The IPv4 address of PC1 is 192.168.1.100 255.255.255.0, and the IPv4 address of PC2 is 192.168.3.100 255.255.255.0.
6. Configure static route 192.168.3.0/24 on Switch1, and static route 192.168.1.0/24 on Switch2.
7. Ping between the PCs.

Note: First make sure that PC1 and Switch1 can ping each other, and that PC2 and Switch2 can ping each other.

Configuration procedure:

```
Switch1(config)#interface vlan 1
Switch1(Config-if-Vlan1)#ip address 192.168.1.1 255.255.255.0
Switch1(config)#interface vlan 2
Switch1(Config-if-Vlan2)#ip address 192.168.2.1 255.255.255.0
Switch1(Config-if-Vlan2)#exit
Switch1(config)#ip route 192.168.3.0 255.255.255.0 192.168.2.2
Switch2(config)#interface vlan 2
Switch2(Config-if-Vlan2)#ip address 192.168.2.2 255.255.255.0
Switch2(config)#interface vlan 3
Switch2(Config-if-Vlan3)#ip address 192.168.3.1 255.255.255.0
Switch2(Config-if-Vlan3)#exit
Switch2(config)#ip route 192.168.1.0 255.255.255.0 192.168.2.1
```

4.3.3.2 IPv6 Configuration Examples

Example 1:

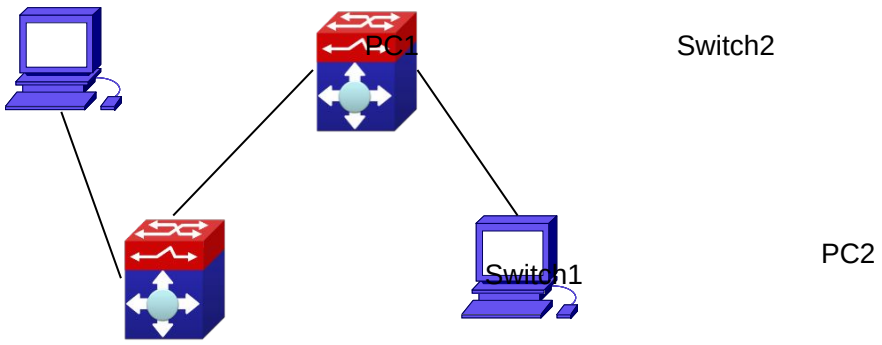


Figure 4-2 IPv6 configuration example

Configuration requirements: Configure IPv6 addresses in different network segments on Switch1 and Switch2, configure static routes, and verify reachability with ping6.

Configuration description:

1. Configure two VLANs on Switch1: VLAN1 and VLAN2.
2. Configure IPv6 address 2001::1/64 on VLAN1 of Switch1, and IPv6 address 2002::1/64 on VLAN2.
3. Configure two VLANs on Switch2: VLAN2 and VLAN3.
4. Configure IPv6 address 2002::2/64 on VLAN2 of Switch2, and IPv6 address 2003::1/64 on VLAN3.
5. The IPv6 address of PC1 is 2001::11/64, and the IPv6 address of PC2 is 2003::33/64.
6. Configure static route 2003:33/64 on Switch1, and static route 2001::11/64 on Switch2.
7. Run ping6 between the PCs.

Note: First make sure that PC1 and Switch1 can ping each other, and that PC2 and Switch2 can ping each other.

Configuration procedure:

```
Switch1(Config)#interface vlan 1
Switch1(Config-if-Vlan1)#ipv6 address 2001::1/64
Switch1(Config)#interface vlan 2
Switch1(Config-if-Vlan2)#ipv6 address 2002::1/64
Switch1(Config-if-Vlan2)#exit
Switch1(Config)#ipv6 route 2003::33/64 2002::2
Switch2(Config)#interface vlan 2
Switch2(Config-if-Vlan2)#ipv6 address 2002::2/64
Switch2(Config)#interface vlan 3
Switch2(Config-if-Vlan3)#ipv6 address 2003::1/64
Switch2(Config-if-Vlan3)#exit
Switch2(Config)#ipv6 route 2001::33/64 2002::1
Switch1#ping6 2003::33
```

Configuration result:

```
Switch1#show run
interface Vlan1
 ipv6 address 2001::1/64
!
interface Vlan2
 ipv6 address 2002::2/64
!
interface Loopback
 mtu 3924
!
ipv6 route 2003::/64 2002::2
```

```
!
no login
!
end
Switch2#show run
interface Vlan2
 ipv6 address 2002::2/64
!
interface Vlan3
 ipv6 address 2003::1/64
!
interface Loopback
 mtu 3924
!
ipv6 route 2001::/64 2002::1
!
no login
!
End
```

Example 2:

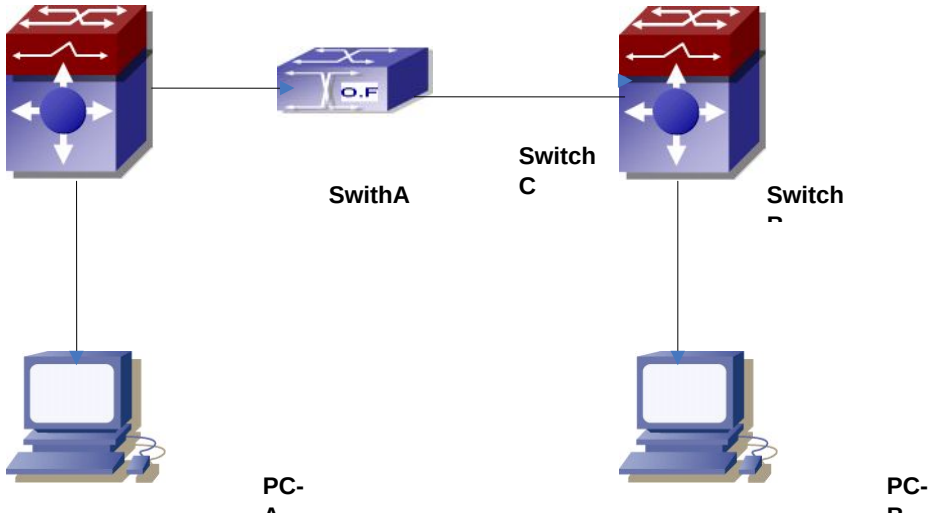


Figure 4-3 IPv6 tunnel

This example shows an IPv6 tunnel with the following configuration requirements: SwitchA and SwitchB are tunnel nodes that support dual stack, SwitchC runs IPv4 only, and PC-A and PC-B communicate with each other.

Configuration description:

1. Configure two VLANs on SwitchA: VLAN1 and VLAN2. VLAN1 is in the IPv6 domain, and VLAN2 connects to the IPv4 domain.
2. Configure IPv6 address 2002:caca:ca01:2::1/64 on VLAN1 of SwitchA and enable RA, and configure IPv4 address 202.202.202.1 on VLAN2.
3. Configure two VLANs on SwitchB: VLAN3 and VLAN4. VLAN4 is in the IPv6 domain, and VLAN3 connects to the IPv4 domain.
4. Configure IPv6 address 2002:cbcb:cb01:2::1/64 on VLAN4 of SwitchB and enable RA, and configure IPv4 address 203.203.203.1 on VLAN3.
5. Configure a tunnel on SwitchA with source IPv4 address 202.202.202.1 and tunnel route ::/0.
6. Configure a tunnel on SwitchB with source IPv4 address 203.203.203.1 and tunnel route ::/0.

7. Configure two VLANs on SwitchC: VLAN2 and VLAN3. Configure IPv4 address 202.202.202.202 on VLAN2 and IPv4 address 203.203.203.203 on VLAN3.
8. PC-A and PC-B obtain the 2002 prefix from SwitchA and SwitchB and configure their IPv6 addresses automatically.
9. On PC-A, ping the IPv6 address of PC-B.

Configuration procedure:

```
SwitchA(Config-if-Vlan1)#ipv6 address 2002:caca:ca01:2::1/64
SwitchA(Config-if-Vlan1)#no ipv6 nd suppress-ra
SwitchA(Config-if-Vlan1)#interface vlan 2
SwitchA(Config-if-Vlan2)#ipv4 address 202.202.202.1 255.255.255.0
SwitchA(Config-if-Vlan1)#exit
SwitchA(config)# interface tunnel 1
SwitchA(Config-if-Tunnel1)#tunnel source 202.202.202.1
SwitchA(Config-if-Tunnel1)#tunnel destination 203.203.203.1
SwitchA(Config-if-Tunnel1)#tunnel mode ipv6ip
SwitchA(config)#ipv6 route ::/0 tunnel1
SwitchB(Config-if-Vlan4)#ipv6 address 2002:cbcb:cb01::2/64
SwitchB(Config-if-Vlan4)#no ipv6 nd suppress-ra
SwitchB (Config-if-Vlan3)#interface vlan 3
SwitchB (Config-if-Vlan2)#ipv4 address 203.203.203.1 255.255.255.0
SwitchB (Config-if-Vlan1)#exit
SwitchB(config)#interface tunnel 1
SwitchB(Config-if-Tunnel1)#tunnel source 203.203.203.1
SwitchB(Config-if-Tunnel1)#tunnel destination 202.202.202.1
SwitchB(Config-if-Tunnel1)#tunnel mode ipv6ip
SwitchB(config)#ipv6 route ::/0 tunnel1
```

4.3.4 IPv6 Troubleshooting

☝ The configured router lifetime must not be shorter than the router advertisement interval. If a connected PC has not obtained an IPv6 address, check whether RA is enabled (it is disabled by default).

4.4 IP Forwarding

4.4.1 Introduction to IP Forwarding

Gateway devices can forward IP packets from one subnet to another; such forwarding uses routes to find a path. IP forwarding on the switch is performed in hardware and can achieve wire-speed forwarding. Flexible management is also provided to adjust and monitor forwarding. The switch can enable or disable optimization of the route aggregation algorithm to adjust how route entries are generated in the switch chip, and can display statistics for IP forwarding and the status of the hardware forwarding chip.

4.4.2 IP Route Aggregation Configuration Task List

IP Route Aggregation Configuration Task List:

1. Specify whether the IP route aggregation algorithm uses optimization

1. Specify whether the IP route aggregation algorithm uses optimization

Command	Explanation
Global Mode	
ip fib optimize **no ip fib optimize**	Enables the optimized IP route aggregation algorithm on the switch. The **no ip fib optimize** command disables the optimized IP

	route aggregation algorithm.

4.5 URPF

4.5.1 Introduction to URPF

URPF (Unicast Reverse Path Forwarding) applies the RPF technology used in multicast to unicast, protecting the network from attacks based on source address spoofing.

When the switch receives a packet, it looks up the routing table using the packet's source address as the destination address. If the outgoing interface of the route found does not match the interface on which the packet was received, the switch considers the packet spoofed and discards it.

In source address spoofing attacks, attackers construct a series of packets with forged source addresses. For applications that rely on IP address verification, such attacks may allow unauthorized users to access the system as authorized users, or even as the administrator. Even if the response packets cannot reach the attackers, they still damage the targets.

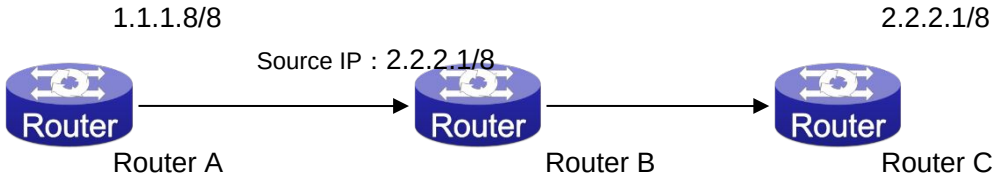


Figure 4-4 URPF application scenario

In the figure above, Router A sends requests to server Router B using forged packets with the source address 2.2.2.1/8. In response, Router B sends packets to the real "2.2.2.1/8". These illegal packets attack both Router B and Router C. Applying URPF in this scenario prevents attacks based on source address spoofing.

4.5.2 URPF Configuration Task List

1. Enable URPF
2. Display and debug URPF information

1. Enable URPF globally

Command	Explanation
Global Mode	
urpf enable **no urpf enable**	Enables URPF globally. The no command disables URPF globally.

2. Display and debug URPF information

Command	Explanation
Admin Mode and Configuration Mode	
show urpf	Displays the interfaces on which URPF is enabled.

4.5.3 URPF Typical Example

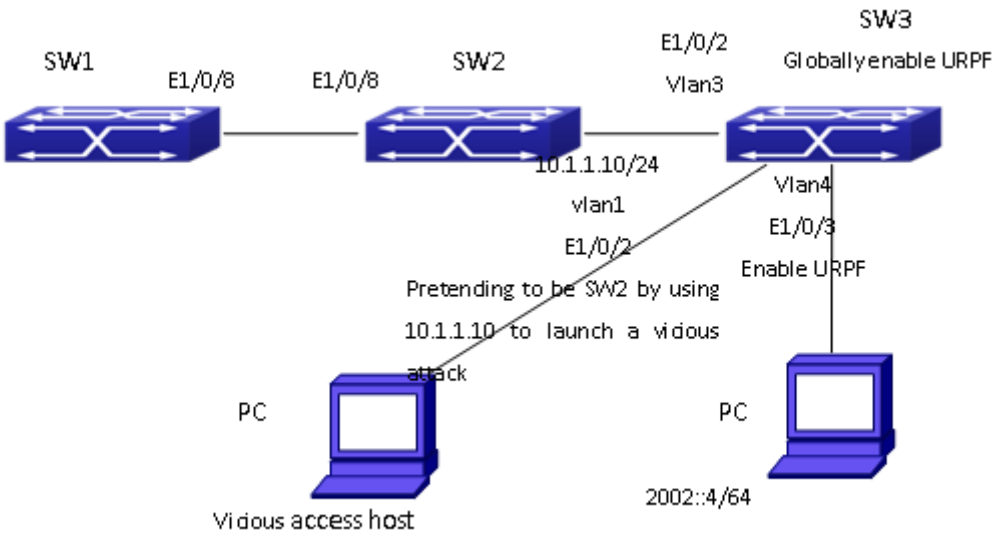


Figure 4-5 URPF typical example

In the network topology shown above, IP URPF is enabled on SW3. When someone in the network launches an attack by impersonating another host's IP address, the switch drops all attack packets directly in hardware.

Enable URPF on SW3.

```
SW3 configuration task sequence:
Switch3#config
Switch3(config)#urpf enable
```

4.5.4 URPF Troubleshooting

If all configurations are correct but URPF still does not work as expected, enable URPF debugging and use the 'show urpf' command to check whether URPF is enabled. If the problem persists, contact EnGenius technical support and provide the results.

4.6 ARP

4.6.1 Introduction to ARP

ARP (Address Resolution Protocol) is mainly used to resolve IP addresses to Ethernet MAC addresses. The switch supports both dynamic and static ARP configuration. The switch also supports proxy ARP for some applications. For example, when a port with proxy ARP enabled receives an ARP request for an IP address in the same IP segment as the port but not on the same physical network, the port replies with its own MAC address and forwards the actual packets it receives. Proxy ARP allows physically separated hosts in the same IP segment to ignore the physical separation and communicate through the proxy ARP interface as if they were on the same physical network.

4.6.2 ARP Configuration Task List

ARP Configuration Task List:

1. Configure static ARP
2. Configure proxy ARP
3. Clear dynamic ARP

4. Clear ARP message statistics

1. Configure static ARP

Command	Explanation
VLAN Interface Mode	
arp <*ip_address*> <*mac_address*> {interface [ethernet] <portName>} **no arp <ip_address>**	Configures a static ARP entry. The no command deletes the ARP entry of the specified IP address.

2. Configure proxy ARP

Command	Explanation
VLAN Interface Mode	
ip proxy-arp **no ip proxy-arp**	Enables proxy ARP on the interface. The no command disables proxy ARP.

3. Clear dynamic ARP

Command	Explanation
Admin Mode	
clear arp-cache	Clears the dynamic ARP entries learned by the switch.

4. Clear ARP message statistics

Command	Explanation
Admin Mode	
clear arp traffic	Clears the ARP message statistics of the switch.

4.6.3 ARP Troubleshooting

If pinging a directly connected network device from the switch fails, use the following checks to find the possible cause and a solution.

- ☞ Check whether the switch has learned the corresponding ARP entry.
- ☞ If the ARP entry has not been learned, enable ARP debugging and check how ARP packets are sent and received.
- ☞ A defective cable is a common cause of ARP problems and can prevent ARP learning.

4.7 Hardware Tunnel Capacity

4.7.1 Introduction to Hardware Tunnel Capacity

Hardware tunnel capacity is the maximum number of tunnel and MPLS entries forwarded in hardware. The capacity can be adjusted with this command. Increasing the capacity reduces the number of hardware routes the switch supports, and vice versa.

4.7.2 Hardware Tunnel Capacity Configuration

Hardware Tunnel Capacity Configuration Task List:

1. Configure the hardware tunnel capacity

Command	Explanation
Global Mode	

hardware tunnel-capacity < *size*> **no hardware tunnel-capacity**	Configures the hardware capacity for tunnels and MPLS. The no command restores the default capacity.

Note: After adjusting the hardware tunnel capacity, restart the switch for the configuration to take effect.

4.7.3 Hardware Tunnel Capacity Troubleshooting

☞ After adjusting the hardware tunnel capacity, save the configuration and restart the switch for the configuration to take effect.

4.8 ARP Scanning Prevention

4.8.1 Introduction to ARP Scanning Prevention Function

ARP scanning is a common network attack method. To detect all active hosts in a network segment, the attack source broadcasts large numbers of ARP messages in the segment, which consume a large part of the network bandwidth. An attacker may even launch a high-traffic attack with forged ARP messages that exhausts the bandwidth and brings down the network. ARP scanning is usually the prelude to more dangerous attacks, such as automatic virus infection, subsequent port scanning, vulnerability scanning aimed at stealing information, malformed packet attacks, and DoS attacks.

Because ARP scanning seriously threatens network security and stability, preventing it is very important. The switch provides a complete solution: if any host or port in the segment shows ARP scanning behavior, the switch cuts off the attack source to keep the network secure.

There are two methods of ARP scanning prevention: port-based and IP-based. Port-based prevention counts the ARP messages received on a port within a certain time; if the number exceeds a preset threshold, the port is shut down. IP-based prevention uses two threshold levels: rate limiting and isolation. When the rate exceeds the level-1 threshold (the rate-limiting threshold), the hardware still forwards the host's ARP packets (including ARP requests and replies) normally and only limits the rate to the CPU, and a trap is generated to warn the administrator of a possible attack. When the rate reaches the level-2 threshold (the isolation threshold), the switch takes action, records a log, and generates a trap. Both the level-1 and level-2 thresholds are activated when IP-based ARP scanning prevention is enabled in Global Mode; the level-1 threshold takes effect only when it is lower than the level-2 threshold. Both kinds of ARP scanning prevention can be used at the same time. A port that has been shut down can be recovered by configuring automatic recovery. A blocked IP address is recovered automatically when the rate of received ARP packets falls below the level-2 threshold.

To improve switch performance, configure trusted ports and IP addresses; the switch does not check ARP messages from them. This effectively reduces the switch load.

4.8.2 ARP Scanning Prevention Configuration Task List

1. Enable ARP scanning prevention
2. Configure the thresholds of port-based and IP-based ARP scanning prevention
3. Configure trusted ports
4. Configure trusted IP addresses
5. Configure the automatic recovery time
6. Display debugging and ARP scanning information
7. Configure the action taken when the level-1 threshold is exceeded

1. Enable ARP scanning prevention

Command	Explanation

Global Mode	
anti-arpscan enable [ip|port] **no anti-arpscan enable [ip|port]**	Enables ARP scanning prevention globally. The no command disables it.

2. Configure the thresholds of port-based and IP-based ARP scanning prevention

Command	Explanation
Global Mode	
anti-arpscan port-based threshold <*threshold-value*> **no anti-arpscan port-based threshold**	Sets the threshold of port-based ARP scanning prevention. The no command restores the default threshold.
anti-arpscan ip-based threshold <*threshold-value*> **no anti-arpscan ip-based threshold**	Sets the threshold of IP-based ARP scanning prevention. The no command restores the default threshold.

3. Configure trusted ports

Command	Explanation
Port Mode	
anti-arpscan trust {port | supertrust-port | iptrust-port} **no anti-arpscan trust {port | supertrust-port | iptrust-port}**	Sets the trust attribute of the port. The no command removes the trust attribute.

4. Configure trusted IP addresses

Command	Explanation
Global Mode	
anti-arpscan trust ip <*ip-address*> [<*netmask*>] **no anti-arpscan trust ip <*ip-address*> [<*netmask*>]**	Sets the trust attribute of the IP address. The no command removes the trust attribute.

5. Configure the automatic recovery time

Command	Explanation
Global Mode	
anti-arpscan recovery enable **no anti-arpscan recovery enable**	Enables automatic recovery. The no command disables automatic recovery.
anti-arpscan recovery time <*seconds*> **no anti-arpscan recovery time**	Sets the automatic recovery time. The no command restores the default recovery time.

6. **6. Display debugging and ARP scanning information**

anti-arpscan log enable **no anti-arpscan log enable**	Enables logging for ARP scanning prevention. The no command disables logging.
anti-arpscan trap enable **no anti-arpscan trap enable**	Enables SNMP traps for ARP scanning prevention. The no command disables the traps.
anti-arpscan FFP max-num <num>	Sets the maximum number of FFP entries that ARP scanning prevention can occupy.
show anti-arpscan [trust {ip | port | supertrust-port | iptrust-port} | prohibited {ip | port}]	Displays the operating status and configuration of ARP scanning prevention.
show anti-arpscan ip-based attack-list [history]	Displays the current or historical source information of ARP scanning attacks.
show anti-arpscan ip-based running-config	Displays the current configuration of ARP scanning prevention.

clear anti-arpscan speed-limit< IP Address>	Manually clears the ARP rate limit for the specified host.
clear anti-arpscan ip-isolate<IP Address>	Manually clears the IP service isolation for the specified host.
clear anti-arpscan attack-list {ip < IP Address > | all }	Manually clears the ARP limit for the specified host or for all hosts.
clear anti-arpscan attack-history-list {ip < IP Address > | all }	Manually clears the historical attack source information of ARP scanning prevention.
Admin Mode	
debug anti-arpscan [port | ip] **no debug anti-arpscan [port | ip]**	Enables debugging for ARP scanning prevention. The no command disables debugging.

7. **7. Configure the action taken when the level-1 threshold is exceeded**

Command	Explanation
Global Mode	
anti-arpscan ip-based arp-to-cpu speed <pps> **no anti-arpscan ip-based arp-to-cpu speed**	Configures the rate at which ARP packets are sent to the CPU when the level-1 threshold is exceeded. The no command restores the default rate.

4.8.3 ARP Scanning Prevention Typical Examples

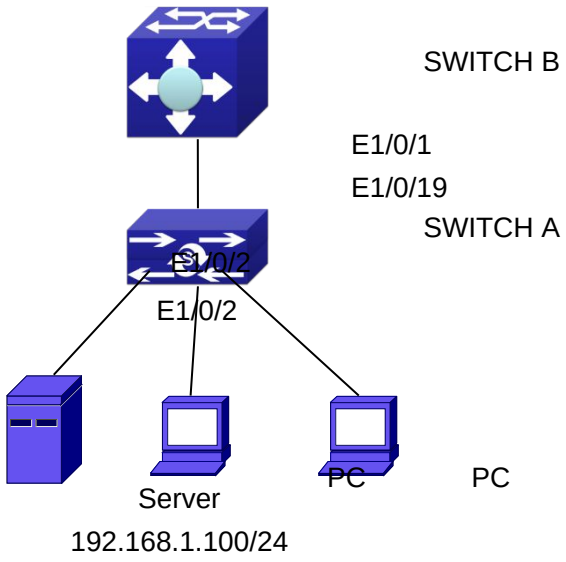


Figure 4-6 ARP scanning prevention typical configuration example

In the network topology above, port E1/0/1 of SWITCH B is connected to port E1/0/19 of SWITCH A, port E1/0/2 of SWITCH A is connected to a file server (IP address 192.168.1.100/24), and all other ports of SWITCH A are connected to ordinary PCs. The following configuration prevents ARP scanning effectively without affecting normal system operation.

SWITCH A configuration:

```
SwitchA(config)#anti-arpscan enable
SwitchA(config)#anti-arpscan recovery time 3600
SwitchA(config)#anti-arpscan trust ip 192.168.1.100 255.255.255.0
SwitchA(config)#interface ethernet1/0/2
SwitchA (Config-If-Ethernet1/0/2)#anti-arpscan trust port
SwitchA (Config-If-Ethernet1/0/2)#exit
SwitchA(config)#interface ethernet1/0/19
SwitchA (Config-If-Ethernet1/0/19)#anti-arpscan trust supertrust-port
Switch A(Config-If-Ethernet1/0/19)#exit
```

SWITCH B configuration:

```
Switch B(config)# anti-arpscan enable
SwitchB(config)#interface ethernet1/0/1
SwitchB(Config-If-Ethernet1/0/1)#anti-arpscan trust port
SwitchB(Config-If-Ethernet1/0/1)exit
```

4.8.4 ARP Scanning Prevention Troubleshooting

☞ ARP scanning prevention is disabled by default. After enabling ARP scanning prevention, enable debugging with the "**debug anti-arpscan**" command to view debugging information.

4.9 Prevent ARP Spoofing

4.9.1 Overview

4.9.1.1 ARP (Address Resolution Protocol)

ARP (RFC 826) is mainly responsible for mapping an IP address to the corresponding 48-bit physical address, that is, the MAC address; for example, IP address 192.168.0.1 maps to network card MAC address 00-03-0F-FD-1D-2B. In the mapping process, a host broadcasts an ARP request containing the IP address of the destination host, and the destination host replies with a packet containing its IP address and MAC address. The two hosts can then exchange data by MAC address.

4.9.1.2 ARP Spoofing

To reduce redundant ARP traffic on the network, the ARP protocol is designed so that a host adds an entry to its ARP cache even when it receives an ARP reply it did not request. This makes "ARP spoofing" possible. A hacker who wants to snoop on the communication between two hosts on the same network (even if they are connected through switches) sends a separate ARP reply to each host, making each of them believe that the other's MAC address is the hacker's MAC address. In this way, what appears to be direct communication actually passes through the hacker's host. The hacker not only obtains the information needed, but also only has to modify some information in the packets and forward them. With this sniffing method, the hacker's host does not need to put its network card into promiscuous mode, because the packets between the two parties are physically sent to the hacker's host, which acts as a relay.

4.9.1.3 How to Prevent ARP Spoofing

There are many sniffing, monitoring, and attack behaviors based on ARP on networks, and most of them rely on ARP spoofing, so preventing ARP spoofing is very important. An ARP spoofer first enters the normal network by counterfeiting a legal IP address and then sends large numbers of counterfeit ARP packets to the switches. After learning these packets, the switches overwrite the previously correct IP-to-MAC address mappings with the mappings forged by the attack packets. The switch then forwards packets incorrectly, which affects the whole network. Alternatively, malicious attackers exploit the switches to intercept and capture the packets they forward, or to attack other switches, hosts, or network equipment.

The essential way to prevent ARP-based attacks and spoofing against switches is to disable automatic ARP updating on the switch, so that a spoofer cannot modify the correct MAC addresses, which avoids incorrect packet forwarding and prevents the spoofer from obtaining other information. At the same time, automatic ARP learning is not interrupted. This prevents ARP spoofing and attacks to a great extent.

4.9.2 Prevent ARP Spoofing Configuration

Prevent ARP Spoofing Configuration Task List:

1. Disable automatic ARP updating
2. Disable automatic ARP learning
3. Convert dynamic ARP entries to static ARP entries

1. Disable automatic ARP updating

Command	Explanation
Global Mode and Port Mode	
ip arp-security updateprotect **no ip arp-security updateprotect**	Disables automatic ARP updating. The no command enables automatic ARP updating.

2. Disable automatic ARP learning

Command	Explanation
Global Mode and Interface Mode	
ip arp-security learnprotect **no ip arp-security learnprotect**	Disables automatic ARP learning. The no command enables automatic ARP learning.

3. Convert dynamic ARP entries to static ARP entries

Command	Explanation
Global Mode and Port Mode	
ip arp-security convert	Converts dynamic ARP entries to static ARP entries.

4.9.3 Prevent ARP Spoofing Example

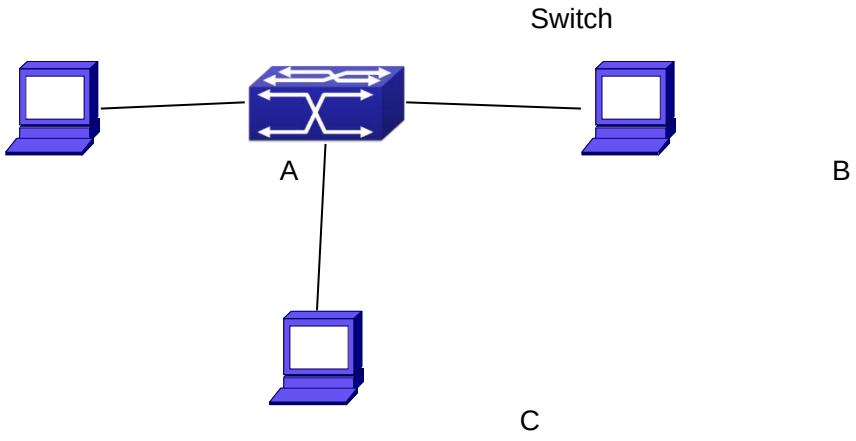


Figure 3-7

Equipment Description

Equipment	Configuration	Quantity
Switch	IP: 192.168.2.4; IP: 192.168.1.4; MAC: 00-00-00-00-00-04	1
A	IP: 192.168.2.1; MAC: 00-00-00-00-00-01	1
B	IP: 192.168.1.2; MAC: 00-00-00-00-00-02	1
C	IP: 192.168.2.3; MAC: 00-00-00-00-00-03	Several

In the diagram above, B and C communicate normally. A wants the switch to forward the packets sent by B to A instead. First, A sends an ARP reply to the switch in the format 192.168.2.3, 00-00-00-00-00-01, mapping its own MAC address to C's IP address. When the switch updates its ARP table,

it changes the mapping, and packets for 192.168.2.3 are then forwarded to 00-00-00-00-00-01 (A's MAC address).

A then forwards the received packets to C after modifying their source and destination addresses, so the data exchanged between B and C passes through A without either party noticing. Because the ARP table is updated periodically, A must also keep sending ARP replies to refresh the switch's ARP table.

Therefore, protecting the ARP table is very important. In a stable environment, disable ARP learning and then convert all dynamic ARP entries to static ARP entries; learned ARP entries are then no longer refreshed, which protects users.

```
Switch#config
Switch(config)#interface vlan 1
Switch(Config-If-Vlan1)#arp 192.168.2.1 00-00-00-00-00-01 interface eth 1/0/2
Switch(Config-If-Vlan1)#interface vlan 2
Switch(Config-If-Vlan2)#arp 192.168.1.2 00-00-00-00-00-02 interface eth 1/0/2
Switch(Config-If-Vlan2#interface vlan 3
Switch(Config-If-Vlan3)#arp 192.168.2.3 00-00-00-00-00-03 interface eth 1/0/2
Switch(Config-If-Vlan3)#exit
Switch(Config)#ip arp-security learnprotect
Switch(Config)#
Switch(config)#ip arp-security convert
```

If the environment changes, disable ARP updating instead: once an ARP entry is learned, it is not refreshed by new ARP replies, which protects user data from sniffing.

```
Switch#config
Switch(config)#ip arp-security updateprotect
```

4.10 ARP Guard

4.10.1 Introduction to ARP Guard

The ARP protocol design has a serious security vulnerability: any network device can send ARP messages advertising a mapping between an IP address and a MAC address. This creates an opportunity for ARP spoofing. Attackers can send ARP REQUEST or ARP REPLY messages advertising a false IP-to-MAC mapping, which disrupts network communication. ARP spoofing causes two types of damage: 1. PC4 sends an ARP message advertising that the IP address of PC2 maps to the MAC address of PC4, so all IP packets destined for PC2 are sent to PC4, which can then monitor and capture them; 2. PC4 sends ARP messages advertising that the IP address of PC2 maps to an illegal MAC address, which prevents PC2 from receiving packets destined for it. In particular, if the attacker impersonates the gateway in an ARP spoofing attack, the whole network can collapse.

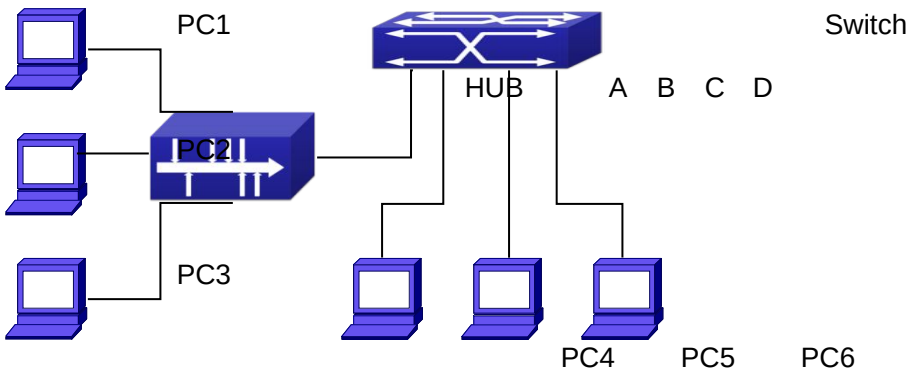


Figure 4-8 ARP Guard schematic diagram

The switch uses its filtering entries to protect the ARP entries of important network devices from being impersonated by other devices. The principle is that the filtering entries check all ARP messages entering through the port; if the source address of an ARP message is protected, the message is dropped and not forwarded.

ARP Guard is usually used to protect the gateway from attacks. Protecting every PC in the network from ARP spoofing would require configuring a large number of ARP Guard addresses on the port, which would use a large part of the FFP entries in the chip and might affect other applications, so it is not recommended. Instead, the FREE RESOURCE access scheme is recommended; refer to the related documents for details.

4.10.2 ARP Guard Configuration Task List

1. Configure protected IP addresses

Command	Explanation
Port Mode	
arp-guard ip <addr> **no arp-guard ip <addr>**	Configures an ARP Guard address. The no command deletes the ARP Guard address.

4.11 ARP Local Proxy

4.11.1 Introduction to ARP Local Proxy

In real deployments, switches at the aggregation layer are required to provide local ARP proxy to prevent ARP spoofing. This function restricts the forwarding of ARP messages within the same VLAN and thus directs the data flow to be forwarded at Layer 3 through the switch.

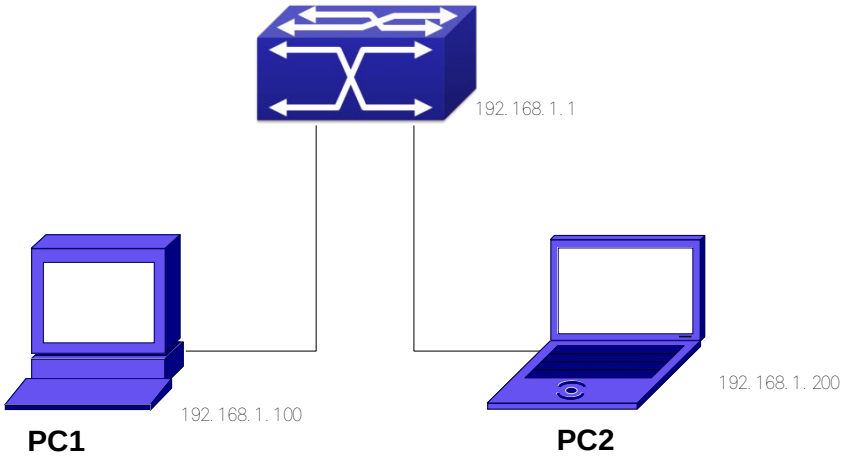


Figure 4-9

As shown in the figure above, when PC1 sends an IP packet to PC2, the overall procedure is as follows (some non-ARP details are omitted):

1. Because PC1 does not have an ARP entry for PC2, it broadcasts an ARP request.

2. When the switch hardware receives the ARP message, it sends the ARP request to the CPU instead of forwarding it in hardware, according to the new ARP handling rules.

3. With local ARP proxy enabled, the switch sends an ARP reply to PC1 (containing the switch's MAC address).

4. After receiving the ARP reply, PC1 creates an ARP entry and sends the IP packet with the destination MAC address in the Ethernet header set to the switch's MAC address.

5. After receiving the IP packet, the switch looks up the routing table (to create a route cache) and installs hardware entries.

6. If the switch has an ARP entry for PC2, it encapsulates the Ethernet header directly and sends the packet (with PC2's MAC address as the destination MAC address).

7. If the switch does not have an ARP entry for PC2, it sends an ARP request for PC2 and then sends the IP packet.

This function should be used together with other security functions. When local ARP proxy is configured on an aggregation switch and port isolation is configured on the Layer 2 switch connected to it, all IP traffic is forwarded at Layer 3 through the aggregation switch. Because of port isolation, ARP messages are not forwarded within the VLAN, so other PCs do not receive them.

4.11.2 ARP Local Proxy Configuration Task List

1. Enable or disable ARP local proxy

1. Enable or disable ARP local proxy

Command	Explanation
VLAN Interface Mode	
ip local proxy-arp **no ip local proxy-arp**	Enables ARP local proxy. The no command disables ARP local proxy.

4.11.3 ARP Local Proxy Typical Example

As shown in the following figure, S1 is a mid-range or high-end Layer 3 switch that supports ARP local proxy, and S2 is a Layer 2 access switch that supports port isolation.

For security, port isolation is enabled on S2, so all downlink ports of S2 are isolated from each other and all ARP messages must be forwarded through S1. If ARP local proxy is enabled on S1, all interfaces on S1 isolate ARP and S1 acts as the ARP proxy. As a result, IP traffic is forwarded at Layer 3 through S1 instead of S2.

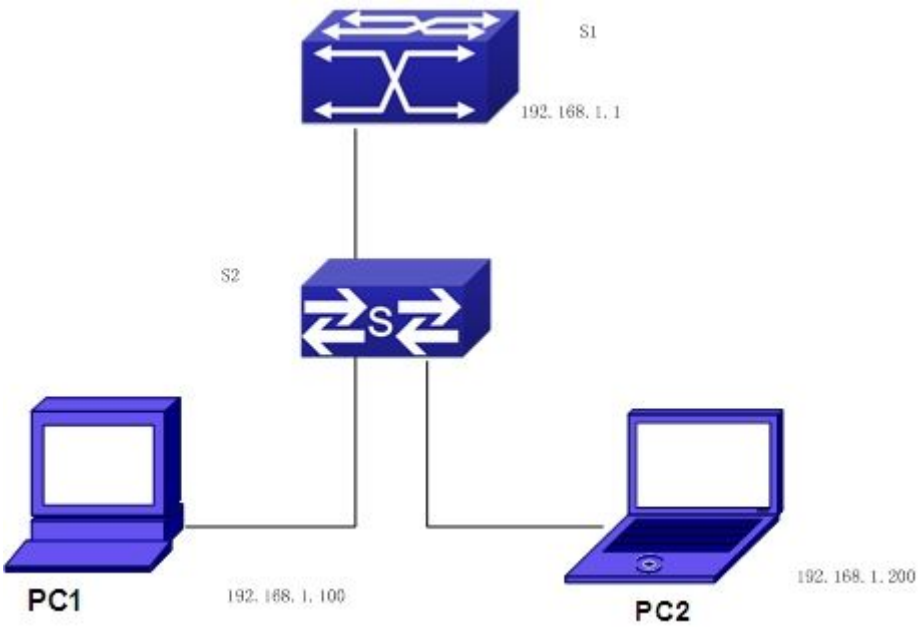


Figure 4-10

Configure S1 as follows:

```
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ip address 192.168.1.1 255.255.255.0
```

```
Switch(Config-if-Vlan1)#ip local proxy-arp
Switch(Config-if-Vlan1)#exit
```

4.11.4 ARP Local Proxy Troubleshooting

ARP local proxy is disabled by default. Use the show commands to view the current configuration. If the configuration is correct, enable ARP debugging to check whether the ARP proxy works normally and sends proxy ARP messages.

If an operational error occurs, the system displays a corresponding prompt.

4.12 Gratuitous ARP

4.12.1 Introduction to Gratuitous ARP

Gratuitous ARP is an ARP request that a host sends with its own IP address as the destination of the request.

The switch works as follows: Layer 3 interfaces of the switch can be configured to send gratuitous ARP packets periodically, or the switch can be configured to send gratuitous ARP packets on all interfaces globally.

Gratuitous ARP serves the following purposes:

1. It reduces how often hosts send ARP requests to the switch. Hosts on the network periodically send ARP requests to the gateway to update the gateway's MAC address. If the switch sends gratuitous ARP requests, the hosts do not need to send these requests, which reduces how often hosts send ARP requests for the gateway's MAC address.
2. It helps prevent ARP spoofing. Gratuitous ARP requests from the switch force hosts to update their ARP cache, so a forged ARP entry for the gateway cannot take effect.

4.12.2 Gratuitous ARP Configuration Task List

1．Enable gratuitous ARP and configure the interval for sending gratuitous ARP requests

2．Display the gratuitous ARP configuration

1. Enable gratuitous ARP and configure the interval for sending gratuitous ARP requests

Command	Explanation
Global Mode and Interface Mode	
ip gratuitous-arp <*5-1200*> **no ip gratuitous-arp**	Enables gratuitous ARP and configures the interval for sending gratuitous ARP requests. The no command disables gratuitous ARP.

2. Display the gratuitous ARP configuration

Command	Explanation
Admin Mode and Configuration Mode	
show ip gratuitous-arp [interface vlan <*1-4094*>]	Displays the gratuitous ARP configuration.

4.12.3 Gratuitous ARP Configuration Example

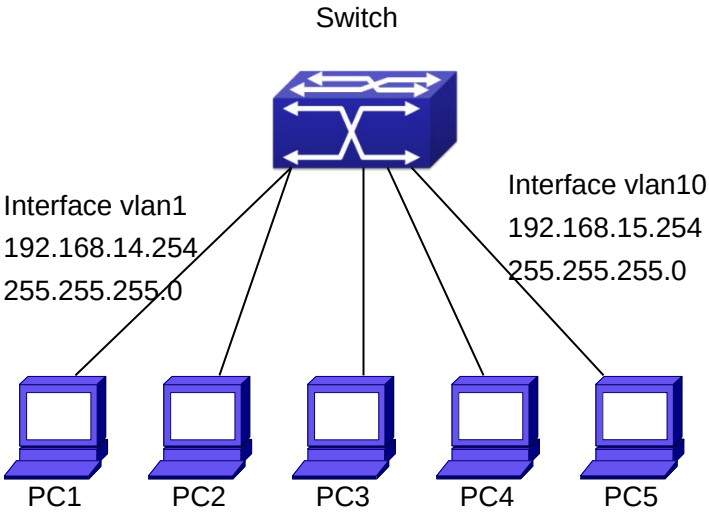


Figure 4-11 Gratuitous ARP configuration example

In the network topology shown above, interface VLAN10 on the switch has IP address 192.168.15.254 and network mask 255.255.255.0, and three PCs (PC3, PC4, and PC5) are connected to it. Interface VLAN 1 has IP address 192.168.14.254 and network mask 255.255.255.0, and two PCs (PC1 and PC2) are connected to it. Gratuitous ARP can be enabled with the following configuration:

1. Enable gratuitous ARP on both interfaces at once:

```
Switch(config)#ip gratuitous-arp 300
Switch(config)#exit
```

2. Enable gratuitous ARP on only one interface:

```
Switch(config)#interface vlan 10
Switch(Config-if-Vlan10)#ip gratuitous-arp 300
Switch(Config-if-Vlan10)#exit
Switch(config) #exit
```

4.12.4 Gratuitous ARP Troubleshooting

Gratuitous ARP is disabled by default. When gratuitous ARP is enabled, debugging information about ARP packets can be displayed with the debug ARP send command.

If gratuitous ARP is enabled in Global Mode, it can be disabled only in Global Mode. If gratuitous ARP is configured in Interface Mode, it can be disabled only in Interface Mode.

4.13 Keepalive Gateway

4.13.1 Introduction to Keepalive Gateway

When an Ethernet port is used for backup or load balancing, it is a broadcast channel and may not detect a change in the physical signal, so it may fail to go down when the gateway goes down. Keepalive gateway detects connectivity to the upstream gateway when an Ethernet port connects to the upstream gateway in a point-to-point topology.

For example, a router connects to an optical terminal device and the line is always up. If the line between the modem and the remote gateway goes down, an effective method is needed to detect whether the remote gateway is reachable. The switch detects gateway connectivity by sending ARP requests to the gateway periodically: if ARP resolution fails, the interface is shut down; if ARP resolution succeeds, the interface stays up.

Only Layer 3 switches support keepalive gateway.

4.13.2 Keepalive Gateway Configuration Task List

1. Enable or disable keepalive gateway, and configure the ARP request interval and the retry count after detection fails
2. Display the keepalive gateway and IPv4 running status of interfaces

1. Enable or disable keepalive gateway, and configure the ARP request interval and the retry count after detection fails

Command	Explanation
Interface Mode	
keepalive gateway <*ip-address*> [{<*interval-seconds*> | msec <*interval-millisecond* >} [retry-count]] **no keepalive gateway**	Enables keepalive gateway and configures the gateway IP address, the interval for sending ARP requests, and the retry count after detection fails. The no command disables keepalive gateway.

2. Display the keepalive gateway and IPv4 running status of interfaces

Command	Explanation
Admin Mode and Configuration Mode	
show keepalive gateway [interface-name]	Displays the keepalive running status of the specified interface; if no interface is specified, displays the keepalive running status of all interfaces.
show ip interface [interface-name]	Displays the IPv4 running status of the specified interface; if no interface is specified, displays the IPv4 running status of all interfaces.

4.13.3 Keepalive Gateway Example

Figure 4-12 Keepalive gateway typical example

In the network topology above, the address of interface vlan10 on gateway A is 1.1.1.1 255.255.255.0, and the address of interface vlan100 on gateway B is 1.1.1.2 255.255.255.0. Gateway B supports keepalive gateway and is configured as follows:

1. Use the default ARP request interval and retry count (the default interval is 10s and the default retry count is 5).

```
Switch(config)#interface vlan 100
Switch(config-if-vlan100)#keepalive gateway 1.1.1.1
Switch(config-if-vlan100)#exit
```

2. Configure the ARP request interval and retry count manually.

```
Switch(config)#interface vlan 100
Switch(config-if-vlan100)#keepalive gateway 1.1.1.1 3 3
Switch(config-if-vlan100)#exit
```

An ARP detection is sent every 3 seconds to check whether gateway A is reachable; after 3 failed detections, gateway A is considered unreachable.

4.13.4 Keepalive Gateway Troubleshooting

If a problem occurs when using keepalive gateway, check whether it is caused by any of the following:

- Make sure the device is a Layer 3 switch; Layer 2 switches do not support keepalive gateway.
- This detection method applies only to point-to-point topologies.
- This method detects IPv4 reachability, so the detection result affects only IPv4 traffic; other traffic, such as IPv6, is not affected.
- The physical state of the interface is controlled only by the physical signal.
- After the gateway is determined to be unreachable, the interface cannot run IPv4: all related IPv4 routes are deleted, and IPv4 routing protocols cannot establish neighbors on the interface.

4.14 Dynamic ARP Inspection

4.14.1 Introduction to Dynamic ARP Inspection

DAI (Dynamic ARP Inspection) is a security feature that verifies ARP packets in the network. With DAI, the administrator can intercept, log, and drop ARP packets with invalid MAC or IP addresses.

Dynamic ARP inspection determines whether ARP packets are legitimate based on the valid IP and MAC addresses in a trusted database. This database can be built through manual static configuration or through dynamic learning by DHCP snooping. ARP packets received on a trusted port are forwarded directly without inspection. For ARP packets received on an untrusted port, the switch forwards only legitimate packets; illegitimate packets are dropped and the action is logged.

Note: The trusted/untrusted ports above are not those of DHCP snooping; they are the ports configured for dynamic ARP inspection.

4.14.2 Dynamic ARP Inspection Configuration Task List

1. **Enable dynamic ARP inspection on a VLAN**

Command	Explanation
Global Mode	
ip arp inspection vlan <vlan-id> **no ip arp inspection vlan <vlan-id>**	Enables dynamic ARP inspection on the VLAN. The no command disables it.

2. **Configure trusted ports**

Command	Explanation
Port Mode	
ip arp inspection trust **no ip arp inspection trust**	Configures the port as a trusted port for dynamic ARP inspection. The no command configures the port as an untrusted port.

3. **Configure the ARP packet rate limit for untrusted ports**

Command	Explanation
Port Mode	
ip arp inspection limit-rate <rate> **no ip arp inspection limit-rate <rate>**	Limits the ARP packet rate on an untrusted port. The no command removes the CPU rate limit.

4.14.3 Dynamic ARP Inspection Configuration Example

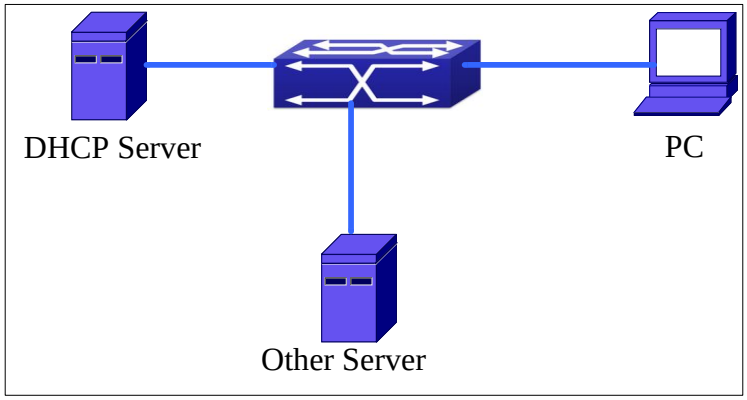


Figure 4-13

Environment: The DHCP server and the PC client are both in VLAN 10.

The DHCP server has MAC address 00-24-8c-01-05-90 and static IP address 192.168.10.2, and is connected to e 1/0/1. The other server has MAC address 00-24-8c-01-05-80 and static IP address 192.168.10.3, and is connected to e 1/0/2. The PC client has MAC address 00-24-8c-01-05-96, obtains its IP address dynamically through DHCP, and is connected to e 1/0/3. The Layer 3 interface of VLAN 10 has IP address 192.168.10.1 and MAC address 00-03-0F-01-02-03.

The configuration is as follows:

```
ip arp inspection vlan 10
ip dhcp snooping enable
ip dhcp snooping vlan 10
!
Interface Ethernet1/0/1
 description connect DHCP Server
 switchport access vlan 10
 ip dhcp snooping trust
 ip arp inspection limit-rate 50
ip arp inspection trust
!
Interface Ethernet1/0/2
 description connect to Other Server
 switchport access vlan 10
 ip arp inspection limit-rate 50
!
Interface Ethernet1/0/3
 description connect to PC
 switchport access vlan 10
```

```
 ip arp inspection limit-rate 50
!
interface Vlan10
 ip address 192.168.10.1 255.255.255.0
```

Explanation: This example uses DAI with both static and dynamic addresses. ARP packets from untrusted ports are all checked against the DHCP snooping binding table to determine whether they are legitimate.

After the client obtains its IP address dynamically, the address can be configured statically on the client, but it must be the same as the dynamically obtained address. If it is changed to a different IP address, the client cannot access the network and the switch reports an illegal ARP warning.

4.15 DHCP

4.15.1 Introduction to DHCP

DHCP (Dynamic Host Configuration Protocol, RFC2131) dynamically assigns IP addresses from an address pool, together with other network configuration parameters such as the default gateway, DNS server, default route, and the location of the host image file. DHCP is an enhanced version of BOOTP. It is a mainstream technology that not only provides boot information for diskless workstations but also frees administrators from manually recording IP address allocations, reducing configuration effort and cost. DHCP also partially relieves IP address shortages: when the user of an IP address leaves the network, that address can be assigned to another user.

DHCP is a client-server protocol. The DHCP client requests a network address and configuration parameters from the DHCP server, and the server provides them to the client. If the DHCP server and clients are in different subnets, a DHCP relay is required to transfer DHCP packets between them. The DHCP process is shown below:

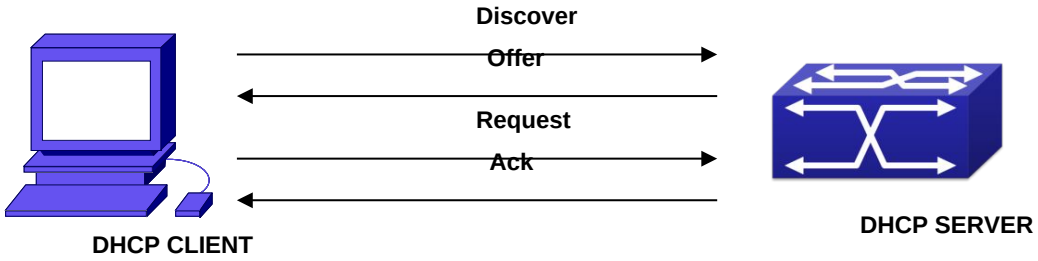


Figure 4-14 DHCP Protocol Interaction

Explanation:

1. The DHCP client broadcasts a DHCPDISCOVER packet in the local subnet.
2. On receiving the DHCPDISCOVER packet, the DHCP server sends the client a DHCPOFFER packet that contains an IP address and other network parameters.
3. After selecting a DHCP server from the received DHCPOFFER packets, the DHCP client broadcasts a DHCPREQUEST packet that identifies the selected server.
4. The selected DHCP server sends a DHCPACK packet, and the client obtains the IP address and other network configuration parameters.

These four steps complete the dynamic host configuration process. However, if the DHCP server and the DHCP client are not in the same network, the server does not receive the broadcast packets sent by the client and therefore sends no DHCP packets to the client. In this case, a DHCP relay is required to forward the DHCP packets so that the exchange between the client and server can be completed.

The switch can act as both a DHCP server and a DHCP relay. The DHCP server supports both dynamic IP address assignment and manual IP address binding (that is, assigning a specific IP address to a specified MAC address or device ID for a long period). The differences and relationships between dynamic allocation and manual binding are as follows: 1) A dynamically obtained IP address can differ each time, whereas a manually bound IP address is always the same. 2) The lease of a dynamically obtained IP address is the same as the lease of the address pool and is limited; the lease of a manually bound IP address is theoretically unlimited. 3) A dynamically allocated address cannot be bound manually. 4) A dynamic DHCP address pool can inherit the network configuration parameters of the dynamic DHCP address pool of the related segment.

4.15.2 DHCP Server Configuration

DHCP Server Configuration Task List:

1. Enable or disable the DHCP service
2. Configure a DHCP address pool
   (1) Create or delete a DHCP address pool
   (2) Configure DHCP address pool parameters
   (3) Configure manual DHCP address pool parameters
3. Enable logging for address conflicts

1. **1. Enable or disable the DHCP service**

Command	Explanation
Global Mode	
service dhcp **no service dhcp**	Enables the DHCP server. The no command disables the DHCP server.

2. Configure a DHCP address pool

(1) Create or delete a DHCP address pool

Command	Explanation
Global Mode	
ip dhcp pool <*name*> **no ip dhcp pool <*name*>**	Creates a DHCP address pool. The no command deletes the DHCP address pool.

(2) Configure DHCP address pool parameters

Command	Explanation
DHCP Address Pool Mode	
network-address <*network-number*> **[mask | prefix-length]** **no network-address**	Configures the range of addresses that the address pool can allocate. The no command deletes the allocated address range of the pool.
default-router **[<*address1*>[<*address2*>[…**<*address8*>]]]** **no default-router**	Configures the default gateway for DHCP clients. The no command deletes the default gateway.
dns-server [<*address1*>[<*address2*>[…<*address8*>]]]** **no dns-server**	Configures the DNS server for DHCP clients. The no command deletes the DNS server configuration.
domain-name <*domain*> **no domain-name**	Configures the domain name for DHCP clients. The no command deletes the domain name.

netbios-name-server [<*address1*>[<*address2*>[… <*address8*>]]] **no netbios-name-server**	Configures the WINS server address. The no command deletes the WINS server address.
netbios-node-type {b-node|h-node|m-node|p-node|<*type-number*>} **no netbios-node-type**	Configures the NetBIOS node type for DHCP clients. The no command deletes the node type for DHCP clients.
bootfile <*filename*> **no bootfile**	Configures the file that DHCP clients import at boot. The no command deletes this setting.
next-server [<*address1*>[<*address2*>[… <*address8*>]]] **no next-server [<*address1*>[<*address2*>[… <*address8*>]]]**	Configures the address of the server that hosts the file to be imported. The no command deletes this server address.
option <*code*> {ascii <*string*> | hex <*hex*> | ipaddress <*ipaddress*>} **no option <*code*>**	Configures the network parameter specified by the option code. The no command deletes the network parameter specified by the option code.
lease { days [hours][minutes] | infinite } **no lease**	Configures the lease for addresses allocated from the address pool. The no command deletes the lease setting of the address pool.
max-lease-time {[<*days*>] [<*hours*>] [<*minutes*>] | infinite} **no max-lease-time**	Sets the maximum lease time for addresses in the address pool. The no command restores the default setting.
Global Mode	
ip dhcp excluded-address <*low-address*> [<*high-address*>] **no ip dhcp excluded-address <*low-address*> [<*high-address*>]**	Excludes addresses in the address pool from dynamic allocation.

(3) Configure manual DHCP address pool parameters

Command	Explanation
DHCP Address Pool Mode	
hardware-address <*hardware-address*> [{Ethernet | IEEE802|<*type-number*>}] **no hardware-address**	Specifies or deletes the hardware address for manual address binding.
host <*address*> [<*mask*> | <*prefix-length*>] **no host**	Specifies or deletes the IP address to be assigned to the specified client for manual address binding.
client-identifier <*unique-identifier*> **no client-identifier**	Specifies or deletes the unique client identifier for manual address binding.

3. Enable logging for address conflicts

Command	Explanation
Global Mode	
ip dhcp conflict logging **no ip dhcp conflict logging**	Enables or disables logging of DHCP address conflicts detected by the switch.
Admin Mode	
clear ip dhcp conflict <*address* | all >	Deletes a single address conflict record or all conflict records.

4.15.3 DHCP Relay Configuration

When the DHCP client and server are in different segments, a DHCP relay is required to transfer DHCP packets. With a DHCP relay, a DHCP server is not needed in each segment: one DHCP server can provide network configuration parameters for clients in multiple segments, which reduces both cost and management effort.

Figure 4-15 DHCP Relay

As shown in the figure above, the DHCP client and the DHCP server are in different networks. The DHCP client performs the usual four DHCP steps, with the DHCP relay added to the process.

1. The client broadcasts a DHCPDISCOVER packet. On receiving it, the DHCP relay inserts its own IP address in the relay agent field of the packet and forwards the packet to the specified DHCP server (for the DHCP frame format, see RFC2131).
2. On receiving the DHCPDISCOVER packet forwarded by the DHCP relay, the DHCP server sends a DHCPOFFER packet to the DHCP client through the DHCP relay.
3. The DHCP client selects a DHCP server and broadcasts a DHCPREQUEST packet. The DHCP relay processes the packet and forwards it to the DHCP server.
4. On receiving the DHCPREQUEST packet, the DHCP server responds with a DHCPACK packet to the DHCP client through the DHCP relay.

DHCP Relay Configuration Task List:

1. Enable DHCP relay.

2. Configure DHCP relay to forward DHCP broadcast packets.

1. Enable DHCP relay.

Command	Explanation
Global Mode	
service dhcp **no service dhcp**	Enables the DHCP service; enabling it enables both the DHCP server and DHCP relay. The no command disables the DHCP service.

2. Configure DHCP relay to forward DHCP broadcast packets.

Command	Explanation
Global Mode	
ip forward-protocol udp bootps **no ip forward-protocol udp bootps**	Enables forwarding of DHCP broadcast packets on UDP port 67. The no command disables this forwarding.
Interface Mode	
ip helper-address <*ipaddress*> **no ip helper-address <*ipaddress*>**	Sets the destination IP address for DHCP relay forwarding. The no command deletes the setting.

4.15.4 DHCP Configuration Examples

Scenario 1:

To reduce configuration effort for network administrators and users, a company uses the switch as a DHCP server. The Admin VLAN IP address is 10.16.1.2/16. The company's local area network is divided into network A and network B by office location. The network configurations for locations A and B are shown below.

PoolA (network 10.16.1.0)		PoolB (network 10.16.2.0)	
Device	IP address	Device	IP address
Default gateway	10.16.1.200 10.16.1.201	Default gateway	10.16.1.200 10.16.1.201

DNS server	10.16.1.202	DNS server	10.16.1.202
WINS server	10.16.1.209	WWW server	10.16.1.209
WINS node type	H-node		
Lease	3 days	Lease	1 day

In location A, the machine with MAC address 00-03-22-23-dc-ab is assigned the fixed IP address 10.16.1.210 and named "management".

```
Switch(config)#service dhcp
Switch(config)#interface vlan 1
Switch(Config-Vlan-1)#ip address 10.16.1.2 255.255.0.0
Switch(Config-Vlan-1)#exit
Switch(config)#ip dhcp pool A
Switch(dhcp-A-config)#network 10.16.1.0 24
Switch(dhcp-A-config)#lease 3
Switch(dhcp-A-config)#default-route 10.16.1.200 10.16.1.201
Switch(dhcp-A-config)#dns-server 10.16.1.202
Switch(dhcp-A-config)#netbios-name-server 10.16.1.209
Switch(dhcp-A-config)#netbios-node-type H-node
Switch(dhcp-A-config)#exit
Switch(config)#ip dhcp excluded-address 10.16.1.200 10.16.1.201
Switch(config)#ip dhcp pool B
Switch(dhcp-B-config)#network 10.16.2.0 24
Switch(dhcp-B-config)#lease 1
Switch(dhcp-B-config)#default-route 10.16.2.200 10.16.2.201
Switch(dhcp-B-config)#dns-server 10.16.2.202
Switch(dhcp-B-config)#option 72 ip 10.16.2.209
Switch(dhcp-config)#exit
Switch(config)#ip dhcp excluded-address 10.16.2.200 10.16.2.201
Switch(config)#ip dhcp pool A1
Switch(dhcp-A1-config)#host 10.16.1.210
Switch(dhcp-A1-config)#hardware-address 00-03-22-23-dc-ab
Switch(dhcp-A1-config)#exit
```

Usage Guidelines: When a DHCP/BOOTP client is connected to a VLAN1 port of the switch, the client can obtain an address only from 10.16.1.0/24, not from 10.16.2.0/24. This is because the client's broadcast packet, after being forwarded by the VLAN interface, requests an IP address in the same segment as the VLAN interface. Because the VLAN interface IP address is 10.16.1.2/24, the IP address assigned to the client belongs to 10.16.1.0/24.

For a DHCP/BOOTP client to obtain an address in 10.16.2.0/24, the gateway that forwards the client's broadcast packets must belong to 10.16.2.0/24, and connectivity between the client gateway and the switch must be ensured.

Scenario 2:

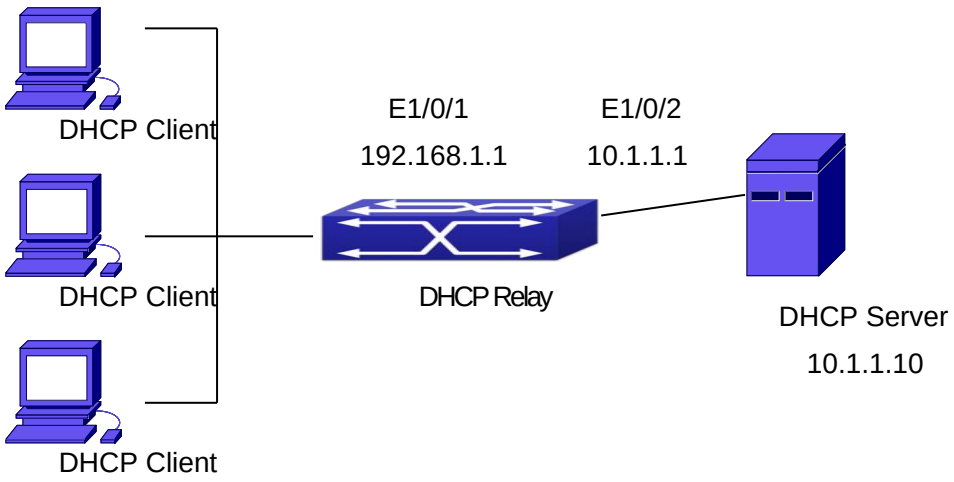


Figure 4-16 DHCP Relay Configuration

As shown in the figure above, the routing switch is configured as a DHCP relay. The DHCP server address is 10.1.1.10. The configuration steps are as follows:

```
Switch(config)#service dhcp
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ip address 192.168.1.1 255.255.255.0
Switch(Config-if-Vlan1)#exit
Switch(config)#vlan 2
Switch(Config-Vlan-2)#exit
Switch(config)#interface Ethernet 1/0/2
Switch(Config-Erthernet1/0/2)#switchport access vlan 2
Switch(Config-Erthernet1/0/2)#exit
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)#ip address 10.1.1.1 255.255.255.0
Switch(Config-if-Vlan2)#exit
Switch(config)#ip forward-protocol udp bootps
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ip help-address 10.1.1.10
Switch(Config-if-Vlan1)#exit
```

Note: Use the **ip forward-protocol udp *<port>*** and **ip helper-address *<ipaddress>*** commands together. **ip help-address** can be configured only on Layer 3 interfaces; it cannot be configured directly on Layer 2 ports.

4.15.5 DHCP Troubleshooting

If DHCP clients cannot obtain IP addresses and other network parameters, and the client hardware and cables have been verified, follow the procedures below.

- ☞ Verify that the DHCP server is running; if it is not, start it. If the DHCP clients and server are not in the same physical network, verify that the router responsible for forwarding DHCP packets supports DHCP relay. If the intermediate router does not support DHCP relay, replace the router or upgrade its software to a version that supports DHCP relay.
    - ☞ In this case, check whether the DHCP server has an address pool in the same segment as the switch VLAN; if not, add one. (This does not mean that the switch cannot assign IP addresses for different segments; see solution 2 for details.)

☞ In the DHCP service, dynamic allocation pools and manual binding pools are mutually exclusive: if both the "**network-address**" and "**host**" commands are configured for a pool, only one of them takes effect. In addition, only one IP-MAC binding can be configured in a manual binding pool. If multiple bindings are required, create multiple manual pools and configure one IP-MAC binding in each. A new configuration in the same pool overwrites the previous one.

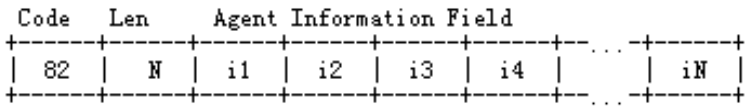
4.16 DHCP Option 82

4.16.1 Introduction to DHCP Option 82

DHCP option 82 is the Relay Agent Information Option (option code 82). It strengthens the security of DHCP servers and improves the IP address allocation policy. The relay agent adds option 82 (including the client's physical access port, the access device ID, and other information) to the DHCP request message from the client and then forwards the message to the DHCP server. A DHCP server that supports option 82 allocates an IP address and other configuration information to the client according to its preconfigured policies and the option 82 information in the message. The DHCP server can also use the option 82 information to identify and defend against possible DHCP attack messages. The DHCP relay agent removes option 82 from the reply messages it receives and forwards each reply message to the network access device port indicated by the physical port information in the option. Option 82 is transparent to the client.

4.16.1.1 DHCP Option 82 Message Structure

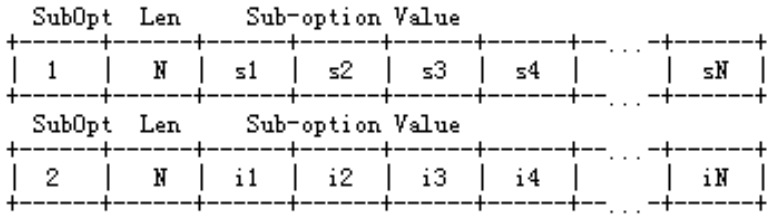
A DHCP message can contain several options, and option 82 is one of them. It must be placed after the other options but before option 255. Its format is as follows:



Code: The code of the Relay Agent Information Option. It is 82, as defined in RFC3046.

Len: The number of bytes in the Agent Information Field, excluding the two bytes of the Code and Len fields.

Option 82 can contain several sub-options and must contain at least one. RFC3046 defines the following two sub-options, whose formats are shown below:



SubOpt: The sub-option code. The Circuit ID sub-option code is 1, and the Remote ID sub-option code is 2.

Len: The number of bytes in the Sub-option Value, excluding the two bytes of the SubOpt and Len fields.

4.16.1.2 Option 82 Working Mechanism

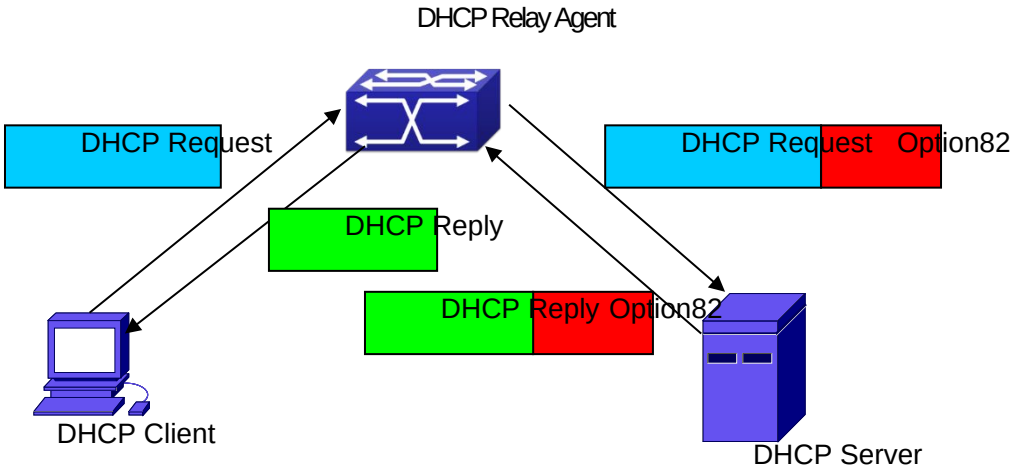


Figure 4-17 DHCP Option 82 Flow Chart

If the DHCP relay agent supports option 82, the DHCP client obtains its IP address from the DHCP server in four steps: discover, offer, select, and acknowledge. The procedure is as follows:

1) While initializing, the DHCP client broadcasts a request message. This request message does not contain option 82.

2) The DHCP relay agent adds option 82 to the end of the received request message and then forwards the message to the DHCP server. By default, sub-option 1 of option 82 (Circuit ID) is the information of the switch interface connected to the DHCP client (VLAN name and physical port name); the Circuit ID can also be user-configured. Sub-option 2 of option 82 (Remote ID) is the MAC address of the DHCP relay device.

3) After receiving the DHCP request message, the DHCP server allocates an IP address and other information to the client according to the option information in the message and its preconfigured policy. It then sends the reply message, which contains the DHCP configuration information and the option 82 information, to the DHCP relay agent.

4) The DHCP relay agent removes the option 82 information from the reply message sent by the DHCP server and then forwards the message with the DHCP configuration information to the DHCP client.

4.16.2 DHCP Option 82 Configuration Task List

1．Enable DHCP option 82 on the relay agent
2．Configure DHCP option 82 attributes of the interface
3．Enable DHCP option 82 on the server
4．Configure the default DHCP option 82 format of the relay agent
5．Configure the delimiter
6．Configure the option 82 creation method
7．Diagnose and maintain DHCP option 82

1. Enable DHCP option 82 on the relay agent.

Command	Explanation
Global Mode	

ip dhcp relay information option **no ip dhcp relay information option**	Enables the option 82 function of the switch relay agent. The no command disables the option 82 function of the switch relay agent.

2. Configure DHCP option 82 attributes of the interface

Command	Explanation
Interface Mode	
ip dhcp relay information policy {drop | keep | replace} **no ip dhcp relay information policy**	Sets the forwarding policy for received DHCP request messages that contain option 82. In drop mode, a message that contains option 82 is dropped without processing. In keep mode, the original option 82 field of the message is kept and the message is forwarded to the server for processing. In replace mode, the option 82 field of the message is replaced with the switch's own option 82 and the message is forwarded to the server for processing. The no command sets the forwarding policy for DHCP messages with option 82 to replace.
ip dhcp relay information option subscriber-id {standard | <*circuit-id*>} **no ip dhcp relay information option subscriber-id**	Sets the format of option 82 sub-option 1 (Circuit ID) added to DHCP request messages received on the interface. standard means the standard VLAN name and physical port name format, such as "Vlan2+Ethernet1/0/12"; <circuit-id> is user-specified Circuit ID content of up to 64 characters. The no command sets the format of the added sub-option 1 (Circuit ID) to standard.
Global Mode	
ip dhcp relay information option remote-id {standard | <*remote-id*>} **no ip dhcp relay information option remote-id**	Sets the content of option 82 sub-option 2 (Remote ID) added to DHCP request packets received on the interface. The no command sets the format of the added sub-option 2 (Remote ID) to standard.

3. Enable DHCP option 82 on the server.

Command	Explanation
Global Mode	
ip dhcp server relay information enable **no ip dhcp server relay information enable**	Enables the switch DHCP server to recognize option 82. The no command makes the server ignore option 82.

4. Configure the default DHCP option 82 format of the relay agent

Command	Explanation
Global Mode	
ip dhcp relay information option subscriber-id format {hex | acsii | vs-hp}	Sets the subscriber-id format of relay agent option 82.
ip dhcp relay information option remote-id format {default | vs-hp}	Sets the remote-id format of relay agent option 82.

5. Configure the delimiter

Command	Explanation
Global Mode	
ip dhcp relay information option delimiter [colon | dot | slash | space] **no ip dhcp relay information option delimiter**	Sets the delimiter between parameters in option 82 sub-options globally. The no command restores the default delimiter (slash).

6. Configure the option 82 creation method

Command	Explanation
Global Mode	
ip dhcp relay information option self-defined remote-id {hostname | mac | string WORD} **no ip dhcp relay information option self-defined remote-id**	Sets the option 82 creation method, allowing the parameters of the remote-id sub-option to be user-defined.
ip dhcp relay information option self-defined remote-id format [ascii | hex]	Sets the self-defined remote-id format for relay option 82.
ip dhcp relay information option self-defined subscriber-id {vlan | port | id (switch-id (mac | hostname)| remote-mac)| string WORD } **no ip dhcp relay information option self-defined subscriber-id**	Sets the option 82 creation method, allowing the parameters of the circuit-id sub-option to be user-defined.
ip dhcp relay information option self-defined subscriber-id format [ascii | hex]	Sets the self-defined circuit-id format for relay option 82.

7. Diagnose and maintain DHCP option 82

Command	Explanation
Admin Mode	
show ip dhcp relay information option	Displays the DHCP option 82 status of the system, including the option 82 enable status, the interface forwarding policy, the Circuit ID mode, and the DHCP server option 82 enable status.
debug ip dhcp relay packet	Displays packet processing information of the DHCP relay agent, including the adding and removing of option 82.

4.16.3 DHCP Option 82 Application Examples

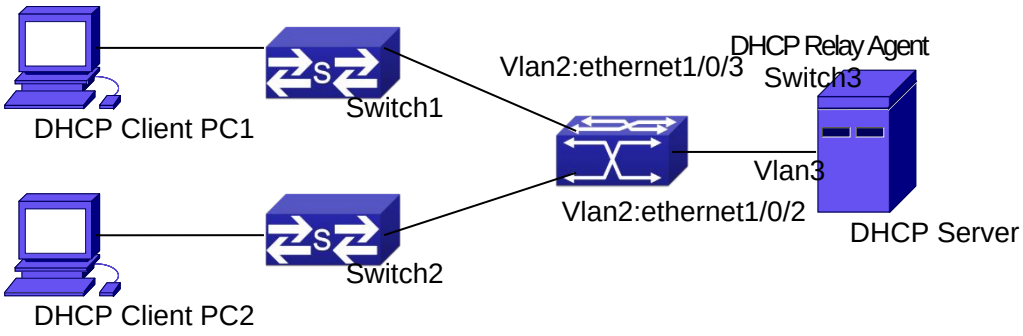


Figure 4-88 Typical DHCP Option 82 Application

In the example above, Layer 2 switches Switch1 and Switch2 are both connected to Layer 3 switch Switch3. As the DHCP relay agent, Switch3 forwards request messages from DHCP clients to the

DHCP server and forwards reply messages from the server to the clients to complete the DHCP procedure. If DHCP option 82 is disabled, the DHCP server cannot distinguish whether a DHCP client is in the network connected to Switch1 or Switch2, so all PCs connected to Switch1 and Switch2 obtain addresses from the public address pool of the DHCP server. With DHCP option 82 enabled, Switch3 appends the information of the Switch3 access port to the client's request message, so the server can tell whether the client is in the network of Switch1 or Switch2 and allocate separate address spaces to the two networks, which simplifies network management.

The configuration of Switch3 (MAC address 00:03:0f:02:33:01) is as follows:

```
Switch3(Config)#service dhcp
Switch3(Config)#ip dhcp relay information option
Switch3(Config)#ip forward-protocol udp bootps
Switch3(Config)#interface vlan 3
Switch3(Config-if-vlan3)#ip address 192.168.10.222 255.255.255.0
Switch3(Config-if-vlan2)#ip address 192.168.102.2 255.255.255.0
Switch3(Config-if-vlan2)#ip helper 192.168.10.88
```

Linux ISC DHCP Server supports option 82. Its configuration file /etc/dhcpd.con is as follows:

```
ddns-update-style interim;
ignore client-updates;
class "Switch3Vlan2Class1" {
match if option agent.circuit-id = "Vlan2+Ethernet1/0/2" and option
agent.remote-id=00:03:0f:02:33:01;
}
class "Switch3Vlan2Class2" {
match if option agent.circuit-id = "Vlan2+Ethernet1/0/3" and option
agent.remote-id=00:03:0f:02:33:01;
}
subnet 192.168.102.0 netmask 255.255.255.0 {
option routers 192.168.102.2;
option subnet-mask 255.255.255.0;
option domain-name "example.com";
option domain-name-servers 192.168.10.3;
authoritative;
pool {
range 192.168.102.21 192.168.102.50;
default-lease-time 86400; #24 Hours
max-lease-time 172800; #48 Hours
allow members of "Switch3Vlan2Class1";
}
pool {
range 192.168.102.51 192.168.102.80;
default-lease-time 43200; #12 Hours
max-lease-time 86400; #24 Hours
allow members of "Switch3Vlan2Class2";
}
}
```

With this configuration, the DHCP server allocates addresses in the range 192.168.102.21 to 192.168.102.50 to network nodes connected to Switch1 that are relayed by Switch3, and addresses in the range 192.168.102.51 to 192.168.102.80 to network nodes connected to Switch2.

4.16.4 DHCP Option 82 Troubleshooting

- ☞ DHCP option 82 is implemented as a sub-function module of the DHCP relay agent. Before using it, make sure that the DHCP relay agent is configured correctly.
- ☞ DHCP option 82 requires the DHCP relay agent and the DHCP server to cooperate in allocating IP addresses. The DHCP server must set the allocation policy correctly according to the network topology of the DHCP relay agent; otherwise, address allocation fails even if

the relay agent operates normally. When more than one kind of relay agent is used, pay attention to the forwarding policy for DHCP request messages on each interface.

- ☞ To troubleshoot the option 82 function of the DHCP relay agent, use the "debug dhcp relay packet" command during operation. It displays information such as the option 82 contents added, the forwarding policy applied, and the server's option 82 contents removed by the relay agent, which helps with troubleshooting.
- ☞ To troubleshoot the option 82 function of the DHCP server, use the "debug ip dhcp server packet" command during operation. It displays the packet processing procedure of the server, including the option 82 information identified in request messages and the option 82 information returned in reply messages.

4.17 DHCP Snooping

4.17.1 Introduction to DHCP Snooping

With DHCP snooping, the switch monitors how DHCP clients obtain IP addresses through DHCP. It prevents DHCP attacks and unauthorized DHCP servers by means of trusted and untrusted ports. DHCP messages from trusted ports are forwarded without verification. Typically, trusted ports connect to DHCP servers or DHCP relay agents, and untrusted ports connect to DHCP clients. The switch forwards DHCP request messages from untrusted ports but not DHCP reply messages. If a DHCP reply message is received on an untrusted port, the switch raises an alarm and also performs the configured action on the port, such as "shutdown" or distributing a "blackhole". If DHCP snooping binding is enabled, the switch saves the binding information (MAC address, IP address, IP lease, VLAN number, and port number) of each DHCP client on untrusted ports in the DHCP snooping binding table. With this information, DHCP snooping can work with modules such as dot1x and ARP, or implement user access control independently.

Defense against fake DHCP servers: When the switch intercepts DHCP server reply packets (DHCPOFFER, DHCPACK, and DHCPNAK), it raises an alarm and responds according to the configuration (shuts down the port or sends a blackhole).

Defense against DHCP overload attacks: To prevent excessive DHCP messages from attacking the CPU, limit the rate at which DHCP packets are received on trusted and untrusted ports.

Recording DHCP binding data: While forwarding DHCP messages, DHCP snooping records the binding data allocated by the DHCP server and can upload the binding data to a specified server for backup. The binding data is mainly used to configure dynamic users on dot1x user-based ports. For details on the dot1x user-based mode, see the "dot1x configuration" chapter.

Adding ARP bindings: After capturing binding data, DHCP snooping can add static ARP bindings based on it to prevent ARP spoofing.

Adding trusted users: After capturing binding data, DHCP snooping can add trusted user list entries based on the parameters in the binding data, so that these users can access all resources without dot1x authentication.

Automatic recovery: Some time after the switch shuts down the port or sends a blackhole, it automatically restores communication of the port or source MAC address and sends information to the log server through syslog.

Log function: When the switch detects abnormal received packets or recovers automatically, it sends syslog information to the log server.

Adding authentication option 82: This function is used with the **dot1x** dhcpoption82 authentication mode. Different option 82 information is added to DHCP messages according to the user's authentication status.

4.17.2 DHCP Snooping Configuration Task Sequence

1. Enable DHCP snooping
2. Enable the DHCP snooping binding function
3. Enable the DHCP snooping binding ARP function
4. Enable the DHCP snooping option 82 function
5. Set the helper server address
6. Set trusted ports
7. Enable the DHCP snooping binding dot1x function
8. Enable the DHCP snooping binding user function
9. Add static binding entries
10. Set defense actions
11. Set the rate limit of DHCP messages
12. Enable debugging
13. Configure DHCP snooping option 82 attributes

1. Enable DHCP snooping

Command	Explanation
Global Mode	
ip dhcp snooping enable **no ip dhcp snooping enable**	Enables or disables the DHCP snooping function.

2. Enable the DHCP snooping binding function

Command	Explanation
Global Mode	
ip dhcp snooping binding enable **no ip dhcp snooping binding enable**	Enables or disables the DHCP snooping binding function.

3. Enable the DHCP snooping binding ARP function

Command	Explanation
Global Mode	
ip dhcp snooping binding arp **no ip dhcp snooping binding arp**	Enables or disables the DHCP snooping binding ARP function.

4. Enable the DHCP snooping option 82 function

Command	Explanation
Global Mode	
ip dhcp snooping information enable **no ip dhcp snooping information enable**	Enables or disables the DHCP snooping option 82 function.

5. Set the helper server address

Command	Explanation
Global Mode	
ip user helper-address A.B.C.D [port <udpport>] source <ipAddr> (secondary|) **no ip user helper-address (secondary|)**	Sets or deletes the helper server address.

6. Set trusted ports

Command	Explanation
Port Mode	
ip dhcp snooping trust **no ip dhcp snooping trust**	Sets or deletes the DHCP snooping trust attribute of ports.

7. Enable the DHCP snooping binding dot1x function

Command	Explanation
Port Mode	
ip dhcp snooping binding dot1x **no ip dhcp snooping binding dot1x**	Enables or disables the DHCP snooping binding dot1x function.

8. Enable the DHCP snooping binding user function

Command	Explanation
Port Mode	
ip dhcp snooping binding user-control **no ip dhcp snooping binding user-control**	Enables or disables the DHCP snooping binding user function.

9. Add static binding entries

Command	Explanation
Global Mode	
ip dhcp snooping binding user <*mac*> address <*ipAddr*> vlan **<*vid*> interface (ethernet|) <*ifname*>** **no ip dhcp snooping binding user <*mac*> interface (ethernet|) <*ifname*>**	Adds or deletes DHCP snooping static binding entries.

10. Set defense actions

Command	Explanation
Port Mode	
ip dhcp snooping action {shutdown|blackhole} [recovery <*second*>] **no ip dhcp snooping action**	Sets or deletes the automatic DHCP snooping defense action of ports.

11. Set the rate limit of DHCP messages

Command	Explanation
Global Mode	
ip dhcp snooping limit-rate <*pps*> **no ip dhcp snooping limit-rate**	Sets the rate limit for transmitting DHCP snooping messages.

12. Enable debugging

Command	Explanation
Admin Mode	
debug ip dhcp snooping packet **debug ip dhcp snooping event** **debug ip dhcp snooping update** **debug ip dhcp snooping binding**	Enables DHCP snooping debugging. For details, see the chapter on system troubleshooting.

13. Configure DHCP snooping option 82 attributes

Command	Explanation
Global Mode	

ip dhcp snooping information option subscriber-id format {hex | acsii | vs-hp}	Sets the subscriber-id format of DHCP snooping option 82.
ip dhcp snooping information option remote-id {standard | <*remote-id*>} **no ip dhcp snooping information option remote-id**	Sets the content of option 82 sub-option 2 (Remote ID) added to DHCP request packets received on the port. The no command sets the format of the added sub-option 2 (Remote ID) to standard.
ip dhcp snooping information option delimiter [colon | dot | slash | space] **no ip dhcp snooping information option delimiter**	Sets the delimiter between parameters in option 82 sub-options globally. The no command restores the default delimiter (slash).
ip dhcp snooping information option self-defined remote-id {hostname | mac | string WORD} **no ip dhcp snooping information option self-defined remote-id**	Sets the option 82 creation method, allowing the parameters of the remote-id sub-option to be user-defined.
ip dhcp snooping information option self-defined remote-id format [ascii | hex]	Sets the self-defined remote-id format for snooping option 82.
ip dhcp snooping information option self-defined subscriber-id {vlan | port | id (switch-id (mac | hostname)| remote-mac) | string WORD} **no ip dhcp snooping information option type self-defined subscriber-id**	Sets the option 82 creation method, allowing the parameters of the circuit-id sub-option to be user-defined.
ip dhcp snooping information option self-defined subscriber-id format [ascii | hex]	Sets the self-defined circuit-id format for snooping option 82.
Port Mode	
ip dhcp snooping information option subscriber-id {standard | <*circuit-id*>} **no ip dhcp snooping information option subscriber-id**	Sets the content of option 82 sub-option 1 (Circuit ID) added to DHCP request packets received on the port. The no command sets the format of the added sub-option 1 (Circuit ID) to standard.
Command	Explanation
Global Mode	
ip dhcp snooping information option allow-untrusted (replace|) **no ip dhcp snooping information option allow-untrusted (replace|)**	Allows untrusted DHCP snooping ports to receive DHCP packets that contain option 82. With "replace", the option 82 in the packet is replaced. The no command makes all untrusted ports drop DHCP packets that contain option 82.

4.17.3 DHCP Snooping Typical Application

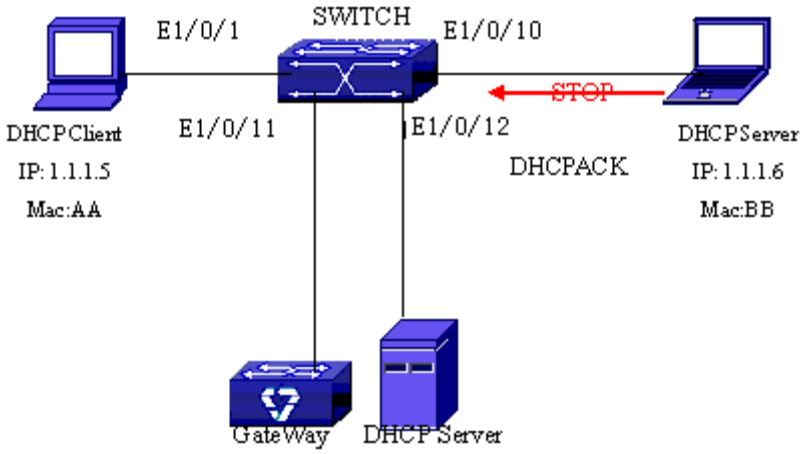


Figure 4-99 Sketch Map of Trunk

As shown in the figure above, Mac-AA is a normal user connected to untrusted port 1/0/1 of the switch. It operates as a DHCP client with IP address 1.1.1.5. The DHCP server and the gateway are connected to trusted ports 1/0/11 and 1/0/12 of the switch. Malicious user Mac-BB, connected to untrusted port 1/0/10, tries to act as a fake DHCP server (by sending DHCPACK). Configuring DHCP snooping on the switch effectively detects and blocks this kind of network attack.

The configuration sequence is as follows:

```
switch#
switch#config
switch(config)#ip dhcp snooping enable
switch(config)#interface ethernet 1/0/11
switch(Config-Ethernet1/0/11)#ip dhcp snooping trust
switch(Config-Ethernet1/0/11)#exit
switch(config)#interface ethernet 1/0/12
switch(Config-Ethernet1/0/12)#ip dhcp snooping trust
switch(Config-Ethernet1/0/12)#exit
switch(config)#interface ethernet 1/0/1-10
switch(Config-Port-Range)#ip dhcp snooping action shutdown
switch(Config-Port-Range)#
```

4.17.4 DHCP Snooping Troubleshooting Help

4.17.4.1 Monitor and Debug Information

Use the "debug ip dhcp snooping" command to monitor debugging information.

4.17.4.2 DHCP Snooping Troubleshooting Help

If a problem occurs when using DHCP snooping, check whether it is caused by the following:

- Check whether DHCP snooping is enabled globally.
- If the port does not react to invalid DHCP server packets, check whether the port is set as an untrusted DHCP snooping port.

4.18 DHCP Option 60 and Option 43

4.18.1 Introduction to DHCP Option 60 and Option 43

The DHCP server analyzes DHCP packets from DHCP clients. If a packet contains option 60, the server decides whether to return option 43 to the client based on the option 60 in the packet and the option 60 and option 43 configured in the DHCP server address pool.

Option 60 and option 43 can be configured in the DHCP server address pool as follows:

1. The address pool is configured with both option 60 and option 43. If a DHCP packet with option 60 received from a DHCP client matches the option 60 of the address pool, the client receives the option 43 configured in the address pool; otherwise, option 43 is not returned to the client.

2. The address pool is configured with option 43 only, which matches any option 60. If a DHCP packet with option 60 is received from a DHCP client, the client receives the option 43 configured in the address pool.

3. The address pool is configured with option 60 only. Option 43 is not returned to the DHCP client.

4.18.2 DHCP Option 60 and Option 43 Configuration Task List

1. Configure basic DHCP option 60 and option 43

Command	Explanation
DHCP Address Pool Mode	
option 60 ascii LINE	Configures the option 60 string in ASCII format in DHCP address pool mode.
option 43 ascii LINE	Configures the option 43 string in ASCII format in DHCP address pool mode.
option 60 hex WORD	Configures the option 60 string in hexadecimal format in DHCP address pool mode.
option 43 hex WORD	Configures the option 43 string in hexadecimal format in DHCP address pool mode.
option 60 ip A.B.C.D	Configures the option 60 string in IP address format in DHCP address pool mode.
option 43 ip A.B.C.D	Configures the option 43 string in IP address format in DHCP address pool mode.
no option 60	Deletes the option 60 configured in the address pool.
no option 43	Deletes the option 43 configured in the address pool.

4.18.3 DHCP Option 60 and Option 43 Example

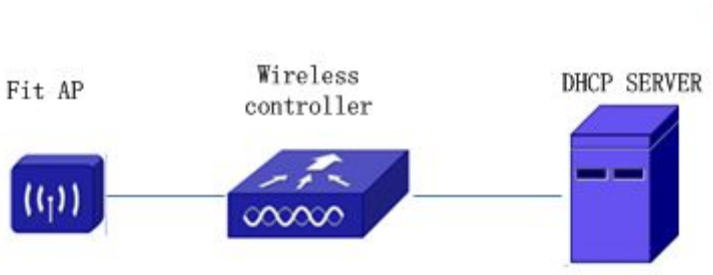


Figure 4-20 Typical DHCP Option 60 and Option 43 Topology

A fit AP obtains its IP address and the option 43 attribute from the DHCP server and then sends a unicast discovery request to the wireless controller. The DHCP server is configured with an option 60 that matches the option 60 of the fit AP, so it returns the option 43 attribute to the fit AP. The wireless controller addresses in DHCP option 43 are 192.168.10.5 and 192.168.10.6.

Configuration procedure:

Configure the DHCP server.

```
switch (config)#ip dhcp pool a
switch (dhcp-a-config)#option 60 ascii AP1000
switch (dhcp-a-config)#option 43 hex 0104C0A80A050104C0A80A06
```

4.18.4 DHCP Option 60 and Option 43 Troubleshooting

If problems occur when configuring DHCP option 60 and option 43, check whether they are caused by the following:

- ☞ Check whether the DHCP service (service dhcp) is enabled.
- ☞ If option 60 is configured in the address pool, check whether it matches the option 60 in the packets.

Chapter 5 Routing Protocol Related Configuration

5.1 Routing Protocol Overview

To communicate with a remote host over the Internet, a host must choose a proper route through a set of routers or Layer 3 switches.

Both routers and Layer 3 switches use the CPU to calculate routes. The difference is that a Layer 3 switch adds the calculated routes to the switching chip, which forwards packets at wire speed, whereas a router stores the calculated routes in its routing table or route cache and forwards data with the CPU. Therefore, although both routers and Layer 3 switches can perform route selection, Layer 3 switches have a great advantage over routers in data forwarding. The following sections describe the basic principles and methods of route selection on a Layer 3 switch.

In route selection, each Layer 3 switch selects a proper intermediate route according to the destination of each received packet and sends the packet to the next Layer 3 switch, until the last Layer 3 switch on the route sends the packet to the destination host. A route is the path that each Layer 3 switch selects to pass the packet to the next Layer 3 switch. Routes are classified into direct routes, static routes, and dynamic routes.

A direct route is a path to a network directly connected to the Layer 3 switch; it is obtained without any calculation.

A static route is a manually specified path to a network or a host; it does not change unless it is reconfigured. Static routes are simple and consistent, prevent unauthorized route modification, and facilitate load balancing and route backup. However, because they are configured manually, static routes are not suitable for medium- or large-scale networks, where the routes are too numerous and complex.

A dynamic route is a path to a network or a host that the Layer 3 switch calculates with the enabled routing protocols. If the next-hop Layer 3 switch on a path becomes unreachable, the Layer 3 switch automatically discards the path through that next hop and chooses a path through other Layer 3 switches.

Dynamic routing protocols fall into two types: Interior Gateway Protocols (IGPs) and Exterior Gateway Protocols (EGPs). An IGP calculates routes to destinations within an autonomous system. The IGPs supported by the switch include RIP and OSPF, which can be configured as required. The switch can run several IGP dynamic routing protocols at the same time. In addition, other dynamic routing protocols and static routes can be redistributed into a dynamic routing protocol so that multiple routing protocols work together.

An EGP, such as BGP, exchanges routing information between different autonomous systems. The EGPs supported by the switch include BGP-4 and BGP-4+.

5.1.1 Routing Table

As mentioned above, a Layer 3 switch mainly establishes routes from itself to networks or hosts and forwards packets according to those routes. Each Layer 3 switch has its own routing table, which contains all the routes the switch uses. Each route entry specifies the physical port to use to forward packets to the destination host, or to the next-hop Layer 3 switch toward that host.

The routing table contains the following main fields:

- ☞ Destination address: identifies the destination address or destination network of an IP packet.
- ☞ Network mask: used together with the destination address to identify the network of the destination host or Layer 3 switch. A network mask consists of a number of consecutive binary 1s and is usually written in dotted decimal notation (an address consisting of one to four 255s). ANDing the destination address with the network mask gives the network address of the

destination host or Layer 3 switch. For example, for destination address 200.1.1.1 with mask 255.255.255.0, the network address is 200.1.1.0.

- ☞ Output interface: specifies the interface through which the Layer 3 switch forwards IP packets.
- ☞ IP address of the next Layer 3 switch (next hop): specifies the next Layer 3 switch that the IP packet passes through.
- ☞ Route entry priority: several routes with different next hops may lead to the same destination. These routes may be learned by different dynamic routing protocols or configured manually as static routes. The entry with the highest priority (smallest value) becomes the current best route. You can configure several routes with different priorities to the same destination; the Layer 3 switch chooses the route used to forward IP packets according to priority.

To prevent the routing table from becoming too large, you can configure a default route. When a routing table lookup fails, the default route is used to forward packets.

The following table lists the routing protocols supported by the switch and their default route lookup priority values.

Routing Protocol or Route Type	Default Priority Value
Direct route	0
OSPF	110
Static route	1
RIP	120
OSPF ASE	150
IBGP	200
EBGP	20
Unknown route	255

5.1.2 IP Routing Policy

5.1.2.1 Introduction to Routing Policy

Routing policies are applied when a router advertises and receives routing information, in order to filter that information; for example, to receive or advertise only routing information that meets specified conditions. A routing protocol may need to redistribute routing information discovered by other protocols, such as OSPF, to expand its own routing knowledge. When redistributing routing information from other routing protocols, the router may need only part of the qualifying routing information, and some attributes may need to be set to suit the protocol.

To implement a routing policy, first define the characteristics of the routing information to which the policy applies, that is, define a set of matching rules. The rules can be based on different attributes of the routing information, such as the destination address and the address of the router that advertises the routing information. Matching rules can be configured in advance and then applied in route advertising, receiving, and redistribution policies.

The switch provides five filters: route-map, ACL, as-path, community-list, and ip-prefix. Each filter is described below.

1. route-map

Matches specified attributes of routing information and sets routing attributes when the conditions are met.

A route-map controls and modifies routing information and controls redistribution between routing protocols. A route-map consists of a series of match and set commands: the match commands specify the conditions to be matched, and the set commands specify the actions to take when a match occurs. A route-map also controls route advertisement between different routing processes. It can also be used for policy-based routing, which selects routes other than the shortest route for packets.

A set of match and set clauses makes up a node. A route-map can consist of several nodes, each of which is a unit of the matching test. Nodes are matched in order of sequence number. Match clauses define the matching rules; the objects matched are attributes of the routing information. Different match clauses in the same node have a logical AND relationship: a node's matching test is passed only when all of its match clauses are matched. Set clauses specify actions, that is, they set attributes of the routing information after the matching test is passed.

Different nodes in a route-map have a logical OR relationship. The system checks the nodes of the route-map in turn; once a node's test is passed, the route-map test is passed and the remaining nodes are not checked.

2. Access control list (ACL)

An ACL (access control list) is a packet filtering mechanism of the switch. The switch controls network access and secures network services by permitting or denying specific packets into or out of the network. You can create a set of rules based on information in packets, with each rule specifying an action (permit or deny) for the matching packets. These rules can be applied to the ingress or egress of the switch, so that traffic in a given direction on a given port must comply with the specified ACL rules. For details, see the chapter "ACL Configuration".

3. IP prefix list (ip-prefix)

An IP prefix list works similarly to an ACL but is more flexible and easier to understand. When used to filter routing information, an ip-prefix list matches the destination address field of the routing information.

An ip-prefix list is identified by its name. Each prefix list can contain multiple entries; each entry specifies a matching range of network prefixes and is identified by a sequence number, which determines the order in which the entries are checked.

During matching, the switch checks the entries in ascending order of sequence number; once an entry is matched, the filter is passed and the remaining entries are not checked.

4. Autonomous system path access list (as-path)

The as-path access list is used only in BGP. BGP routing update packets contain an AS path field, which records the autonomous systems that the routing information has passed through. An as-path list specifies matching conditions for the AS path field.

For as-path configuration, see the ip as-path command in the BGP configuration.

5. community-list

The community-list is used only in BGP. BGP routing update packets contain a community attribute field that identifies a community. A community list specifies matching conditions for the community attribute field.

For community-list configuration, see the ip as-path command in the BGP configuration.

5.1.2.2 IP Routing Policy Configuration Task List

1．Define a route-map
2．Define match clauses in a route-map
3．Define set clauses in a route-map
4．Define an address prefix list

1. Define a route-map

Command	Explanation
Global Mode	

route-map <*map_name*> {deny | permit} <*sequence_num*> **no route-map <*map_name*> [{deny | permit} <*sequence_num*>]**	Configures a route-map. The **no route-map <*map_name*> [{deny | permit} <*sequence_num*>]** command deletes the route-map.

2. Define match clauses in a route-map

Command	Explanation
Route-map Configuration Mode	
match as-path <*list-name*> **no match as-path [<*list-name*>]**	Matches the AS path access list of BGP routes. The **no match as-path [<*list-name*>]** command deletes the match condition.
match community <*community-list-name* | *community-list-num* > [exact-match] **no match community [<*community-list-name* | *community-list-num* > [exact-match]]**	Matches a community attribute access list. The **no match community [<*community-list-name* | *community-list-num* > [exact-match]]** command deletes the match condition.
match interface <*interface-name* > **no match interface [<*interface-name* >]**	Matches by interface. The **no match interface [<*interface-name* >]** command deletes the match condition.
match ip <address | next-hop> <*ip-acl-name* | *ip-acl-num* | prefix-list *list-name*> **no match ip <address | next-hop> [<*ip-acl-name* | *ip-acl-num* | prefix-list [*list-name*]>]**	Matches the address or next hop. The **no match ip <address | next-hop> [<*ip-acl-name* | *ip-acl-num* | prefix-list [*list-name*]>]** command deletes the match condition.
match metric <*metric-val* > **no match metric [<*metric-val* >]**	Matches the route metric value. The **no match metric [<*metric-val* >]** command deletes the match condition.
match origin <egp | igp | incomplete > **no match origin [<egp | igp | incomplete >]**	Matches the route origin. The **no match origin [<egp | igp | incomplete >]** command deletes the match condition.
match route-type external <type-1 | type-2 > **no match route-type external [<type-1 | type-2 >]**	Matches the route type. The **no match route-type external [<type-1 | type-2 >]** command deletes the match condition.
match tag <*tag-val* > **no match tag [<*tag-val* >]**	Matches the route tag. The **no match tag [<*tag-val* >]** command deletes the match condition.

3. Define set clauses in a route-map

Command	Explanation

Route-map Configuration Mode	
set aggregator as <*as-number*> <*ip_addr*> **no set aggregator as [<*as-number*> <*ip_addr*>]**	Sets the AS number of the BGP aggregator. The no command deletes the configuration.
set as-path prepend <*as-num*> **no set as-path prepend [<*as-num*>]**	Prepends the specified AS number to the AS path of BGP routing information. The no command deletes the configuration.
set atomic-aggregate **no set atomic-aggregate**	Sets the BGP atomic aggregate attribute. The no command deletes the configuration.
set comm-list <*community-list-name* | *community-list-num* > delete **no set comm-list <*community-list-name* | *community-list-num* > delete**	Deletes BGP community list values. The no command deletes the configuration.
set community [*AA:NN*] [internet] [local-AS] [no-advertise] [no-export] [none] [additive] **no set community [*AA:NN*] [internet] [local-AS] [no-advertise] [no-export] [none] [additive]**	Sets the BGP community attribute value. The no command deletes the configuration.
set extcommunity <rt | soo> <*AA:NN*> **no set extcommunity <rt | soo> [<*AA:NN*>]**	Sets the BGP extended community attribute. The no command deletes the configuration.
set ip next-hop <*ip_addr*> **no set ip next-hop [<*ip_addr*>]**	Sets the next-hop IP address. The no command deletes the configuration.
set local-preference <*pre_val*> **no set local-preference [<*pre_val*>]**	Sets the local preference. The no command deletes the configuration.
set metric < +/- *metric_val* | *metric_val*> **no set metric [+/- *metric_val* | *metric_val*]**	Sets the route metric value. The no command deletes the configuration.
set metric-type <type-1 | type-2> **no set metric-type [<type-1 | type-2>]**	Sets the OSPF metric type. The no command deletes the configuration.
set origin <egp | igp | incomplete > **no set origin [<egp | igp | incomplete >]**	Sets the BGP route origin. The no command deletes the configuration.
set originator-id <*ip_addr*> **no set originator-id [<*ip_addr*>]**	Sets the route originator ID. The no command deletes the configuration.
set tag <*tag_val*> **no set tag [<*tag_val*>]**	Sets the OSPF route tag value. The no command deletes the configuration.
set vpnv4 next-hop <*ip_addr*> **no set vpnv4 next-hop [<*ip_addr*>]**	Sets the BGP VPNv4 next-hop address. The no command deletes the configuration.
set weight < *weight_val*> **no set weight [<*weight_val*>]**	Sets the BGP route weight. The no command deletes the configuration.

4. Define an address prefix list

Command	Explanation

Global Mode	
ip prefix-list <*list_name*> description <*description*> **no ip prefix-list <*list_name*> description**	Configures the description of a prefix list. The **no ip prefix-list <*list_name*> description** command deletes the description.
ip prefix-list <*list_name*> [seq <*sequence_number*>] <deny | permit> < any | *ip_addr/mask_length* [ge *min_prefix_len*] [le *max_prefix_len*]> **no ip prefix-list <*list_name*> [seq <*sequence_number*>] [<deny | permit> < any | *ip_addr/mask_length* [ge *min_prefix_len*] [le *max_prefix_len*]>]**	Configures a prefix list entry. The **no ip prefix-list <*list_name*> [seq <*sequence_number*>] [<deny | permit> < any | *ip_addr/mask_length* [ge *min_prefix_len*] [le *max_prefix_len*]>]** command deletes the configuration.

5.1.2.3 Configuration Examples

The figure below shows a network of four Layer 3 switches. This example demonstrates how to set the BGP AS path attribute with a route-map. BGP runs between the Layer 3 switches. For SwitchC, network 192.68.11.0/24 can be reached through two paths: AS-PATH 1 through IBGP (via SwitchD) and AS-PATH 2 through EBGP (via SwitchB). BGP selects the shortest path, so AS-PATH 1 is preferred. To make path 2 (the EBGP path) preferred, add two extra AS numbers to the AS-PATH of the routing information that SwitchA sends to SwitchD; this changes the path SwitchC selects to 192.68.11.0/24.

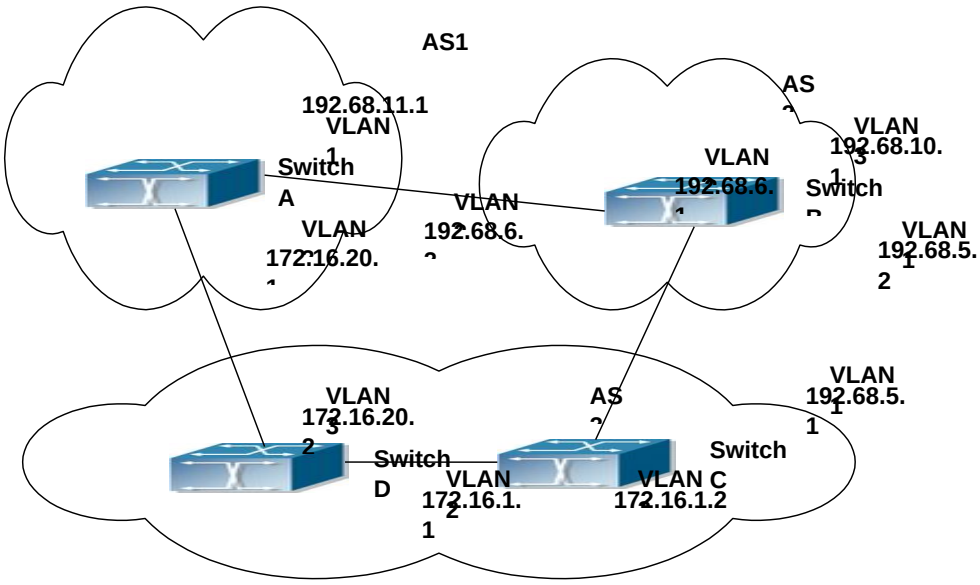


Figure 5-1 Policy Routing Configuration

Configuration procedure (only SwitchA is shown; the configurations of the other switches are omitted):

Configuration of Layer 3 SwitchA:

```
SwitchA #config
SwitchA (config)#router bgp 1
SwitchA (config-router)#neighbor 172.16.20.2 remote-as 3
SwitchA (config-router)#neighbor 192.68.6.1 remote-as 2
SwitchA (config-router)#address-family ipv4 unicast
SwitchA (config-bgp-ipv4)#network 192.68.11.0 mask 255.255.255.0
SwitchA (config-bgp-ipv4)#neighbor 172.16.20.2 route-map AddAsNumbers out
```

```
SwitchA (config-bgp-ipv4)#exit-address-family
SwitchA (config-router)#exit
SwitchA (config)#route-map AddAsNumbers permit 10
SwitchA (config-route-map)#set as-path prepend 1 1
```

5.1.2.4 Troubleshooting

FAQ: The routing protocol cannot learn routing information even though the protocol is running normally.

Troubleshooting: Check for the following errors:

- ☞ At least one node of a route-map must be in permit mode. When a route-map is used to filter routing information, routing information that does not pass the filtering of any node is considered to have failed the filter. If all nodes are set to deny mode, no routing information passes this route-map.
- ☞ At least one entry in an address prefix list must be in permit mode. Deny entries can be defined first to quickly remove unmatched routing information; however, if all entries are in deny mode, no route passes this prefix list. You can define a permit 0.0.0.0/0 le 32 entry after several deny entries to let all other routing information pass. If le 32 is not specified, only the default route is matched.

5.2 Static Route

5.2.1 Introduction to Static Route

As mentioned earlier, a static route is a manually specified path to a network or a host. Static routes are simple and consistent, prevent unauthorized route modification, and facilitate load balancing and route backup. However, they also have drawbacks. As the name indicates, a static route is static: it does not change automatically when a network failure occurs, and manual reconfiguration is required. Therefore, static routes are not suitable for medium- and large-scale networks.

Static routes are mainly used in two situations: 1) in stable networks, to reduce the route selection load and routing traffic; for example, static routes can be used for routes to stub networks; 2) for route backup, by configuring a static route on the backup line with a lower priority than the main line.

Static and dynamic routes can coexist; the Layer 3 switch chooses the route with the highest priority according to the priorities of the routing protocols. Static routes can also be redistributed into dynamic routing protocols, and the priority of the redistributed static routes can be changed as required.

5.2.2 Introduction to Default Route

The default route is a type of static route that is used only when no matching route is found. In the routing table, the default route has destination address 0.0.0.0 and network mask 0.0.0.0. If the routing table contains no route to a packet's destination and no default route is configured, the packet is discarded and an ICMP packet is sent to the source address to indicate that the destination address or network is unreachable.

5.2.3 Static Route Configuration Task List

1. Static route configuration

2. VRF configuration

1. Static route configuration

Command	Explanation
Global Mode	

ip route {*<ip-prefix>* *<mask>* | *<ip-prefix>*/*<prefix-length>*} {*<gateway-address>* | *<gateway-interface>*} [*<distance>*] **no ip route {*<ip-prefix>* *<mask>* | *<ip-prefix>*/*<prefix-length>*} [*<gateway-address>* | *<gateway-interface>*] [*<distance>*]**	Configures a static route. The **no ip route {*<ip-prefix>* *<mask>* | *<ip-prefix>*/*<prefix-length>*} [*<gateway-address>* | *<gateway-interface>*] [*<distance>*]** command deletes a static route entry.

2. VRF configuration

Command	Explanation
Global Mode	
ip route vrf <name> {<ip-prefix> <mask>|<ip-prefix/<prefix-length>} {<gateway-address>|<gateway-interface>} [<distance>] **no ip route vrf <name> {<ip-prefix> <mask>|<ip-prefix/<prefix-length>} [<gateway-address>|<gateway-interface>] [<distance>]**	Configures a static route in a VRF. The no command deletes the static route.

5.2.4 Static Route Configuration Examples

The figure below shows a simple network of three Layer 3 switches; the network mask of all switches and PCs is 255.255.255.0. PC-A and PC-C are connected through the static routes configured on SwitchA and SwitchC; PC-C and PC-B are connected through the static route on SwitchC pointing to SwitchB; PC-B and PC-C are connected through the default route configured on SwitchB.

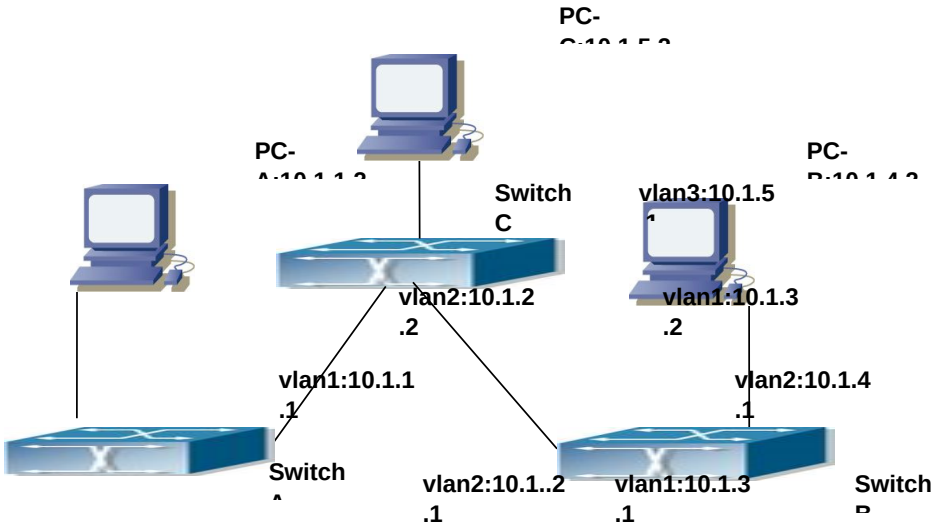


Figure 5-2 Static Route Configuration

Configuration steps:

Configuration of Layer 3 SwitchA:

```
Switch#config
Switch (config) #ip route 10.1.5.0 255.255.255.0 10.1.2.2
```

Configuration of Layer 3 SwitchC:

```
Switch#config
```

The next hop is the peer IP address.

```
Switch(config)#ip route 10.1.1.0 255.255.255.0 10.1.2.1
```

The next hop is the peer IP address.

```
Switch(config)#ip route 10.1.4.0 255.255.255.0 10.1.3.1
```

Configuration of Layer 3 SwitchB:

```
Switch#config
Switch(config)#ip route 0.0.0.0 0.0.0.0 10.1.3.2
```

Ping connectivity is now established between PC-A and PC-C, and between PC-B and PC-C.

5.3 RIP

5.3.1 Introduction to RIP

RIP was first introduced in ARPANET and is designed for small, simple networks. RIP is a distance vector routing protocol based on the Bellman-Ford algorithm. Network devices running a distance vector routing protocol periodically send two kinds of information to neighboring devices:

• The number of hops to the destination network, that is, the metric or the number of networks to traverse.

• The next hop, or the direction (vector) to use to reach the destination network.

Distance vector Layer 3 switches periodically send their entire routing tables to neighboring Layer 3 switches. Each Layer 3 switch builds its own routing table from the information received from its neighbors and then sends this information to its own neighbors. As a result, the routing table is built on secondhand information, and routes beyond 15 hops are considered unreachable.

RIP runs over UDP. Hosts using RIP send and receive packets on UDP port 520. All Layer 3 switches running RIP send their routing tables to all neighboring Layer 3 switches every 30 seconds. If no information is received from a neighbor within 180 seconds, that device is considered to have failed and the networks connected to it are considered unreachable. However, the routes through that Layer 3 switch remain in the routing table for another 120 seconds before they are deleted.

Because Layer 3 switches running RIP build their routing tables from secondhand information, counting to infinity may occur. In a RIP network, when a RIP route becomes unreachable, the neighboring RIP Layer 3 switch does not send a route update immediately; instead, it waits until the update interval expires (every 30 seconds) and then sends update packets containing that route. If, before receiving the update, its neighbors send packets containing information about the failed neighbor, counting to infinity results. In other words, the route to the unreachable Layer 3 switch is still selected, with its metric increasing progressively. This greatly affects route selection and route convergence time.

To prevent counting to infinity, RIP provides mechanisms such as split horizon and triggered updates to resolve routing loops. Split horizon prevents a router from advertising routes back to the gateway from which it learned them. There are two split horizon methods: simple split horizon and split horizon with poison reverse. Simple split horizon removes the routes learned from a neighbor gateway from the routes sent to that gateway; poison reverse not only removes these routes but also sets their cost to infinity. With triggered updates, whenever a gateway changes a route metric, it advertises update packets immediately, regardless of the state of the 30-second update timer.

There are two versions of RIP: version 1 and version 2. RFC 1058 defines RIP-I, and RFC 2453 defines RIP-II, which is compatible with RFC 1723 and RFC 1388. RIP-I sends update packets by broadcast and does not support subnet masks or authentication. Some fields in RIP-I packets are unused and must be all 0s; therefore, these fields should be checked when RIP-I is used, and RIP-I packets in which these fields are non-zero should be discarded. RIP-II is an improved version of RIP-I. RIP-II sends route update packets by multicast (multicast address 224.0.0.9), adds a subnet mask field and a RIP authentication field (simple plaintext password and MD5 password authentication are supported), and supports variable-length subnet masks. RIP-II uses some of the zero fields of RIP-I and does not require zero-field verification. By default, the switch sends RIP-II packets by multicast and accepts both RIP-I and RIP-II packets.

Each Layer 3 switch running RIP has a route database that contains all route entries for reachable destinations, and the routing table is built from this database. When a RIP Layer 3 switch sends route update packets to its neighbors, the packets include the complete routing table. Therefore, in a large network, each Layer 3 switch must transfer and process a large amount of routing data, which degrades network performance.

In addition, RIP allows routing information discovered by other routing protocols to be redistributed into the routing table. RIP can also be used on PE routers as the protocol for exchanging routing information with CEs, and it supports VPN routing/forwarding instances.

RIP operates as follows:

1. After RIP is enabled, the switch broadcasts request packets to neighboring Layer 3 switches; on receiving the request, the neighbors reply with packets containing their local routing information.
2. On receiving the reply packets, the Layer 3 switch updates its local routing table and sends triggered update packets to its neighbors to advertise the route updates. On receiving a triggered update, the neighboring Layer 3 switches send triggered updates to their own neighbors. After a series of triggered update broadcasts, all Layer 3 switches obtain and maintain the latest routing information.

In addition, RIP Layer 3 switches advertise their local routing tables to their neighbors every 30 seconds. On receiving these packets, the neighbors update their local routing tables, select the best routes, and advertise the updated information to their own neighbors, so that the updated routes become valid throughout the network. RIP also uses a timeout mechanism for outdated routes: if a switch does not receive regular update packets from a neighbor within a certain interval (the invalid timer interval), it considers the routes from that neighbor invalid, and after holding the routes for a certain interval (the holddown timer interval), it deletes them.

5.3.2 RIP Configuration Task List

1. Enable RIP (required)

(1) Enable or disable the RIP module

(2) Enable interfaces to send and receive RIP packets

2. Configure RIP parameters (optional)

(1) Configure the RIP packet sending mechanism

1) Configure unicast transmission of RIP packets to specified addresses

2) Configure RIP broadcast on interfaces

(2) Configure RIP route parameters

1) Configure route redistribution (the default route metric and the routes of other protocols to be redistributed into RIP)

2) Configure the interface authentication mode and password

3) Configure route metric offsets

4) Configure and apply route filters

5) Configure split horizon

(3) Configure other RIP parameters

1) Configure the administrative distance of RIP routes

2) Configure the limit on the number of RIP routes in the routing table

3) Configure the RIP update, timeout, holddown, and other timers

4) Configure the RIP UDP receive buffer size

3. Configure RIP-I/RIP-II switching

(1) Configure the RIP version used on all interfaces

(2) Configure the RIP version sent and received on all interfaces

(3) Configure whether interfaces send and receive RIP packets

4. Delete specified routes from the RIP routing table

5. Configure RIP route aggregation

(1) Configure IPv4 aggregate routes in Router Mode

(2) Configure IPv4 aggregate routes in Interface Mode

(3) Display IPv4 aggregate route information

6. Configure redistribution of OSPF routes into RIP

(1) Enable redistribution of OSPF routes into RIP

(2) Display and debug the configuration of redistribution of OSPF routes into RIP

7. Configure the VRF address family mode for RIP

(1) Enable or disable the RIP module

(2) Configure the VRF address family

1. Enable RIP

The basic RIP configuration on the switch is simple. Normally, you only need to enable RIP and configure the network segments that run RIP; RIP packets are then sent and received with the default RIP configuration. If required, you can change the version of the RIP packets sent and received and allow or deny sending and receiving RIP packets. For details, see step 3.

Command	Explanation
Global Mode	
router rip **no router rip**	Enables RIP. The **no router rip** command disables RIP.
Router Mode and Address Family Mode	
network <*A.B.C.D/M* | *ifname*|*vlan*> **no network <*A.B.C.D/M* | *ifname*|*vlan*>**	Enables RIP on the specified network segment. The **no network <*A.B.C.D/M* | *ifname*|*vlan*>** command deletes the segment.

2. Configure RIP parameters

(1) Configure the RIP packet sending mechanism

1) Configure unicast transmission of RIP packets

2) Configure RIP broadcast

Command	Explanation
Router Mode	
neighbor <*A.B.C.D*> **no neighbor <*A.B.C.D*>**	Specifies the IP address of a neighbor router to which RIP packets are sent by unicast. The **no neighbor <*A.B.C.D*>** command removes the specified router.
passive-interface<*ifname*|*vlan*> **no passive-interface<*ifname*|*vlan* >**	Blocks RIP broadcasts on the specified interface, so that RIP packets are exchanged only with Layer 3 switches configured as neighbors. The **no passive-interface<*ifname*| *vlan* >** command cancels this function.

(2) Configure RIP route parameters

1) Configure route redistribution (the default route metric and the routes of other protocols to be redistributed into RIP)

Command	Explanation

Router Mode	
default-metric <*value*> **no default-metric**	Sets the default metric for redistributed routes. The **no default-metric** command restores the default setting.
redistribute {kernel |connected| static| ospf | isis| bgp} [metric<*value*>] [route-map<*word*>] **no redistribute {kernel |connected| static| ospf | isis| bgp} [metric<*value*>] [route-map<*word*>]**	Redistributes routes from other routing protocols into RIP. The **no redistribute {kernel | connected| static| ospf | isis| bgp} [metric<value>] [route-map<word>]** command cancels the redistribution of routes from the corresponding protocol.
default-information originate **no default-information originate**	Generates a default route into RIP. The **no default-information originate** command cancels this function.

2) Configure the interface authentication mode and password

Command	Explanation
Interface Mode	
ip rip authentiaction mode { text| md5| sm3} **no ip rip authentication mode [text| md5|sm3]**	Sets the authentication mode. The **no ip rip authentication mode [text| md5|sm3]** command cancels authentication.
ip rip authentication string <*text*> **no ip rip authentication string**	Sets the authentication key. The **no ip rip authentication string** command removes the key, so that no key is required.
ip rip authentication key-chain <*name-of-chain*> **no ip rip authentication key-chain [<*name-of-chain*>]**	Sets the key chain used for authentication. The **no ip rip authentication key-chain [<*name-of-chain*>]** command stops using the key chain.
ip rip authentication cisco-compatible **no ip rip authentication cisco-compatible**	Enables reception of RIP packets from Cisco devices when MD5 authentication is configured after this command. The no command restores the default configuration.
Global Mode	
key chain <*name-of-chain*> **no key chain < *name-of-chain* >**	Enters Keychain Mode and configures a key chain. The **no key chain < *name-of-chain* >** command deletes the key chain.
Keychain Mode	
key <*keyid*> **no key <*keyid*>**	Enters Keychain-key Mode and configures a key of the key chain. The **no key <*keyid*>** command deletes the key.
Keychain-key Mode	
key-string <*text*> **no key-string <*text*>**	Configures the password used by the key. The **no key-string <*text*>** command deletes the password.
accept-lifetime <*start-time*> {<*end-time*>| duration<*seconds*>| infinite} **no accept-lifetime**	Configures the period during which a key on the key chain is accepted as valid. The **no accept-lifetime** command deletes the setting.
send-lifetime <*start-time*> {<*end-time*>| duration<*seconds*>| infinite} **no send-lifetime**	Configures the period during which a key on the key chain is sent. The **no send-lifetime** command deletes the setting.

3) Configure route metric offsets

Command	Explanation
Router Mode	

offset-list <access-list-number | access-list-name> {in | out } <number> [<ifname>] **no offset-list <access-list-number | access-list-name> {in|out }<number >[<ifname>]**	Adds an offset to the route metric when the interface sends or receives RIP packets. The **no offset-list <access-list-number |access-list-name> {in|out } <number >[<ifname>]** command removes the offset list.

4) Configure and apply route filters

Command	Explanation
Router Mode	
distribute-list {< *access-list-number* | *access-list-name* >|prefix<*prefix-list-name*>}{in|out} [<*ifname*>] **no distribute-list {< *access-list-number* |*access-list-name* >|prefix<*prefix-list-name*>}{in|out} [<*ifname*>]**	Configures and applies an access list or prefix list to filter routes. The **no distribute-list {< *access-list-number* |*access-list-name*>| prefix<*prefix-list-name*>}{in|out} [<*ifname*>]** command stops using the access list or prefix list.

5) Configure split horizon

Command	Explanation
Interface Mode	
ip rip split-horizon [poisoned] **no ip rip split-horizon**	Enables split horizon when the interface sends packets; poisoned enables poison reverse. The **no ip rip split-horizon** command disables split horizon.

(3) Configure other RIP parameters

1) Configure the RIP route priority

2) Configure the limit on the number of RIP routes in the routing table

3) Configure the RIP update, timeout, and holddown timers

4) Configure the RIP UDP receive buffer size

Command	Explanation
Router Mode	
distance <*number*> [<*A.B.C.D/M*>] [<*access-list-name*|*access-list-number* >] **no distance [<*A.B.C.D/M*>]**	Specifies the administrative distance of RIP routes. The **no distance [<*A.B.C.D/M*>]** command restores the default value 120.
maximum-prefix <*maximum-prefix*>[<*threshold*>] **no maximum-prefix <*maximum-prefix* >** **no maximum-prefix**	Configures the maximum number of RIP routes. The **no maximum-prefix <*maximum-prefix* >** or **no maximum-prefix** command cancels the limit.
timers basic <*update*> <*invalid*> <*garbage*> **no timers basic**	Adjusts the update, timeout, and garbage collection timers. The **no timers basic** command restores the default configuration.
recv-buffer-size <*size*> **no recv-buffer-size**	Configures the RIP UDP receive buffer size. The **no recv-buffer-size** command restores the system default value.

3. Configure RIP-I/RIP-II switching

(1) Configure the RIP version used on all ports

Command	Explanation
Router Mode	

version { 1 | 2 } **no version**	Configures the version of all RIP packets sent and received on the Layer 3 switch ports. The **no version** command restores the default configuration (version 2).

(2) Configure the RIP version sent and received on all ports

(3) Configure whether ports send and receive RIP packets

Command	Explanation
Interface Mode	
ip rip send version { 1 | 1-compatible | 2 } **no ip rip send version**	Sets the version of RIP packets sent on the interface. The **no ip rip send version** command restores the version configured by the version command.
ip rip receive version {1 | 2 | } **no ip rip receive version**	Sets the version of RIP packets received on the interface. The no command restores the version configured by the version command.
ip rip receive-packet **no ip rip receive-packet**	Enables receiving RIP packets on the interface. The **no ip rip receive-packet** command disables receiving RIP packets on the interface.
ip rip send-packet **no ip rip send-packet**	Enables sending RIP packets on the interface. The **no ip rip send-packet** command disables sending RIP packets on the interface.

4. Delete specified routes from the RIP routing table

Command	Explanation
Admin Mode	
clear ip rip route {<*A.B.C.D/M*>|kernel| static|connected|rip|ospf|isis|bgp|all}	Deletes the specified routes from the RIP routing table.

5. Configure RIP route aggregation

(1) Configure IPv4 aggregate routes globally

Command	Explanation
Router Mode	
ip rip aggregate-address A.B.C.D/M **no ip rip aggregate-address A.B.C.D/M**	Configures or deletes an IPv4 aggregate route globally.

(2) Configure IPv4 aggregate routes on an interface

Command	Explanation
Interface Mode	
ip rip aggregate-address A.B.C.D/M **no ip rip aggregate-address A.B.C.D/M**	Configures or deletes an IPv4 aggregate route on the interface.

(3) Display IPv4 aggregate route information

Command	Explanation
Admin Mode and Configuration Mode	
show ip rip aggregate	Displays aggregate route information.

6. Configure redistribution of OSPF routes into RIP

(1) Enable redistribution of OSPF routes into RIP

Command	Explanation
Router Mode	
redistribute ospf [<*process-id*>] [metric <*value*>] [route-map <*word*>] **no redistribute ospf [<process-id>]**	Enables or disables the redistribution of OSPF routes into RIP.

(2) Display and debug the configuration of redistribution of OSPF routes into RIP

Command	Explanation
Admin Mode and Configuration Mode	
show ip rip redistribute	Displays the configuration of route redistribution from other routing protocols.
Admin Mode	
debug rip redistribute message send **no debug rip redistribute message send** **debug rip redistribute route receive** **no debug rip redistribute route receive**	Enables or disables debugging of the messages sent by RIP for the redistribution of OSPF routes. Enables or disables debugging of the messages received from NSM.

7. Configure the VRF address family mode for RIP

Command	Explanation
Router Mode	
address-family ipv4 vrf <*vrf-name*> **no address-family ipv4 vrf <*vrf-name*>**	Configures a RIP address family for a VRF on the PE router. The no command deletes the address family.
Address Family Mode	
exit-address-family	Exits Address Family Mode.

5.3.3 RIP Examples

5.3.3.1 Typical RIP Example

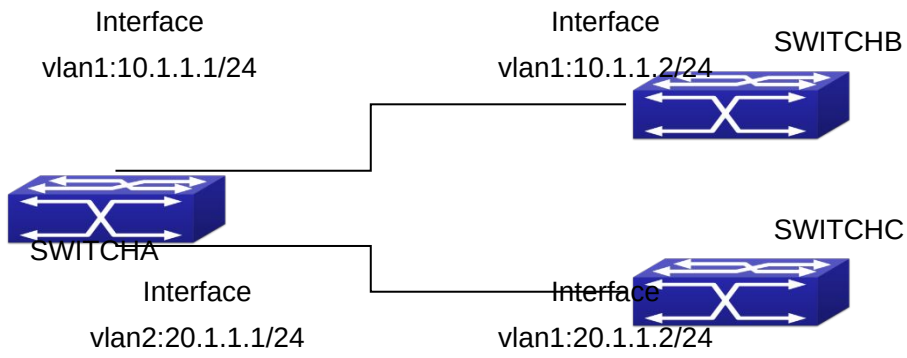


Figure 5-3 RIP Example

In the figure above, the network consists of three Layer 3 switches: SwitchA is connected to SwitchB and SwitchC, and RIP runs on all three switches. SwitchA (interface vlan1: 10.1.1.1, interface vlan2: 20.1.1.1) exchanges route updates only with SwitchB (interface vlan1: 10.1.1.2), not with SwitchC (interface vlan 2: 20.1.1.2).

The configurations of SwitchA, SwitchB, and SwitchC are as follows:

a) Layer 3 SwitchA:

Configure the IP address of interface vlan 1.

```
SwitchA#config
SwitchA(config)# interface vlan 1
SwitchA(Config-if-Vlan1)# ip address 10.1.1.1 255.255.255.0
SwitchA(config-if-Vlan1)#
```

Configure the IP address of interface vlan 2.

```
SwitchA(config)# vlan 2
SwitchA(Config-Vlan2)# switchport interface ethernet 1/0/2
```

```
Set the port Ethernet1/0/2 access vlan 2 successfully
SwitchA(Config-Vlan2)# exit
SwitchA(config)# interface vlan 2
SwitchA(Config-if-Vlan2)# ip address 20.1.1.1 255.255.255.0
```

Enable RIP and configure the RIP network segments.

```
SwitchA(config)#router rip
SwitchA(config-router)#network vlan 1
SwitchA(config-router)#network vlan 2
SwitchA(config-router)#exit
```

Configure interface vlan 2 not to send RIP messages to SwitchC.

```
SwitchA(config)#router rip
SwitchA(config-router)#passive-interface vlan 2
SwitchA(config-router)#exit
SwitchA(config) #
```

b) Layer 3 SwitchB:

Configure the IP address of interface vlan 1.

```
SwitchB#config
SwitchB(config)# interface vlan 1
SwitchB(Config-if-Vlan1)# ip address 10.1.1.2 255.255.255.0
SwitchB(Config-if-Vlan1)exit
```

Enable RIP and configure the RIP network segments.

```
SwitchB(config)#router rip
SwitchB(config-router)#network vlan 1
SwitchB(config-router)#exit
```

c) Layer 3 SwitchC:

```
SwitchC#config
SwitchC(config)# interface vlan 1
```

Configure the IP address of interface vlan 1.

```
SwitchC(Config-if-Vlan1)# ip address 20.1.1.2 255.255.255.0
SwitchC(Config-if-Vlan1)#exit
```

Enable RIP and configure the RIP network segments.

```
SwitchC(config)#router rip
SwitchC(config-router)#network vlan 1
SwitchC(config-router)#exit
```

5.3.3.2 Typical Example of RIP Aggregation

The application topology is as follows:

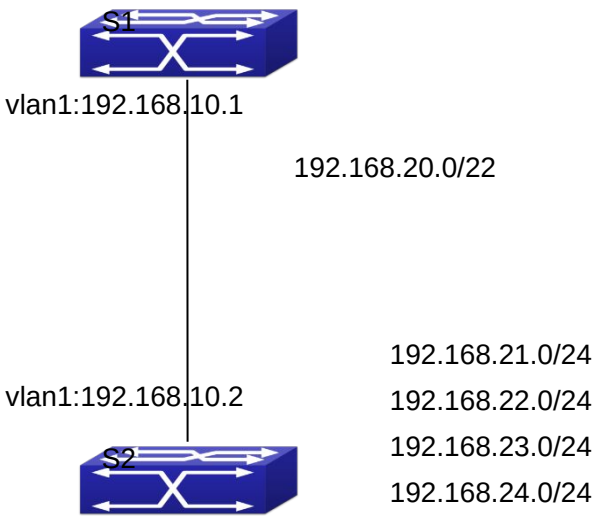


Figure 5-4 Typical Application of RIP Aggregation

In the topology above, S2 is connected to S1 through interface vlan1, and S2 has four other subnet routes: 192.168.21.0/24, 192.168.22.0/24, 192.168.23.0/24, and 192.168.24.0/24. S2 supports route aggregation. After aggregate route 192.168.20.0/22 is configured on interface vlan1 of S2, S2 aggregates the four subnet routes into one route, 192.168.20.0/22, when it sends routing updates to S1 through vlan1, and does not send the subnet routes to the neighbor. This reduces the size of the routing table of S1 and saves memory.

S1 configuration:

```
S1(config)#router rip
S1(config-router) #network vlan 1
```

S2 configuration:

```
S2(config)#router rip
S2(config-router) #network vlan 1
S2(config-router) #exit
S2(config)#in vlan 1
S2(Config-if-Vlan1)# ip rip agg 192.168.20.0/22
```

5.3.4 RIP Troubleshooting

RIP may fail to work properly because of errors such as physical connection problems or configuration errors. Pay attention to the following:

- First, ensure that the physical connections are correct.
- Second, ensure that the interface and link protocol are up (use the **show interface** command).
- Then enable RIP (use the **router rip** command), configure the network segments (use the **network** command), and set the RIP parameters on the corresponding interfaces, such as the choice between RIP-I and RIP-II.
- Note the following RIP behavior: a Layer 3 switch running RIP sends route updates to all neighboring Layer 3 switches every 30 seconds. If no route update is received from a Layer 3 switch within 180 seconds, that switch is considered unreachable, and the route to it remains in the routing table for 120 seconds before it is deleted. Therefore, when a RIP route is to be deleted, the route entry is guaranteed to be removed from the routing table only after 300 seconds.
- When RIP is used on a PE router to exchange routing information with a CE, first create the corresponding VPN routing/forwarding instances and associate them with the corresponding

interfaces. Then enter RIP Address Family Mode and configure the corresponding parameters. If the RIP routing problem persists, use the debug rip command to record debugging messages for three minutes, and contact EnGenius technical support.

5.4 OSPF

5.4.1 Introduction to OSPF

OSPF (Open Shortest Path First) is a link-state interior dynamic routing protocol for autonomous systems. Layer 3 switches exchange link states to create a link-state database and then use the Shortest Path First algorithm to generate a routing table from that database.

An autonomous system (AS) is a self-managed internetwork. In large networks such as the Internet, the giant internetwork is divided into autonomous systems. Large enterprise networks connected to the Internet are independent ASs, because the other hosts on the Internet are not managed by these ASs and do not share interior routing information with their Layer 3 switches.

Each link-state Layer 3 switch can provide the following topology information to its neighboring Layer 3 switches:

• The network segments (links) connected to the Layer 3 switch

• The state of the connected links

Link-state information is flooded throughout the network so that all Layer 3 switches obtain firsthand information. Link-state Layer 3 switches do not broadcast all the information in their routing tables; instead, they send only changed link-state information. Link-state Layer 3 switches establish neighbor relationships by sending Hello packets to their neighbors, and then exchange link-state advertisements (LSAs) with them. Neighboring Layer 3 switches copy the LSAs into their routing tables and pass the information to the rest of the network. This process is called flooding. In this way, firsthand information is sent throughout the network, providing an accurate map for creating and updating routes. Link-state routing protocols use cost instead of hop count to select routes. Cost is assigned automatically or manually. According to the link-state algorithm, cost can be calculated from the number of hops a packet passes, the link bandwidth, and the current load of the link. The administrator can also add weights to better assess link states.

1) When a link-state Layer 3 switch joins a link-state internetwork, it sends Hello packets to discover its neighbors and establish neighbor relationships.

2) The neighbors respond with information about the links they are connected to and the related costs.

3) The originating Layer 3 switch uses this information to build its own routing table.

4) Then, as part of regular updates, the Layer 3 switch sends link-state advertisement (LSA) packets to its neighboring Layer 3 switches. The LSAs include the links of that Layer 3 switch and the related costs.

5) Each neighboring Layer 3 switch copies the LSA packet and passes it to the next neighbor (that is, flooding).

6) Because a Layer 3 switch does not recalculate its routing database before it forwards flooded LSAs, convergence time is greatly reduced.

One major advantage of link-state routing protocols is that counting to infinity cannot occur, because of the way these protocols build their routing tables. The second advantage is that convergence in a link-state internetwork is very fast: when the routing topology changes, updates are flooded throughout the network very quickly. These advantages free up Layer 3 switch resources, because little processing capacity and bandwidth are consumed by bad routing information.

OSPF has the following features: OSPF supports networks of various sizes; an OSPF network can support several hundred Layer 3 switches. Routing topology changes are detected quickly and LSA

updates are sent immediately, so routes converge quickly. Route calculation uses link-state information in the shortest path algorithm, which eliminates routing loops. OSPF divides the autonomous system into areas, which reduces database size, bandwidth usage, and calculation load. (According to their position in the autonomous system, Layer 3 switches are classified as internal area switches, area border switches, AS boundary switches, and backbone switches.) OSPF supports load balancing over multiple equal-cost routes to the same destination. OSPF supports four levels of routing (routes are processed in the order intra-area path, inter-area path, type 1 external path, and type 2 external path). OSPF supports IP subnets, redistribution of routes from other routing protocols, and interface-based packet authentication. OSPF supports sending packets by multicast.

Each OSPF Layer 3 switch maintains a database describing the topology of the whole autonomous system. Each Layer 3 switch collects local status information, such as available interfaces and reachable neighbors, and sends link-state advertisements to exchange link-state information with other OSPF Layer 3 switches, forming a link-state database that describes the whole autonomous system. Each Layer 3 switch builds a shortest path tree rooted at itself from the link-state database; this tree provides the routes to all nodes in the autonomous system. If two or more Layer 3 switches exist on a network (that is, a multi-access network), a designated Layer 3 switch and a backup designated Layer 3 switch are elected. The designated Layer 3 switch is responsible for advertising the link states of the network. This reduces the traffic between Layer 3 switches on a multi-access network.

OSPF requires the autonomous system to be divided into areas, that is, into area 0 (the backbone area) and non-zero areas. Routing information between areas is further abstracted and summarized to reduce the bandwidth required in the network. OSPF uses four types of routes, in order of priority from highest to lowest: intra-area routes, inter-area routes, type 1 external routes, and type 2 external routes. Intra-area and inter-area routes describe the internal network structure of an autonomous system, while external routes describe how to select routes to destinations outside the autonomous system. Type 1 external routes correspond to information that OSPF imports from other interior routing protocols; their costs are comparable to those of OSPF routes. Type 2 external routes correspond to information that OSPF imports from exterior routing protocols; their costs are far greater than those of OSPF routes, so the OSPF route cost is ignored when the route cost is calculated.

OSPF areas are centered on the backbone area, identified as Area 0. All other areas must be logically connected to Area 0, and Area 0 must be contiguous. For this reason, virtual links are introduced for the backbone area, so that physically separated areas still have logical connectivity to the backbone area. All Layer 3 switches in the same area must have the same area configuration.

In summary, LSAs are transferred only between neighboring Layer 3 switches. OSPF includes five types of LSA: router LSA, network LSA, network summary LSA (to other areas), ASBR summary LSA, and AS external LSA, also called type 1, type 2, type 3, type 4, and type 5 LSAs. A router LSA is generated by each Layer 3 switch in an OSPF area and is sent to all other neighboring Layer 3 switches in the same area. A network LSA is generated by the designated Layer 3 switch of a multi-access network in the OSPF area and is sent to all other neighboring Layer 3 switches in the area. (To reduce the traffic between Layer 3 switches on a multi-access network, a designated Layer 3 switch and a backup designated Layer 3 switch are elected on the network, and the designated Layer 3 switch advertises the network link state.) A network summary LSA is generated by the area border switches of an OSPF area and is transferred between area border Layer 3 switches. An AS external LSA is generated by the Layer 3 switches at the external boundary of the AS and is transferred throughout the AS.

For autonomous systems that mainly advertise external link states, OSPF allows some areas to be configured as stub areas to reduce the size of the topology database. Type 4 LSAs (ASBR summary LSAs) and type 5 LSAs (AS external LSAs) are not allowed to flood into or through stub areas. Stub areas must use default routes: the Layer 3 switches at the edge of a stub area advertise default routes into the stub area with type 3 summary LSAs, and these default routes are flooded only inside

the stub area. Each stub area has a corresponding default route, and routes from a stub area to destinations outside the AS depend on the default route of that area.

The OSPF route calculation process is outlined below:

1） Each OSPF-enabled Layer 3 switch maintains a database (the LS database) describing the link states of the topology of the whole autonomous system. Each Layer 3 switch generates a link-state advertisement (router LSA) based on its surrounding network topology and sends the LSA to other Layer 3 switches in link-state update (LSU) packets. Thus, each Layer 3 switch receives LSAs from the other Layer 3 switches, and all the LSAs are combined into the link-state database.

2） Because an LSA describes the network topology around one Layer 3 switch, the LS database describes the topology of the whole network. The Layer 3 switches can easily create a weighted directed graph from the LS database. All Layer 3 switches in the same autonomous system have the same network topology map.

3） Each Layer 3 switch uses the shortest path first (SPF) algorithm to calculate a shortest path tree rooted at itself. The tree provides routes to all nodes in the autonomous system; the leaf nodes contain external routing information. External routes can be tagged by the Layer 3 switch that advertises them, so that additional information about the autonomous system can be recorded. As a result, the routing table of each Layer 3 switch is different.

OSPF was developed by the IETF; the widely used OSPFv2 is implemented as described in RFC 2328.

5.4.2 OSPF Configuration Task List

OSPF configuration on the switch may differ from the configuration procedure on switches from other vendors. It is a two-step process:

1、Enable OSPF in Global Mode; 2. Configure the OSPF area for the interfaces. The configuration task list is as follows:

1. Enable OSPF (required)
   (1) Enable or disable OSPF (required)
   (2) Configure the ID of the Layer 3 switch running OSPF (optional)
   (3) Configure the network range running OSPF (optional)
   (4) Configure the area of the interface (required)
2. Configure OSPF parameters (optional)
   (1) Configure the OSPF packet sending mechanism parameters
      1) Configure OSPF packet authentication
      2) Set the OSPF interface to receive only
      3) Configure the cost of sending packets from the interface
      4) Configure the OSPF packet timers (the Hello interval of broadcast interfaces, the neighbor dead interval, the LSA transmission delay, and the LSA retransmission interval)
   (2) Configure OSPF route redistribution parameters
      1) Configure default parameters (default type, default tag value, and default cost)
      2) Configure the routes of other protocols to be redistributed into OSPF
   (3) Configure OSPF to import routes from other OSPF processes
      1) Enable OSPF to import routes from other OSPF processes
      2) Display related information

3) Debug

(4) Configure other OSPF parameters

1) Configure the OSPF route priority
2) Configure the cost of OSPF stub areas and the default route
3) Configure OSPF virtual links
4) Configure the priority of the interface in designated Layer 3 switch (DR) election
5) Configure whether to log OSPF adjacency changes
6) Filter the routes learned by OSPF

3. Disable OSPF

1. Enable OSPF

The basic OSPF configuration on the switch is simple: usually, you only need to enable OSPF and configure the OSPF area of the interfaces. The OSPF parameters can use their default settings. To modify OSPF parameters, see "2. Configure OSPF parameters".

Command	Explanation
Global Mode	
[no] router ospf [process <*id*>] [VRF Name]	Enables OSPF. The **no** command disables OSPF. (Required)
OSPF Router Mode	
router-id <*router_id*> no **router-id**	Configures the ID of the Layer 3 switch running OSPF. The **no router id** command cancels the ID; the IP address of an interface is then selected as the Layer 3 switch ID. (Optional)
[no] network {**<*network*> <*mask*> | <*network*>/<*prefix*>**} **area <*area_id*>**	Assigns the specified network segment to the specified area. The **no network {<*network*> <*mask*> | <*network*>/<*prefix*>} area <*area_id*> command** cancels this configuration. (Required)

2. Configure OSPF parameters

(1) Configure the OSPF packet sending mechanism parameters

1) Configure OSPF packet authentication

2) Set the OSPF interface to receive only

3) Configure the cost of sending packets from the interface

Command	Explanation
Interface Mode	
ip ospf authentication { message-digest | null} **no ip ospf authentication**	Configures the authentication method the interface uses to accept OSPF packets. The **no ip ospf authentication** command restores the default setting.
ip ospf [<*ip-address*>] authentication-key <*0 LINE* | *7 WORD* | *LINE*> **no ip ospf [<*ip-address*>] authentication**	Specifies the authentication key required to send and receive OSPF packets on the interface. The no command cancels the authentication key.
[no] passive-interface <*ifname*> [<*ip-address*>]	Sets an interface to receive only. The **no passive-interface <*ifname*>[<*ip-address*>]** command cancels this configuration.
ip ospf cost <*cost* > **no ip ospf cost**	Sets the cost of running OSPF on the interface. The **no ip ospf cost** command restores the default setting.

4) Configure the OSPF packet timers (the Hello interval of broadcast interfaces, the neighbor dead interval, the LSA transmission delay, and the LSA retransmission interval)

Command	Explanation
Interface Mode	
ip ospf hello-interval <*time*> **no ip ospf hello-interval**	Sets the interval for sending Hello packets. The **no ip ospf hello-interval** command restores the default setting.
ip ospf dead-interval <*time* > **no ip ospf dead-interval**	Sets the interval after which a neighbor Layer 3 switch is considered invalid. The **no ip ospf dead-interval** command restores the default setting.
ip ospf transit-delay <*time*> **no ip ospf transit-delay**	Sets the delay before link-state advertisements are sent. The **no ip ospf transmit-delay** command restores the default setting.
ip ospf retransmit <*time*> **no ip ospf retransmit**	Sets the interval for retransmitting link-state advertisements to neighbor Layer 3 switches. The **no ip ospf retransmit** command restores the default setting.

(2) Configure OSPF route redistribution parameters

Configure the routes of other protocols to be redistributed into OSPF.

Command	Explanation
OSPF Router Mode	
redistribute { bgp | connected | static | rip | kernel} [metric-type { 1 | 2 }] [tag <*tag*>] [metric <*cost_value*>] [router-map <*WORD*>] **no redistribute { bgp | connected | static | rip | kernel }**	Redistributes routes learned by other protocols and static routes into OSPF as external routing information. The **no redistribute {bgp | connected | static | rip | kernel}** command cancels the redistribution of the external routing information.

(3) Configure OSPF to import routes from other OSPF processes

1) Enable OSPF to import routes from other OSPF processes

Command	Explanation
OSPF Router Mode	
redistribute ospf [<*process-id*>] [metric<*value*>] [metric-type {1|2}] [route-map<*word*>] **no redistribute ospf [<*process-id*>] [metric<*value*>] [metric-type {1|2}] [route-map<*word*>]**	Enables or disables the import of routes from other OSPF processes into OSPF.

2) Display related information

Command	Explanation
Admin Mode or Configuration Mode	
show ip ospf [<*process-id*>] redistribute	Displays the configuration of the OSPF process importing external routes.

3) Debug

Command	Explanation
Admin Mode	

debug ospf redistribute message send **no debug ospf redistribute message send** **debug ospf redistribute route receive** **no debug ospf redistribute route receive**	Enables or disables debugging of the messages sent by an OSPF process for routes redistributed to other OSPF processes. Enables or disables debugging of the routing messages received from NSM by an OSPF process.

(4) Configure other OSPF parameters

1) Configure the OSPF SPF calculation timer

2) Configure the LSA limit of the OSPF link-state database

3) Configure various OSPF parameters

Command	Explanation
OSPF Router Mode	
timers spf <*interval*> **no timers spf**	Configures the OSPF SPF timer. The **no timers spf** command restores the default setting.
overflow database {<*max-LSA*> [hard | soft] | external <*max-LSA*> <*recover time*>} **no overflow database [external <*max-LSA* > < *recover time* >]**	Configures the LSA limit of the database of the current OSPF process. The **no overflow database [external < *max-LSA* > < *recover time* >]** command restores the default setting.
area <id> {authentication [message-digest] | default-cost <*cost*> | filter-list {access | prefix} <*WORD*> {in | out} | nssa [default-information-originate | no-redistribution | no-summary | translator-role] | range <*range*> | stub [no-summary] | virtual-link <*neighbor*>} **no area <id> {authentication | default-cost | filter-list {access | prefix} <*WORD*> {in | out} | nssa [default-information-originate | no-redistribution | no-summary | translator-role] | range <*range*> | stub [no-summary] | virtual-link <*neighbor*>}**	Configures OSPF area parameters (stub areas, NSSAs, and virtual links). The **no area <id> {authentication | default-cost | filter-list {access | prefix} <*WORD*> {in | out} | nssa [default-information-originate | no-redistribution | no-summary | translator-role] | range <*range*> | stub [no-summary] | virtual-link <*neighbor*>}** command restores the default settings.

4) Configure the priority of the interface in designated Layer 3 switch (DR) election

Command	Explanation
Interface Mode	
ip ospf priority <*priority*> **no ip ospf priority**	Sets the priority of the interface in designated Layer 3 switch election. The **no ip ospf priority** command restores the default setting.

5) Configure whether to log OSPF adjacency changes

Command	Explanation
OSPF Router Mode	
log-adjacency-changes detail **no log-adjacency-changes detail**	Enables logging of OSPF adjacency changes. The no command disables the logging.

6) Filter the routes learned by OSPF

Command	Explanation
OSPF Router Mode	
filter-policy <access-list-name> **no filter-policy**	Filters the routes learned by OSPF with an access list. The no command cancels the route filtering.

3. Disable OSPF

Command	Explanation
Global Mode	
no router ospf [process *<id>*]	Disables OSPF.

5.4.3 OSPF Examples

5.4.3.1 OSPF Configuration Example

Scenario 1: OSPF autonomous system

This scenario uses an OSPF autonomous system consisting of five switches as an example.

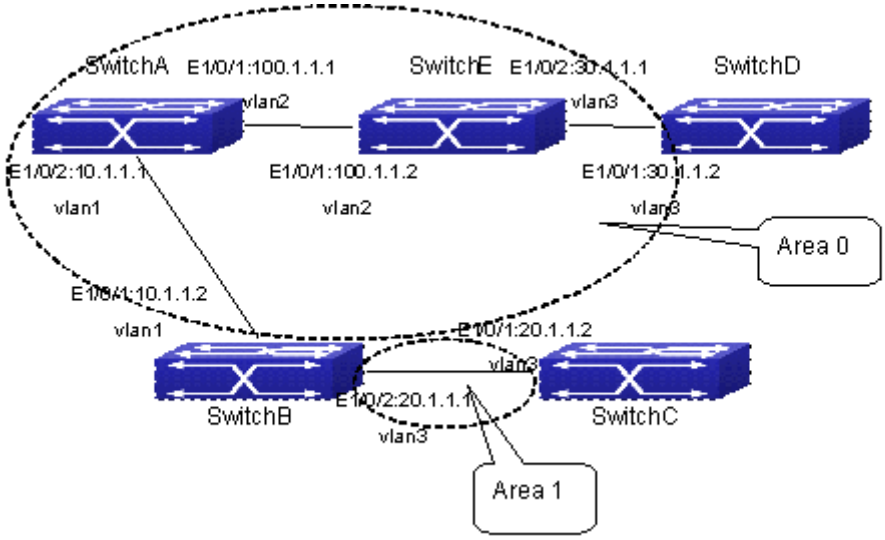


Figure 5-5 Network Topology of the OSPF Autonomous System

The configurations of Layer 3 Switch1 to Switch5 are shown below:

Layer 3 Switch1:

Configure the IP address of interface vlan1.

```
Switch1#config
Switch1(config)# interface vlan 1
Switch1(config-if-vlan1)# ip address 10.1.1.1 255.255.255.0
Switch1(config-if-vlan1)#exit
```

Configure the IP address of interface vlan2.

```
Switch1(config)# interface vlan 2
Switch1(config-if-vlan2)# ip address 100.1.1.1 255.255.255.0
Switch1 (config-if-vlan2)#exit
```

Enable OSPF and configure the area of interfaces vlan1 and vlan2.

```
Switch1(config)#router ospf
Switch1(config-router)#network 10.1.1.0/24 area 0
Switch1(config-router)#network 100.1.1.0/24 area 0
Switch1(config-router)#exit
Switch1(config)#exit
Switch1#
```

Layer 3 Switch2:

Configure the IP addresses of interfaces vlan1 and vlan3.

```
Switch2#config
Switch2(config)# interface vlan 1
Switch2(config-if-vlan1)# ip address 10.1.1.2 255.255.255.0
Switch2(config-if-vlan1)#no shutdown
```

```
Switch2(config-if-vlan1)#exit
Switch2(config)# interface vlan 3
Switch2(config-if-vlan3)# ip address 20.1.1.1 255.255.255.0
Switch2(config-if-vlan3)#no shutdown
Switch2(config-if-vlan3)#exit
```

Enable OSPF and configure the OSPF areas of interfaces vlan1 and vlan3.

```
Switch2(config)#router ospf
Switch2(config-router)# network 10.1.1.0/24 area 0
Switch2(config-router)# network 20.1.1.0/24 area 1
Switch2(config-router)#exit
Switch2(config)#exit
Switch2#
```

Layer 3 Switch3:

Configure the IP address of interface vlan3.

```
Switch3#config
Switch3(config)# interface vlan 3
Switch3(config-if-vlan1)# ip address 20.1.1.2 255.255.255.0
Switch3(config-if-vlan3)#no shutdown
Switch3(config-if-vlan3)#exit
```

Enable OSPF and configure the OSPF area of interface vlan3.

```
Switch3(config)#router ospf
Switch3(config-router)# network 20.1.1.0/24 area 1
Switch3(config-router)#exit
Switch3(config)#exit
Switch3#
```

Layer 3 Switch4:

Configure the IP address of interface vlan3.

```
Switch4#config
Switch4(config)# interface vlan 3
Switch4(config-if-vlan3)# ip address30.1.1.2 255.255.255.0
Switch4(config-if-vlan3)#no shutdown
Switch4(config-if-vlan3)#exit
```

Enable OSPF and configure the OSPF area of interface vlan3.

```
Switch4(config)#router ospf
Switch4(config-router)# network 30.1.1.0/24 area 0
Switch4(config-router)#exit
Switch4(config)#exit
Switch4#
```

Layer 3 Switch5:

Configure the IP address of interface vlan2.

```
Switch5#config
Switch5(config)# interface vlan 2
Switch5(config-if-vlan2)# ip address 100.1.1.2 255.255.255.0
Switch5(config-if-vlan2)#no shutdown
Switch5(config-if-vlan2)#exit
```

Configure the IP address of interface vlan3.

```
Switch5(config)# interface vlan 3
Switch5(config-if-vlan3)# ip address 30.1.1.1 255.255.255.0
Switch5(config-if-vlan3)#no shutdown
Switch5(config-if-vlan3)#exit
```

Enable OSPF and configure the areas of interfaces vlan2 and vlan3.

```
Switch5(config)#router ospf
Switch5(config-router)# network 30.1.1.0/24 area 0
```

```
Switch5(config-router)# network 100.1.1.0/24 area 0
Switch5(config-router)#exit
Switch5(config)#exit
Switch5#
```

Scenario 2: Typical complex OSPF topology

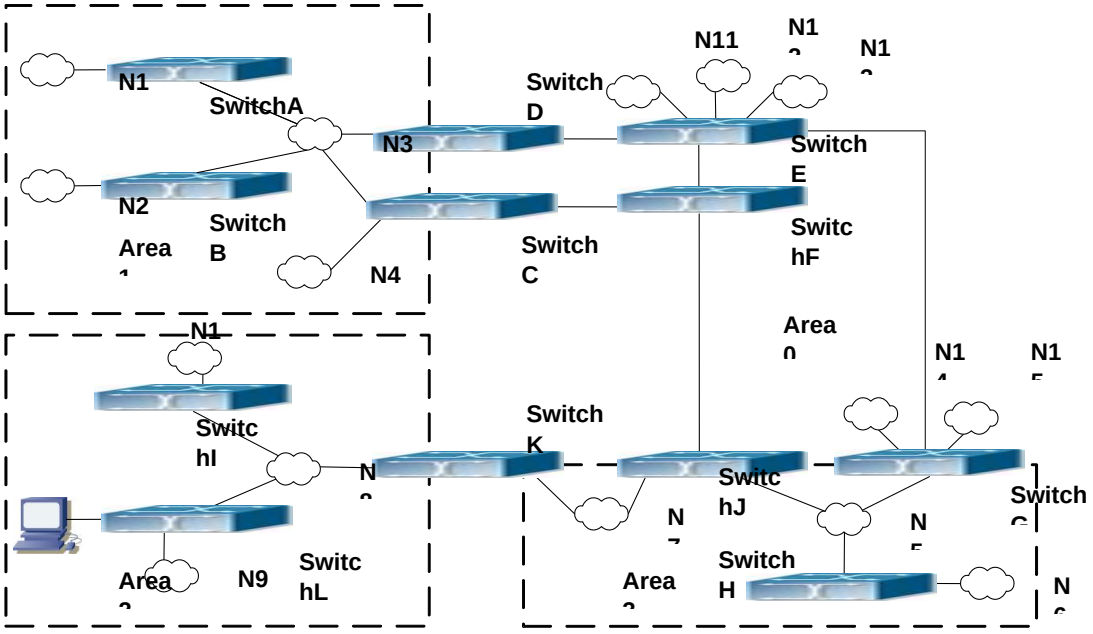


Figure 5-6 Typical Complex OSPF Autonomous System

This scenario is a typical complex OSPF autonomous system topology. Area 1 includes networks N1-N4 and Layer 3 SwitchA-SwitchD; area 2 includes networks N8-N10, host H1, and Layer 3 SwitchH; area 3 includes networks N5-N7 and Layer 3 SwitchF, SwitchG, SwitchA0, and Switch11. Networks N8-N10 and host H1 share a summary route (that is, area 3 is defined as a stub area). Layer 3 SwitchA, SwitchB, SwitchD, SwitchE, SwitchG, SwitchH, and Switch12 are internal area Layer 3 switches; SwitchC, SwitchD, SwitchF, Switch10, and Switch11 are area border Layer 3 switches; and SwitchD and SwitchF are AS boundary Layer 3 switches.

In area 1, Layer 3 SwitchA and SwitchB are internal area switches. Area border switches SwitchC and SwitchD report the distance costs to all destinations outside the area and also report the positions of AS boundary Layer 3 switches SwitchD and SwitchF. AS external LSAs from SwitchD and SwitchF are flooded throughout the autonomous system. When ASE LSAs are flooded in area 1, they are included in the area 1 database to provide routes to networks N11 and N15.

In addition, Layer 3 SwitchC and SwitchD must summarize the topology of area 1 into the backbone area (area 0; all non-zero areas must be connected through area 0 and cannot be connected directly) and advertise the networks in area 1 (N1-N4) and the costs from SwitchC and SwitchD to these networks. Because the backbone area must remain contiguous, a virtual link is required between backbone Layer 3 Switch10 and Switch11. The area border Layer 3 switches exchange summary information through the backbone Layer 3 switches, and each area border Layer 3 switch listens to the summary information from the other border Layer 3 switches.

A virtual link not only maintains the connectivity of the backbone area but also strengthens it. For example, if the connection between backbone Layer 3 SwitchG and Switch10 is broken, the backbone area becomes discontiguous. A virtual link between backbone Layer 3 switches SwitchF and Switch10 makes the backbone area more robust. In addition, the virtual link between SwitchF and Switch10 provides a short path from area 3 to Layer 3 SwitchF.

Take area 1 as an example. Assume that the IP address of Layer 3 SwitchA is 10.1.1.1, the IP address of interface VLAN2 of Layer 3 SwitchB is 10.1.1.2, that of interface VLAN2 of Layer 3

SwitchC is 10.1.1.3, and that of interface VLAN2 of Layer 3 SwitchD is 10.1.1.4. SwitchA connects to network N1 through interface VLAN1 (IP address 20.1.1.1); SwitchB connects to network N2 through interface VLAN1 (IP address 20.1.2.1); SwitchC connects to network N4 through interface VLAN3 (IP address 20.1.3.1). All three addresses belong to area 1. SwitchC connects to Layer 3 SwitchE through interface VLAN1 (IP address 10.1.5.1); SwitchD connects to Layer 3 SwitchD through interface VLAN1 (IP address 10.1.6.1); both addresses belong to area 1. Simple authentication is used between the Layer 3 switches in area 1, and the area border Layer 3 switches of area 1 use MD5 authentication with the area 0 backbone Layer 3 switches.

Only the configurations of the Layer 3 switches in area 1 (SwitchA, SwitchB, SwitchC, and SwitchD) are shown below; the configurations of the Layer 3 switches in other areas are omitted.

1) SwitchA:

Configure the IP address of interface vlan2.

```
SwitchA#config
SwitchA(config)# interface vlan 2
SwitchA(config-If-Vlan2)# ip address 10.1.1.1 255.255.255.0
SwitchA(config-If-Vlan2)#exit
```

Enable OSPF and configure the area of interface vlan2.

```
SwitchA(config)#router ospf
SwitchA(config-router)#network 10.1.1.0/24 area 1
SwitchA(config-router)#exit
```

Configure simple key authentication.

```
SwitchA(config)#interface vlan 2
SwitchA(config-If-Vlan2)#ip ospf authentication
SwitchA(config-If-Vlan2)#ip ospf authentication-key test
SwitchA(config-If-Vlan2)exit
```

Configure the IP address and area of interface vlan1.

```
SwitchA(config)# interface vlan 1
SwitchA(config-If-Vlan1)#ip address 20.1.1.1 255.255.255.0
SwitchA(config-If-Vlan1)#exit
SwitchA(config)#router ospf
SwitchA(config-router)#network 20.1.1.0/24 area 1
SwitchA(config-router)#exit
```

2) SwitchB:

Configure the IP address of interface vlan2.

```
SwitchB#config
SwitchB(config)# interface vlan 2
SwitchB(config-If-Vlan2)# ip address 10.1.1.2 255.255.255.0
SwitchB(config-If-Vlan2)#exit
```

Enable OSPF and configure the area of interface vlan2.

```
SwitchB(config)#router ospf
SwitchB(config-router)#network 10.1.1.0/24 area 1
SwitchB(config-router)#exit
SwitchB(config)#interface vlan 2
```

Configure simple key authentication.

```
SwitchB(config)#interface vlan 2
SwitchB(config-If-Vlan2)#ip ospf authentication
SwitchB(config-If-Vlan2)#ip ospf authentication-key test
SwitchB(config-If-Vlan2)#exit
```

Configure the IP address and area of interface vlan1.

```
SwitchB(config)# interface vlan 1
SwitchB(config-If-Vlan1)#ip address 20.1.2.1 255.255.255.0
SwitchB(config-If-Vlan1)#exit
```

```
SwitchB(config)#router ospf
SwitchB(config-router)#network 20.1.2.0/24 area 1
SwitchB(config-router)#exit
SwitchB(config)#exit
```

3) SwitchC:

Configure the IP address of interface vlan2.

```
SwitchC#config
SwitchC(config)# interface vlan 2
SwitchC(config-If-Vlan2)# ip address 10.1.1.3 255.255.255.0
SwitchC(config-If-Vlan2)#exit
```

Enable OSPF and configure the area of interface vlan2.

```
SwitchC(config)#router ospf
SwitchC(config-router)#network 10.1.1.0/24 area 1
SwitchC(config-router)#exit
```

Configure simple key authentication.

```
SwitchC(config)#interface vlan 2
SwitchC(config-If-Vlan2)#ip ospf authentication
SwitchC(config-If-Vlan2)#ip ospf authentication-key test
SwitchC(config-If-Vlan2)#exit
```

Configure the IP address and area of interface vlan3.

```
SwitchC(config)# interface vlan 3
SwitchC(config-If-Vlan3)#ip address 20.1.3.1 255.255.255.0
SwitchC(config-If-Vlan3)#exit
SwitchC(config)#router ospf
SwitchC(config-router)#network 20.1.3.0/24 area 1
SwitchC(config-router)#exit
```

Configure the IP address and area of interface vlan 1.

```
SwitchC(config)# interface vlan 1
SwitchC(config-If-Vlan1)#ip address 10.1.5.1 255.255.255.0
SwitchC(config-If-Vlan1)#exit
SwitchC(config)#router ospf
SwitchC(config-router)#network 10.1.5.0/24 area 0
SwitchC(config-router)#exit
```

Configure MD5 key authentication.

```
SwitchC(config)#interface vlan 1
SwitchC (config-If-Vlan1)#ip ospf authentication message-digest
SwitchC (config-If-Vlan1)#ip ospf authentication-key test
SwitchC (config-If-Vlan1)#exit
SwitchC(config)#exit
SwitchC#
```

4) SwitchD:

Configure the IP address of interface vlan2.

```
SwitchD#config
SwitchD(config)# interface vlan 2
SwitchD(config-If-Vlan2)# ip address 10.1.1.4 255.255.255.0
SwitchD(config-If-Vlan2)#exit
```

Enable OSPF and configure the area of interface vlan2.

```
SwitchD(config)#router ospf
SwitchD(config-router)#network 10.1.1.0/24 area 1
SwitchD(config-router)#exit
```

Configure simple key authentication.

```
SwitchD(config)#interface vlan 2
SwitchD(config-If-Vlan2)#ip ospf authentication
```

```
SwitchD(config-If-Vlan2)#ip ospf authentication-key test
SwitchD(config-If-Vlan2)#exit
```

Configure the IP address and area of interface vlan 1.

```
SwitchD(config)# interface vlan 1
SwitchD(config-If-Vlan1)# ip address 10.1.6.1 255.255.255.0
SwitchD(config-If-Vlan1)exit
SwitchD(config)#router ospf
SwitchD(config-router)#network 10.1.6.0/24 area 0
SwitchD(config-router)#exit
```

Configure MD5 key authentication.

```
SwitchD(config)#interface vlan 1
SwitchD(config-If-Vlan1)#ip ospf authentication message-digest
SwitchD(config-If-Vlan1)#ip ospf authentication-key test
SwitchD(config-If-Vlan1)exit
SwitchD(config)#exit
SwitchD#
```

Scenario 3: OSPF importing routes from other OSPF processes

As shown in the following figure, a switch running OSPF connects two networks: network A and network B. For some reason, network A must be able to learn the routes of network B, but network B must not learn the routes of network A. To achieve this, start two OSPF processes, one on interface vlan 1 and one on interface vlan 2. Configure the OSPF process of interface vlan 1 to import the routes of the OSPF process of interface vlan 2, but do not configure the OSPF process of interface vlan 2 to import the routes of the OSPF process of interface vlan 1.

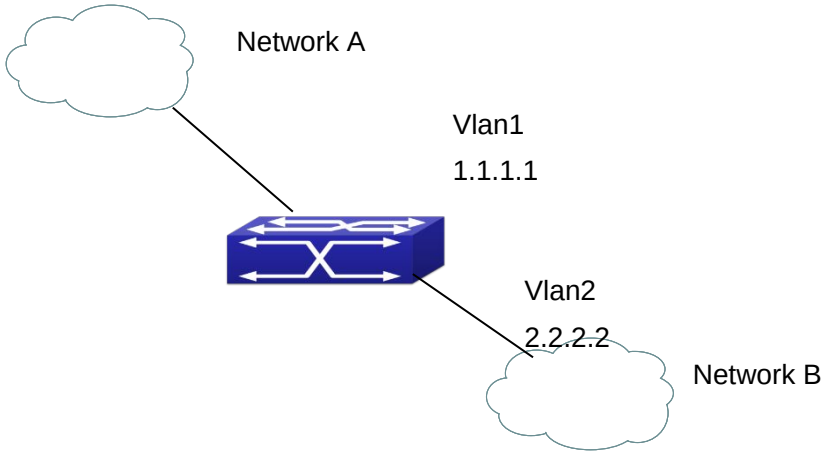


Figure 5-7 Example of OSPF Importing Routes from Other OSPF Processes

Configure the switch as follows:

```
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ip address 1.1.1.1 255.255.255.0
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)#ip address 2.2.2.2 255.255.255.0
Switch(Config-if-Vlan2)#exit
Switch(config)#router ospf 10
Switch(config-router)#network 2.2.2.0/24 area 1
Switch(config-router)#exit
Switch(config)#router ospf 20
Switch(config-router)#network 1.1.1.0/24 area 1
Switch(config-router)#redistribute ospf 10
Switch(config-router)#exit
```

5.4.3.2 Configuration Example of OSPF VPN

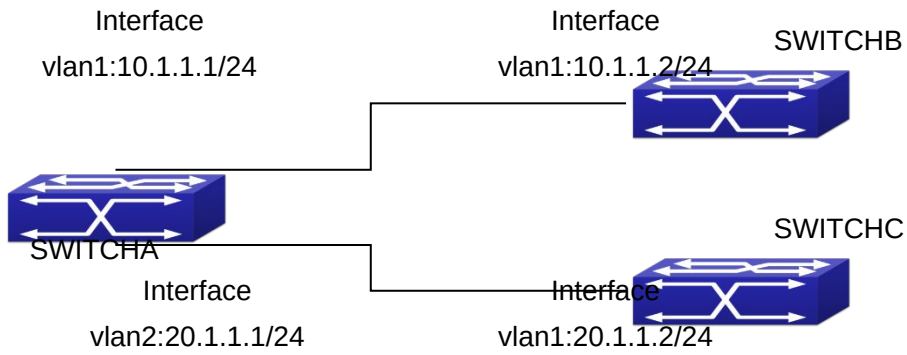


Figure 5-7 OSPF VPN Example

The figure above shows a network of three Layer 3 switches: SwitchA is the PE, and SwitchB and SwitchC are CE1 and CE2. The PE is connected to CE1 and CE2 through vlan1 and vlan2, respectively. Routing information is exchanged between the PE and the CEs through OSPF.

a) Layer 3 SwitchA as the PE:

Configure VPN routing/forwarding instances vpnb and vpnc.

```
SwitchA#config
SwitchA(config)#ip vrf vpnb
SwitchA(config-vrf)#
SwitchA(config-vrf)#exit
SwitchA#(config)
SwitchA(config)#ip vrf vpnc
SwitchA(config-vrf)#
SwitchA(config-vrf)#exit
```

Associate vlan 1 and vlan 2 with vpnb and vpnc, respectively, and configure their IP addresses.

```
SwitchA(config)#in vlan1
SwitchA(config-if-Vlan1)#ip vrf forwarding vpnb
SwitchA(config-if-Vlan1)#ip address 10.1.1.1 255.255.255.0
SwitchA(config-if-Vlan1)#exit
SwitchA(config)#in vlan2
SwitchA(config-if-Vlan2)#ip vrf forwarding vpnc
SwitchA(config-if-Vlan2)#ip address 20.1.1.1 255.255.255.0
SwitchA(config-if-Vlan2)#exit
```

Configure the OSPF instances associated with vpnb and vpnc, respectively.

```
SwitchA(config)#
SwitchA(config)#router ospf 100 vpnb
SwitchA(config-router)#network 10.1.1.0/24 area 0
SwitchA(config-router)#redistribute bgp
SwitchA(config-router)#exit
SwitchA(config)#router ospf 200 vpnc
SwitchA(config-router)#network 20.1.1.0/24 area 0
SwitchA(config-router)#redistribute bgp
```

b) Layer 3 SwitchB as CE1:

Configure the IP address of Ethernet E 1/0/2.

```
SwitchB#config
SwitchB(config)# interface Vlan1
SwitchB(config-if-vlan1)# ip address 10.1.1.2 255.255.255.0
SwitchB (config-if-vlan1)exit
```

Enable OSPF and configure the OSPF network segments.

```
SwitchB(config)#router ospf
```

```
SwitchB(config-router-rip)#network 10.1.1.0/24 area 0
SwitchB(config-router-rip)#exit
```

c) Layer 3 SwitchC as CE2:

Configure the IP address of Ethernet E 1/0/2.

```
SwitchC#config
SwitchC(config)# interface Vlan1
SwitchC(config-if-vlan1)# ip address 20.1.1.2 255.255.255.0
SwitchC(config-if-vlan1)#exit
```

Enable OSPF and configure the OSPF network segments.

```
SwitchC(config)#router ospf
SwitchC(config-router)#network 20.1.1.0/24 area 0
SwitchC(config-router)#exit
```

5.4.4 OSPF Troubleshooting

OSPF may fail to work properly because of errors such as physical connection problems or configuration errors. Pay attention to the following:

- First, ensure that the physical connections are correct.
- Second, ensure that the interface and link protocol are up (use the **show interface** command).
- Configure IP addresses from different network segments on different interfaces.
- Then enable OSPF (use the **router-ospf** command) and configure the OSPF area on the corresponding interfaces.
- Next, check the following OSPF requirements: the OSPF backbone area must be contiguous; if it is not, use virtual links to make it contiguous. Non-zero areas can connect to other non-zero areas only through area 0. A border Layer 3 switch has some interfaces in area 0 and others in non-zero areas. A DR must be specified for multi-access networks such as broadcast networks.

5.5 BGP

5.5.1 Introduction to BGP

BGP (Border Gateway Protocol) is a dynamic inter-autonomous system (inter-AS) routing protocol. Its basic function is to exchange routing information automatically without creating loops. By exchanging reachability information that carries the AS path attribute (the sequence of AS numbers), BGP can build a topology map of the autonomous systems, eliminate routing loops, and enforce user-configured policies. Generally, the switches within an AS use one or more IGPs (Interior Gateway Protocols), such as RIP and OSPF, to exchange routing information inside the AS, and use an EGP (Exterior Gateway Protocol) to exchange information between ASes. BGP is an EGP. An AS is usually under a single administrative authority. BGP is often used on switches between ISPs or between the divisions of a multinational corporation.

BGP has been in use since 1989. Its first three versions are RFC1105 (BGP-1), RFC1163 (BGP-2), and RFC1267 (BGP-3). The most widely used version is RFC1771 (BGP-4). The switch supports BGP-4.

1. Characteristics of BGP-4

BGP-4 suits distributed network structures and supports Classless Inter-Domain Routing (CIDR). BGP-4 has become the de facto standard exterior routing protocol of the global Internet. The features of BGP-4 are as follows:

- BGP is an exterior routing protocol. Unlike interior routing protocols such as OSPF and RIP, BGP cannot discover or calculate routes, but it can control the propagation of routes and select the best route.
- Routing loops are prevented by carrying AS path information in route updates.
- BGP uses TCP port 179 as its transport, which enhances the reliability of the protocol.
- BGP-4 supports CIDR (Classless Inter-Domain Routing), an important improvement over BGP-3. CIDR treats IP addresses in a new way: it does not distinguish Class A, Class B, and Class C networks. For example, the invalid Class C address 192.213.0.0 255.255.0.0 can be represented in CIDR as 192.213.0.0/16, which is a valid supernet. /16 means that the network number consists of the first 16 bits (from the left) of the address. CIDR simplifies route aggregation, which combines several different routes into one, so that a single route can be advertised instead of several, reducing the size of the routing table.
- When routes change, BGP sends only incremental updates. This greatly reduces the bandwidth used by BGP and makes it suitable for transmitting large amounts of routing information across the Internet.
- For political and economic reasons, each AS needs to filter and control routes. BGP-4 provides a rich set of routing policies, which makes it highly extensible and supports the growth of the Internet.

2. Overview of BGP-4 Operation

Unlike RIP and OSPF, BGP is connection-oriented. BGP switches must establish a connection before exchanging routing information. BGP operation is message-driven, and BGP messages fall into four types:

Open message: The first message sent after a TCP connection is established. It is used to establish a BGP session between peers. Some parameters in the Open message are used to negotiate whether a session can be established between the BGP peers.

Keepalive message: Checks whether the connection is available. It is usually sent periodically to maintain the BGP connection. If neither a Keepalive message nor an Update message is received within the hold time, the BGP connection is closed.

Update message: The most important message in BGP. It is used to exchange routing information between peers. Switches exchange not only new routing information but also unavailable or withdrawn routes. An Update message consists of three parts: unreachable (withdrawn) routes, NLRI (Network Layer Reachability Information), and path attributes.

Notification message: The error notification message. When a BGP speaker receives this message, it shuts down the BGP connection with the neighbor.

BGP-4 is connection-oriented. BGP operates as an upper-layer protocol on specific devices. When a neighbor is detected, a TCP session is established and maintained, and the routing tables are then exchanged and synchronized. The entire BGP routing table is exchanged only at system initialization. After that, routing information is exchanged only when updates are available, and only incremental Update messages are sent. BGP-4 maintains links and sessions by periodically sending and receiving Keepalive messages to check whether the connections are normal.

Switches that participate in a BGP session are called BGP speakers. A BGP speaker continuously receives or generates new routing information and advertises it to other BGP speakers. When a BGP speaker receives a new route from another AS, and this route is better than the currently known route or no acceptable route exists, it sends the route to all other BGP speakers in its AS. A BGP speaker calls the other speakers with which it exchanges routing information neighbors or peers. Several related neighbors can form a peer group. BGP operates on the switches in the following two modes:

- IBGP: Internal BGP
- EBGP: External BGP

When BGP runs within the same AS, it is called IBGP; when it runs between different ASes, it is called EBGP. Generally, external neighbors are physically connected, whereas internal neighbors can be anywhere in the AS. The difference lies in how BGP handles the routing information. A device checks the AS number in the Open message from a neighbor to decide whether to treat the neighbor as an external or an internal neighbor.

IBGP is used within an AS and sends messages to all BGP neighbors in the AS. IBGP exchanges AS routing information within a large organization. Note that the switches in the AS do not need to be physically connected; switches can be neighbors as long as they are in the same AS. Because BGP cannot discover routes, the routing tables of other internal routing protocols (such as static, direct, OSPF, and RIP routes) must contain the neighbor IP addresses, and these routes are used to exchange information between BGP speakers. To avoid routing loops, a BGP speaker that receives a route from an internal neighbor does not advertise this route to other internal neighbors.

EBGP is used between ASes and sends routing information to BGP neighbors in external ASes. EBGP requires a physical connection over a shared medium. Therefore, the border devices between two ASes usually run EBGP. When a BGP speaker receives routing information from an external neighbor, it advertises these routes to its internal neighbors.

3. Route Attributes

BGP-4 can share and query the IP routing table through related mechanisms, but it also maintains its own routing table. In the BGP routing table, each route has a network number, the list of ASes it has passed through (the AS path), and other routing attributes (such as origin). The routing attributes used by BGP-4 are complex and can be used as metrics for path selection.

4. BGP Route Selection Policy

When BGP receives advertisements for the same route from several neighbors, it must select the best route after route filtering. This is called the BGP route selection process, which starts only when the following conditions are met:

- The next hop of the route must be reachable; that is, the routing table must contain a route to the next hop.
- BGP must be synchronized with the IGP (unless synchronization is disabled; this applies only to IBGP).

BGP route selection is based on BGP attributes. When several routes lead to the same destination, BGP selects the best route as follows:

1. Select the route with the highest weight.

2. If the weights are equal, select the route with the highest local preference.

3. If the local preferences are equal, select the route originated by the local switch.

4. If the local preferences are equal and no route is originated by the local switch, select the route with the shortest AS path.

5. If the AS paths are equal, select the route with the lowest origin type (IGP < EGP < INCOMPLETE).

6. If the origin types are equal, select the route with the lowest MED. Unless the **bgp always-compare-med** command is enabled, this comparison applies only to routes from the same neighboring AS.

7. If the MEDs are equal, EBGP routes are preferred over confederation-external routes, and confederation-external routes are preferred over IBGP routes.

8. If the routes are still equal, the BGP router ID is used as the tiebreaker: the route from the lowest router ID is preferred.

5.5.2 BGP Configuration Task List

BGP configuration tasks include basic and advanced tasks. The basic BGP configuration tasks are:

1．Enable BGP routing (required)
2．Configure BGP neighbors (required)
3．Manage routing policy changes
4．Configure BGP weights
5．Configure neighbor-based BGP route filtering policies
6．Configure the BGP next hop
7．Configure EBGP multi-hop
8．Configure the BGP session identifier
9．Configure the BGP version

The advanced BGP configuration tasks are:

1．Use route maps to modify routes
2．Configure route aggregation
3．Configure BGP community filtering
4．Configure a BGP confederation
5．Configure a route reflector
6．Configure peer groups
7．Configure neighbor and peer group parameters
8．Adjust BGP timers
9．Adjust the BGP advertisement interval
10．Configure the default local preference
11．Allow the default route to be advertised
12．Configure the BGP MED value
13．Configure BGP route redistribution
14．Configure BGP route dampening
15．Configure BGP capability negotiation
16．Configure a route server
17．Configure path selection rules
18．Configure redistribution of OSPF routes into BGP
(1) Enable redistribution of OSPF routes into BGP
(2) Display and debug the configuration of OSPF route redistribution into BGP

I. Basic BGP Configuration Tasks

1. Enable BGP routing

Command	Explanation
Global Mode	
router bgp <*as-id*> **no router bgp <*as-id*>**	Enables BGP. The no command disables the BGP process.
BGP Router Mode	
bgp *asnotation asdot* ***no* bgp asnotation asdot**	Displays AS numbers and matches regular expressions in ASDOT notation. The no command cancels this notation.

address-family ipv4 {unicast| multicast|vrf <*vrf-nam*>} **no address-family ipv4 {unicast| multicast|vrf <*vrf-nam*>}**	Creates an IPv4 address family for BGP and enters the BGP-VPN view. No IPv4 address family is created by default.
Address Family Mode	
network <ip-address/M> **no network <ip-address/M>**	Sets the network that BGP advertises. The no command cancels the advertisement of the network.

2. Configure BGP neighbors

Command	Explanation
BGP Router Mode	
neighbor {<*ip-address*>|<*TAG*>} **remote-as <*as-id*>** **no neighbor {<*ip-address*>|<*TAG*>}** **[remote-as <*as-id*>]**	Specifies a BGP neighbor. The no command deletes the neighbor.

3. Manage routing policy changes

(1) Configure hard reconfiguration.

Command	Explanation
Admin Mode	
clear ip bgp {<*>|<*as-id*>| external| **peer-group <*NAME*>|<*ip-address*>}**	Performs a hard reconfiguration.

(2) Configure outbound soft reconfiguration.

Command	Explanation
Admin Mode	
clear ip bgp {<*>|<*as-id*>| external| **peer-group <NAME>|<*ip-address*>} soft out**	Performs an outbound soft reconfiguration.

(3) Configure inbound soft reconfiguration.

Command	Explanation
Address Family Mode	
neighbor { <*ip-address*> | <*TAG*> } soft-reconfiguration inbound **no neighbor { <*ip-address*> | <*TAG*> } soft-reconfiguration inbound**	Stores the routing information received from neighbors and peers. The no command cancels the storage of routing information.
Admin Mode	
clear ip bgp {<*>|<*as-id*>| external|peer-group <*NAME*>|<*ip-address*>} soft in	Performs an inbound BGP soft reconfiguration.

4. Configure BGP weights

Command	Explanation
BGP Router Mode	
neighbor { <*ip-address*> | <*TAG*> } weight <*weight*> **no neighbor { <ip-address> | <TAG> }**	Configures the weight of a BGP neighbor. The no command restores the default weight.

5. Configure neighbor-based BGP route filtering policies

Command	Explanation
Address Family Mode	

neighbor {<*ip-address*>|<*TAG*>} distribute-list {<*1-199*>|<*1300-2699*>|<*WORD*>} {in|out} **no neighbor {<*ip-address*>|<*TAG*>} distribute-list {<*1-199*>|<*1300-2699*>|<*WORD*>} {in|out}**	Filters the routing updates of a neighbor. The no command cancels the route filtering.

6. Configure the next hop

1) Set the next hop to the switch's own address

Command	Explanation
Address Family Mode	
neighbor { <*ip-address*> | <*TAG*> } **next-hop-self** **no neighbor { <*ip-address*> | <*TAG*> }** **next-hop-self**	Sets the next hop of advertised routes to the switch's own address. The no command cancels this setting.

2) Override the default next hop with a route map

Command	Explanation
Route Map Mode	
set ip next-hop <*ip-address*> **no set ip next-hop**	Sets the next-hop attribute of outbound routes. The no command cancels this setting.

7. Configure EBGP multi-hop

If external neighbors are not directly connected, use the following command to configure EBGP multi-hop.

Command	Explanation
BGP Router Mode	
neighbor {<*ip-address*>|<*TAG*>} ebgp-multihop [<*1-255*>] **no neighbor {<*ip-address*>|<*TAG*>} ebgp-multihop [<*1-255*>]**	Allows EBGP connections with neighbors on networks that are not directly connected. The no command cancels this setting.

8. Configure the BGP session identifier

Command	Explanation
BGP Router Mode	
bgp router-id <*ip-address*> **no bgp router-id**	Configures the router ID. The no command restores the default value.

9. Configure the BGP version

Command	Explanation
BGP Router Mode	
neighbor {<*ip-address*> | <*TAG*>} version <value> **no neighbor {<*ip-address*> | <*TAG*>} version**	Sets the BGP version used with the neighbor. The no command restores the default setting. Currently, only version 4 is supported.

II. Advanced BGP Configuration Tasks

1. Use Route Maps to Modify Routes

Command	Explanation
Address Family Mode	
neighbor { <*ip-address*> | <*TAG*> } route-map <*map-name* > {in | out} **no neighbor { <*ip-address*> | <*TAG*> } route-map <*map-name* > {in | out}**	Applies a route map to incoming or outgoing routes. The no command removes the route map.

2. Configure Route Aggregation

Command	Explanation
Address Family Mode	
aggregate-address <*ip-address/M*> [summary-only] [as-set] **no aggregate-address <*ip-address/M*> [summary-only] [as-set]**	Creates an aggregate entry in the BGP routing table. The no command deletes the aggregate entry.

3. Configure BGP Community Filtering

Command	Explanation
Address Family Mode	
neighbor {<*ip-address*> | <*TAG*>} send-community **no neighbor {<*ip-address*> | <*TAG*>} send-community**	Allows routing updates with community attributes to be sent to the BGP neighbor. The no command sends routes without community attributes.

4. Configure a BGP Confederation

Command	Explanation
BGP Router Mode	
bgp confederation identifier <*as-id*> **no bgp confederation identifier <*as-id*>**	Configures a BGP AS confederation identifier. The no command deletes the BGP AS confederation identifier.
bgp confederation peers <*as-id*> [<*as-id*>..] **no bgp confederation peers <*as-id*> [<*as-id*>..]**	Configures the ASes that belong to the AS confederation. The no command deletes the AS from the AS confederation.

5. Configure a Route Reflector

(1) **Use the following command to configure a route reflector and its clients.**

Command	Explanation
Address Family Mode	
neighbor <*ip-address*> route-reflector-client **no neighbor <*ip-address*> route-reflector-client**	Configures the switch as a route reflector and specifies a client. The no command deletes the client.

(2) **If a cluster has more than one route reflector, use the following command to configure the cluster ID.**

Command	Explanation
BGP Router Mode	
bgp cluster-id <*cluster-id*> **no bgp cluster-id**	Configures the cluster ID. The no command cancels the cluster ID configuration.

(3) **To enable client-to-client route reflection, use the following command.**

Command	Explanation
BGP Router Mode	
bgp client-to-client reflection **no bgp client-to-client reflection**	Allows the route reflector to reflect routes from client to client. The no command prohibits client-to-client reflection.

6. Configure Peer Groups

(1) **Create peer groups**

Command	Explanation
BGP Router Mode	

neighbor <*TAG*> peer-group **no neighbor <*TAG*> peer-group**	Creates a peer group. The no command deletes the peer group.

(2) **Add neighbors to peer groups**

Command	Explanation
BGP Router Mode	
neighbor <*ip-address*> peer-group <*TAG*> **no neighbor <*ip-address*> peer-group <*TAG*>**	Adds a neighbor to the peer group. The no command removes the specified member.

7. Configure Neighbor and Peer Group Parameters

Command	Explanation
BGP Router Mode	
neighbor {<*ip-address*> | <*TAG*>} remote-as <*as-id*> **no neighbor {<*ip-address*> | <*TAG*>} remote-as <*as-id*>**	Specifies a BGP neighbor. The no command deletes the neighbor.
neighbor { <*ip-address*> | <*TAG*> } description <*.LINE*> **no neighbor { <*ip-address*> | <*TAG*> } description**	Associates a description with a neighbor. The no command deletes the description.
neighbor { <*ip-address*> | <*TAG*> } timers <*keep alive*> <*holdtime*> **no neighbor { <*ip-address*> | <*TAG*> } timers**	Configures the keepalive and hold-time timers of a specific neighbor. The no command restores the default values.
neighbor {<*ip-address*> | <*TAG*>} advertisement-interval <*seconds*> **no neighbor {<*ip-address*> | <*TAG*>} advertisement-interval**	Configures the minimum interval for sending BGP routing updates. The no command restores the default value.
neighbor {<*ip-address*> | <*TAG*>} ebgp-multihop [<*1-255*>] **no neighbor {<*ip-address*> | <*TAG*>} ebgp-multihop**	Allows EBGP connections with networks that are not directly connected. The no command cancels this setting.
neighbor { <*ip-address*> | <*TAG*> } weight <weight> **no neighbor { <*ip-address*> | <*TAG*> } weight**	Configures the weight of a BGP neighbor. The no command restores the default weight.
neighbor { <*ip-address*> | <*TAG*> } version <*value*> **no neighbor { <*ip-address*> | <*TAG*> } version**	Specifies the BGP version used to communicate with the BGP neighbor. The no command restores the default setting.
neighbor { <*ip-address*> | <*TAG*> } shutdown **no neighbor { <*ip-address*> | <*TAG*> } shutdown**	Shuts down a BGP neighbor or peer group. The no command re-enables the BGP neighbor or peer group.
Address Family Mode	
neighbor { <*ip-address*> | <*TAG*> } default-originate [route-map <*NAME*>] **no neighbor { <*ip-address*> | <*TAG*> } default-originate [route-map <*NAME*>]**	Allows the default route 0.0.0.0 to be sent. The no command cancels sending the default route.
neighbor { <*ip-address*> | <*TAG*> } send-community **no neighbor { <*ip-address*> | <*TAG*> } send-community**	Configures sending community attributes to the neighbor.
neighbor { <*ip-address*> | <*TAG*> } distribute-list { <*access-list-number*> | <*name*> } { in | out }	Filters the routing updates of a neighbor. The no command cancels route filtering.

no neighbor { *<ip-address>* | *<TAG>* } distribute-list { *<access-list-number>* | *<name>* } { *in* | *out* }	
neighbor { *<ip-address>* | *<TAG>* } route-reflector-client **no neighbor { *<ip-address>* | *<TAG>* } route-reflector-client**	Configures the switch as a route reflector and specifies a client. The no command deletes the client.
neighbor { *<ip-address>* | *<TAG>* } next-hop-self **no neighbor { *<ip-address>* | *<TAG>* } next-hop-self**	Sets the next hop of advertised routes to the switch's own address. The no command cancels the setting.
neighbor { *<ip-address>* | *<TAG>* } route-map *<map-name>* {in | out} **no neighbor { *<ip-address>* | *<TAG>* } route-map *<map-name>* {in | out}**	Applies a route map to incoming or outgoing routes. The no command cancels the route map setting.
neighbor { *<ip-address>* | *<TAG>* } soft-reconfiguration inbound **no neighbor { *<ip-address>* | *<TAG>* } soft-reconfiguration inbound**	Stores the routing information received from neighbors or peers. The no command cancels the storage.

8. Adjust BGP Timers

（1） **Configure the BGP timers of all neighbors**

Command	Explanation
BGP Router Mode	
timers bgp *<keep alive>* *<holdtime>* **no timers bgp**	Configures the BGP timers of all neighbors. The no command restores the default values.

（2） **Configure the timers of a specific neighbor**

Command	Explanation
BGP Router Mode	
neighbor { *<ip-address>* | *<TAG>* } timers *<keep alive>* *<holdtime>* **no neighbor { *<ip-address>* | *<TAG>* } timers**	Configures the keepalive and hold-time timers of a specific neighbor. The no command restores the default values.

9. Adjust the BGP Advertisement Interval

Command	Explanation
BGP Router Mode	
neighbor {*<ip-address>* | *<TAG>*} advertisement-interval <seconds> **no neighbor {*<ip-address>* | *<TAG>*} advertisement-interval**	Configures the minimum interval between BGP routing updates. The no command restores the default setting.

10. Configure the Local Preference Value

Command	Explanation
BGP Router Mode	
bgp default local-preference *<value>* **no bgp default local-preference**	Changes the default local preference. The no command restores the default value.

11. Enable Sending the Default Route

Command	Explanation

Address Family Mode	
neighbor { *<ip-address>* | *<TAG>* } default-originate **no neighbor { *<ip-address>* | *<TAG>* } default-originate**	Allows the default route 0.0.0.0 to be sent. The no command cancels sending the default route.

12. Configure the BGP MED Value

(1) **Configure the MED value**

Command	Explanation
Route Map Mode	
set metric *<metric-value>* **no set metric**	Configures the metric value. The no command restores the default value.

(2) **Enable MED comparison for paths from different ASes**

Command	Explanation
BGP Router Mode	
bgp always-compare-med **no bgp always-compare-med**	Allows MED comparison between routes from different ASes. The no command prohibits the comparison.

13. Configure BGP Route Redistribution

Command	Explanation
Address Family Mode	
redistribute { connected | static | rip | ospf} [metric *<metric>*] [route-map *<NAME>*] **no redistribute { connected | static | rip | ospf}**	Redistributes IGP routes into BGP and optionally specifies the metric and route map for the redistributed routes. The no command cancels the redistribution.

14. Configure Route Dampening

Command	Explanation
Address Family Mode	
bgp dampening [*<1-45>*] [*<1-20000>* *<1-20000>* *<1-255>*] [*<1-45>*] **no bgp dampening**	Enables BGP route dampening with the specified parameters. The no command stops route dampening.

15. Configure BGP Capability Negotiation

Command	Explanation
BGP Router Mode	
neighbor {*<ip-address>*|*<TAG>*} capability {dynamic | route-refresh} **no neighbor {*<ip-address>*|*<TAG>*} capability {dynamic | route-refresh}** **neighbor {*<ip-address>*|*<TAG>*} dont-capability-negotiate** **no neighbor {*<ip-address>*|*<TAG>*} dont-capability-negotiate** **neighbor {*<ip-address>*|*<TAG>*} override-capability** **no neighbor {*<ip-address>*|*<TAG>*} override-capability** **neighbor {*<ip-address>*|*<TAG>*} strict-capability-match** **no neighbor {*<ip-address>*|*<TAG>*} strict-capability-match**	Configures BGP capability negotiation. BGP performs capability matching when it establishes a connection. The currently supported capabilities include route refresh, dynamic capability, outbound route filtering, and address family negotiation. These commands enable the capabilities, and their no forms disable them. The commands can also configure the switch to skip capability negotiation, to perform strict capability matching, or to ignore the negotiation results.
Address Family Mode	

neighbor {<*ip-address*>|<*TAG*>} capability orf prefix-list {<*both*>|<*send*>|<*receive*>} **no neighbor {<*ip-address*>|<*TAG*>} capability orf prefix-list {<*both*>|<*send*>|<*receive*>}**	

16. Configure a Route Server

Command	Explanation
Address Family Mode	
neighbor {<*ip-address*>|<*TAG*>} route-server-client **no neighbor {<*ip-address*>|<*TAG*>} route-server-client**	Configures the switch as a route server and specifies the clients it serves. In an EBGP environment, a route server reduces the number of peers that each client must configure. The no command deletes the client.

17. Configure Path Selection Rules

Command	Explanation
BGP Router Mode	
bgp always-compare-med **no bgp always-compare-med** **bgp bestpath as-path ignore** **no bgp bestpath as-path ignore** **bgp bestpath compare-confed-aspath** **no bgp bestpath compare-confed-aspath** **bgp bestpath compare-routerid** **no bgp bestpath compare-routerid** **bgp bestpath med {[confed] [missing-is-worst]}** **no bgp bestpath med {[confed] [missing-is-worst]}**	Modifies the BGP best-path selection rules: compares MED in an EBGP environment, ignores the AS-path length, compares the confederation AS-path length, compares router IDs, compares the confederation MED, and so on. The no commands restore the default path selection rules.

18. Configure Redistribution of OSPF Routes into BGP

(1) Enable redistribution of OSPF routes into BGP

Command	Explanation
Address Family Mode	
redistribute ospf [<*process-id*>] [route-map<*word*>] **no redistribute ospf [<process-id>]**	Enables redistribution of OSPF routes into BGP. The no command disables the redistribution.

(2) Display and debug the configuration of OSPF route redistribution into BGP

Command	Explanation
Admin Mode and Configuration Mode	
show ip bgp redistribute	Displays the configuration of OSPF route redistribution into BGP.
Admin Mode	
debug bgp redistribute message send **no debug bgp redistribute message send** **debug bgp redistribute route receive** **no debug bgp redistribute route receive**	Enables debugging of messages sent by BGP for OSPF route redistribution. The no command disables the debugging. Enables debugging of messages received from NSM for OSPF route redistribution. The no command disables the debugging.

5.5.3 BGP Configuration Examples

5.5.3.1 Example 1: Configure BGP Neighbors

SwitchB, SwitchC, and SwitchD are in AS200, and SwitchA is in AS100. SwitchA and SwitchB are on the same network segment. SwitchB and SwitchD are not physically connected.

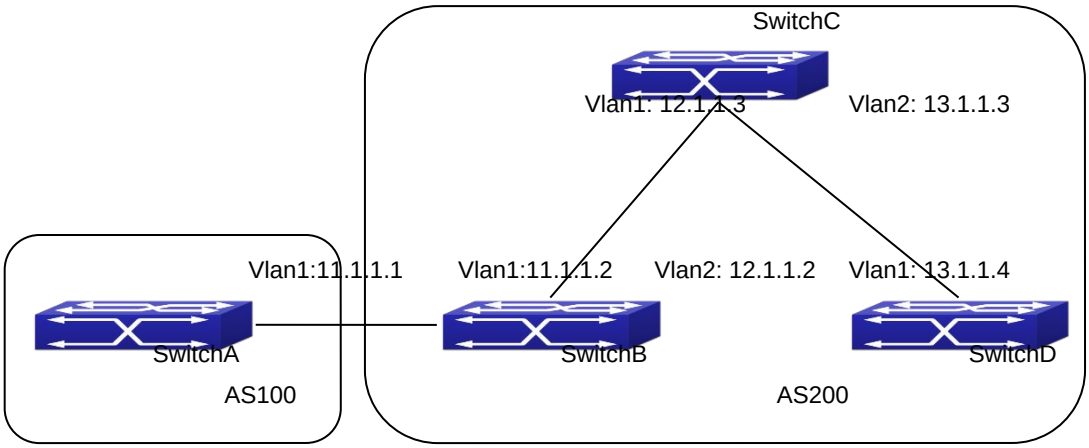


Figure 5-9 BGP Network Topology

Configuration of SwitchA:

```
SwitchA(config)#router bgp 100
SwitchA(config-router)#neighbor 11.1.1.2 remote-as 200
SwitchA(config-router)#exit
```

Configuration of SwitchB:

```
SwitchB(config)#router bgp 200
SwitchB(config-router)#address-family ipv4 unicast
SwitchB(config-bgp-ipv4)#network 11.0.0.0/8
SwitchB(config-bgp-ipv4)#network 12.0.0.0/8
SwitchB(config-bgp-ipv4)#network 13.0.0.0/8
SwitchB(config-bgp-ipv4)#exit-address-family
SwitchB(config-router)#neighbor 11.1.1.1 remote-as 100
SwitchB(config-router)#neighbor 12.1.1.3 remote-as 200
SwitchB(config-router)#neighbor 13.1.1.4 remote-as 200
SwitchB(config-router)#exit
```

Configuration of SwitchC:

```
SwitchC(config)#router bgp 200
SwitchC(config-router)#address-family ipv4 unicast
SwitchC(config-bgp-ipv4)#network 12.0.0.0/8
SwitchC(config-bgp-ipv4)#network 13.0.0.0/8
SwitchC(config-bgp-ipv4)#exit-address-family
SwitchC(config-router)#neighbor 12.1.1.2 remote-as 200
SwitchC(config-router)#neighbor 13.1.1.4 remote-as 200
SwitchC(config-router)#exit
```

Configuration of SwitchD:

```
SwitchD(config)#router bgp 200
SwitchD(config-router)#address-family ipv4 unicast
SwitchD(config-bgp-ipv4)#network 13.0.0.0/8
SwitchD(config-bgp-ipv4)#exit-address-family
SwitchD(config-router)#neighbor 12.1.1.2 remote-as 200
SwitchD(config-router)#neighbor 13.1.1.3 remote-as 200
SwitchD(config-router)#exit
```

In this setup, the connection between SwitchB and SwitchA is EBGP, and the connections with SwitchC and SwitchD are IBGP. SwitchB and SwitchD can establish a BGP connection without a physical connection, provided that each has a reachable route to the other. This route can be obtained through a static route or an IGP.

5.5.3.2 Example 2: Configure BGP Aggregation

This example configures route aggregation. First, use the redistribute command to redistribute static routes into the BGP routing table:

```
SwitchB(config)#ip route 193.0.0.0/24 11.1.1
SwitchB(config)#router bgp 100
SwitchB(config-router)#address-family ipv4 unicast
SwitchB(config-bgp-ipv4)#redistribute static
```

When at least one route falls within the specified range, the following configuration creates an aggregate route in the BGP routing table. The aggregate route is regarded as originating from this AS. More specific routes within 193.0.0.0 are still advertised.

```
SwitchB(config)#router bgp 100
SwitchB(config-router)#address-family ipv4 unicast
SwitchB(config-bgp-ipv4)#aggregate-address 193.0.0.0/16
```

Alternatively, modify the aggregate-address command as follows. The switch then advertises only the aggregate route 193.0.0.0 and suppresses the more specific routes to all neighbors.

```
SwitchB(config-bgp-ipv4)#aggregate-address 193.0.0.0/16 summary-only
```

5.5.3.3 Example 3: Configure BGP Community Attributes

In the following example, the route map set-community is applied to outgoing updates to neighbor 16.1.1.6. Routes permitted by access list 1 are assigned the community value 1111; other routes are advertised normally.

```
Switch(config)#router bgp 100
Switch(config-router)#neighbor 16.1.1.6 remote-as 200
Switch(config-router)#address-family ipv4 unicast
Switch(config-bgp-ipv4)#neighbor 16.1.1.6 route-map set-community out
Switch(config-bgp-ipv4)#exit
Switch(config)#route-map set-community permit 10
Switch(config-route-map)#match ip address 1
Switch(config-route-map)#set community 1111
Switch(config-route-map)#exit
Switch(config)#route-map set-community permit 20
Switch(config-route-map)#match ip address 2
Switch(config-route-map)#exit
Switch(config)#access-list 1 permit 11.1.0.0 0.0.255.255
Switch(config)#access-list 2 permit 0.0.0.0 255.255.255.255
Switch(config)#exit
Switch#clear ip bgp 16.1.1.6 soft out
```

In the following example, the MED and local preference of routes from neighbor 16.1.1.6 are set selectively based on their community values. All routes that match community list com1 have their MED set to 2000; com1 permits routes with the community value "100 200 300" or "900 901", and these routes may also carry other community attributes. All routes that pass community list com2 have their local preference set to 500. Routes that pass neither com1 nor com2 are rejected.

```
Switch(config)#router bgp 100
Switch(config-router)#neighbor 16.1.1.6 remote-as 200
Switch(config-router)#address-family ipv4 unicast
Switch(config-bgp-ipv4)#neighbor 16.1.1.6 route-map match-community in
Switch(config-bgp-ipv4)#exit
Switch(config)#route-map match-community permit 10
Switch(config-route-map)#match community com1
```

```
Switch(config-route-map)#set metric 2000
Switch(config-route-map)#exit
Switch(config)#route-map match-community permit 20
Switch(config-route-map)#match community com2
Switch(config-route-map)#set local-preference 500
Switch(config-route-map)#exit
Switch(config)#ip community-list com1 permit 100 200 300
Switch(config)#ip community-list com1 permit 900 901
Switch(config)#ip community-list com2 permit 88
Switch(config)#ip community-list com2 permit 90
Switch(config)#exit
Switch#clear ip bgp 16.1.1.6 soft out
```

5.5.3.4 Example 4: Configure a BGP Confederation

The following example configures an AS confederation. As shown in the figure, SwitchB and SwitchC establish an IBGP connection. SwitchD belongs to AS 20. SwitchB and SwitchD establish an EBGP connection within the AS confederation. AS10 and AS20 form an AS confederation with the AS number 200. SwitchA belongs to AS100, and SwitchB establishes an EBGP connection with it as AS200.

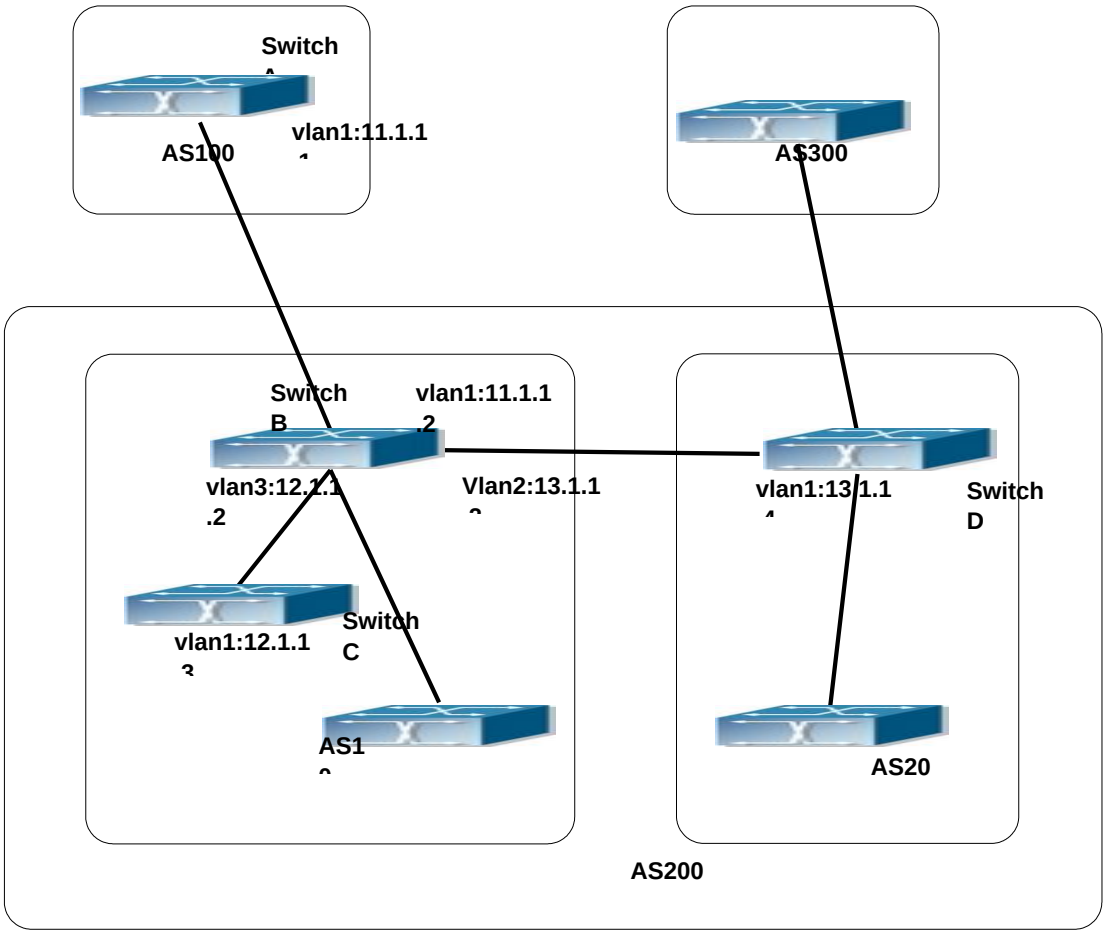


Figure 5-10 Confederation Configuration Topology

The configurations are as follows:

```
SwitchA:
SwitchA(config)#router bgp 100
SwitchA(config-router)#neighbor 11.1.1.2 remote-as 200
SwitchB:
SwitchB (config)#router bgp 10
SwitchB (config-router)#bgp confederation identifier 200
SwitchB (config-router)#bgp confederation peers 20
```

```
SwitchB (config-router)#neighbor 12.1.1.3 remote-as 10
SwitchB (config-router)#neighbor 13.1.1.4 remote-as 20
SwitchB (config-router)#neighbor 11.1.1.1 remote-as 100
SwitchC:
SwitchC (config)#router bgp 10
SwitchC (config-router)#bgp confederation identifier 200
SwitchC (config-router)#bgp confederation peers 20
SwitchC (config-router)#neighbor 12.1.1.2 remote-as 10
SwitchD:
SwitchD (config)#router bgp 20
SwitchD (config-router)#bgp confederation identifier 200
SwitchD (config-router)#bgp confederation peers 10
SwitchD (config-router)#neighbor 13.1.1.2 remote-as 10
```

5.5.3.5 Example 5: Configure a BGP Route Reflector

The following example configures route reflectors. As shown in the figure, SwitchA, SwitchB, SwitchC, SwitchD, SWE, SWF, and SWG belong to AS100 and establish IBGP connections. SwitchC establishes an EBGP connection with AS200, and SwitchA establishes an EBGP connection with AS300. SwitchC, SwitchD, and SWG act as route reflectors.

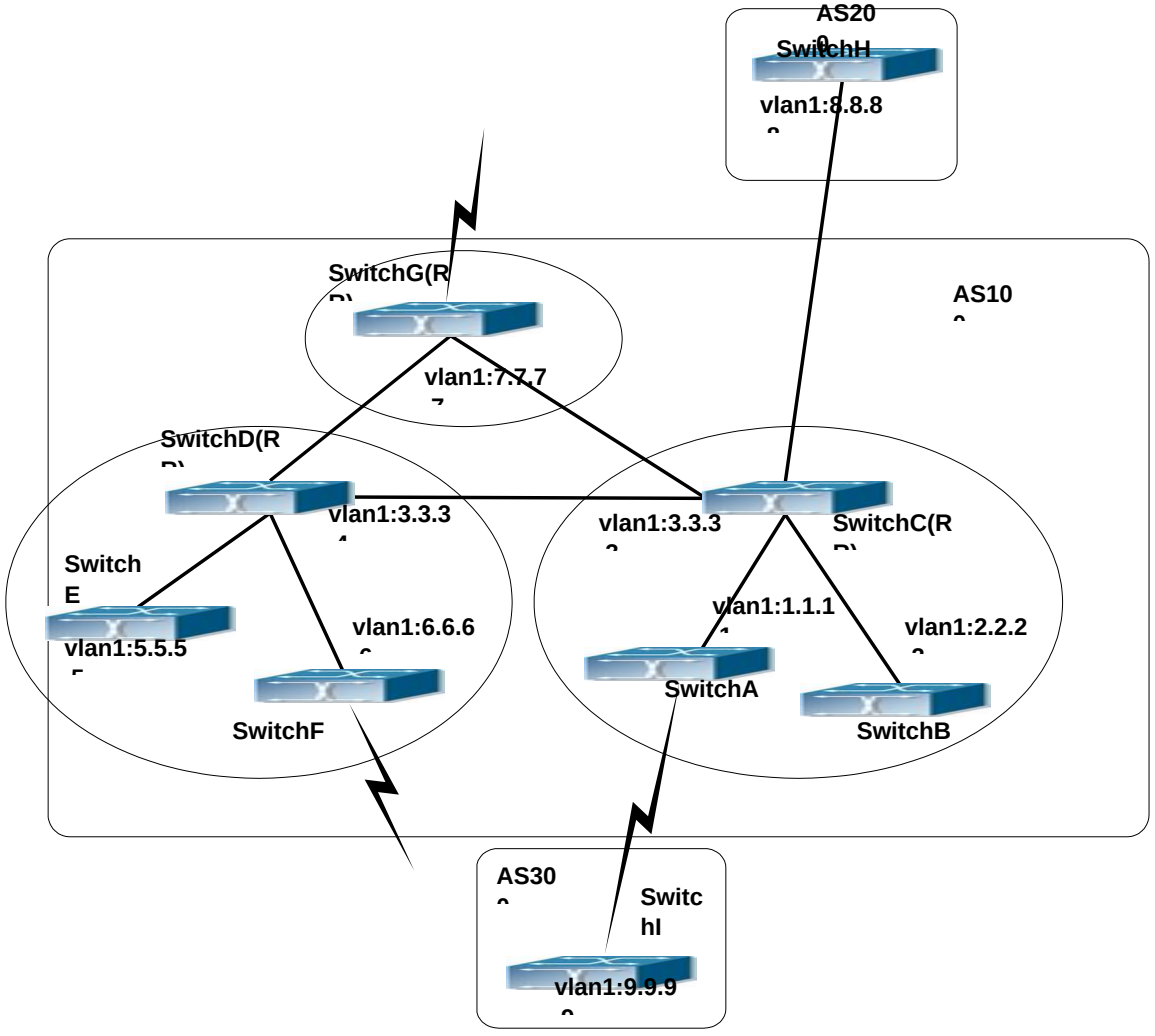


Figure 5-11 Route Reflector Topology

The configurations are as follows:

Configuration of SwitchC:

```
SwitchC (config)#router bgp 100
SwitchC (config-router)#neighbor 1.1.1.1 remote-as 100
SwitchC (config-router)#address-family ipv4 unicast
SwitchC (config-bgp-ipv4)#neighbor 1.1.1.1 route-reflector-client
SwitchC (config-bgp-ipv4)#exit-address-family
SwitchC (config-router)#neighbor 2.2.2.2 remote-as 100
SwitchC (config-router)#address-family ipv4 unicast
SwitchC (config-bgp-ipv4)#neighbor 2.2.2.2 route-reflector-client
SwitchC (config-bgp-ipv4)#exit-address-family
SwitchC (config-router)#neighbor 7.7.7.7 remote-as 100
SwitchC (config-router)#neighbor 3.3.3.4 remote-as 100
SwitchC (config-router)#neighbor 8.8.8.8 remote-as 200
```

Configuration of SwitchD:

```
SwitchD (config)#router bgp 100
SwitchD (config-router)#neighbor 5.5.5.5 remote-as 100
SwitchD (config-router)#address-family ipv4 unicast
SwitchD (config-bgp-ipv4)#neighbor 5.5.5.5 route-reflector-client
SwitchD (config-bgp-ipv4)#exit-address-family
SwitchD (config-router)#neighbor 6.6.6.6 remote-as 100
SwitchD (config-router)#address-family ipv4 unicast
SwitchD (config-bgp-ipv4)#neighbor 6.6.6.6 route-reflector-client
SwitchD (config-bgp-ipv4)#exit-address-family
SwitchD (config-router)#neighbor 3.3.3.3 remote-as 100
SwitchD (config-router)#neighbor 7.7.7.7 remote-as 100
```

Configuration of SwitchA:

```
SwitchA(config)#router bgp 100
SwitchA(config-router)#neighbor 1.1.1.2 remote-as 100
SwitchA(config-router)#neighbor 9.9.9.9 remote-as 300
```

SwitchA does not need to establish IBGP connections with all switches in AS100 and can still receive BGP routes from the other switches in the AS.

5.5.3.6 Example 6: Configure BGP MED

The following example configures MED. As shown in the figure, SwitchA belongs to AS100, SwitchB belongs to AS400, and SwitchC and SwitchD belong to AS300.

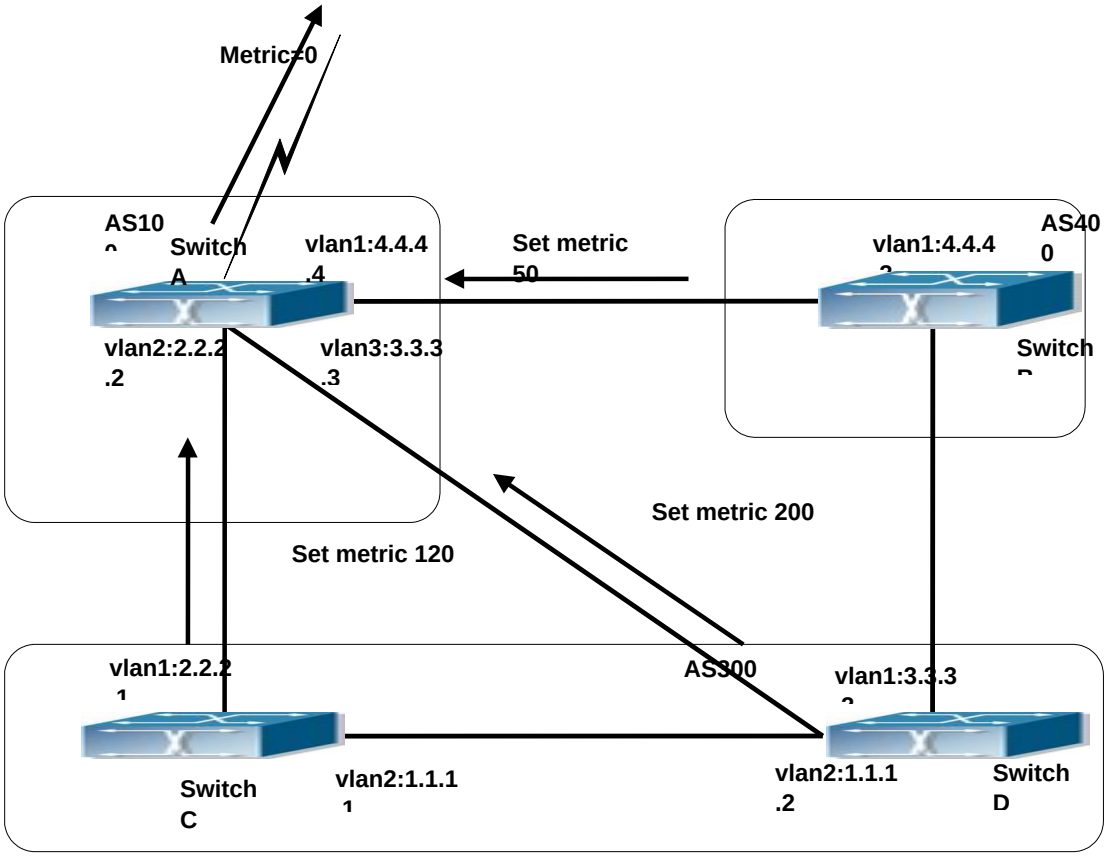


Figure 5-12 MED Configuration Topology

Configuration of SwitchA:

```
SwitchA(config)#router bgp 100
SwitchA(config-router)#neighbor 2.2.2.1 remote-as 300
SwitchA(config-router)#neighbor 3.3.3.2 remote-as 300
SwitchA(config-router)#neighbor 4.4.4.3 remote-as 400
```

Configuration of SwitchC:

```
SwitchC (config)#router bgp 300
SwitchC (config-router)#neighbor 2.2.2.2 remote-as 100
SwitchC (config-router)#address-family ipv4 unicast
SwitchC (config-bgp-ipv4)#neighbor 2.2.2.2 route-map set-metric out
SwitchC (config-bgp-ipv4)#exit-address-family
SwitchC (config-router)#neighbor 1.1.1.2 remote-as 300
SwitchC (config-router)#exit
SwitchC (config)#route-map set-metric permit 10
SwitchC (config-router-map)#set metric 120
```

Configuration of SwitchD:

```
SwitchD (config)#router bgp 300
SwitchD (config-router)#neighbor 3.3.3.3 remote-as 100
SwitchD (config-router)#address-family ipv4 unicast
SwitchD (config-bgp-ipv4)#neighbor 3.3.3.3 route-map set-metric out
SwitchD (config-bgp-ipv4)#exit-address-family
SwitchD (config-router)#neighbor 1.1.1.1 remote-as 300
SwitchD (config-router)#exit
SwitchD (config)#route-map set-metric permit 10
SwitchD (config-router-map)#set metric 200
```

Configuration of SwitchB:

```
SwitchB (config)#router bgp 400
```

```
SwitchB (config-router)#neighbor 4.4.4.4 remote-as 100
SwitchB (config-router)#address-family ipv4 unicast
SwitchB (config-bgp-ipv4)#neighbor 4.4.4.4 route-map set-metric out
SwitchB (config-bgp-ipv4)#exit-address-family
SwitchB (config-router)#exit
SwitchB (config)#route-map set-metric permit 10
SwitchB (config-router-map)#set metric 50
```

After the above configuration, assume that SwitchB, SwitchC, and SwitchD each send a route to 12.0.0.0 to SwitchA, and that the three routes have the same attribute values up to the metric comparison in the BGP route selection process. The route with the lower metric is preferred, but metrics are compared only among routes from the same AS. For SwitchA, the route through SwitchC is preferred over the route through SwitchD. Because SwitchC and SwitchB are not in the same AS, SwitchA does not compare their metrics. To compare metrics across different ASes, use the **bgp always-compare-med** command. With this command configured, the route through SwitchB is the best route for SwitchA. To do this, add the following command on SwitchA:

```
SwitchA (config-router)#bgp always-compare-med
```

5.5.3.7 Example 7: BGP VPN

In an MPLS VPN configuration, BGP is part of the core routing system and is also an important means of supporting ILM and FTN entries on edge devices. On the switch, BGP together with LDP forms the foundation of MPLS VPN applications. LDP works on the WLAN side, and BGP does not function on routers that are not at the network edge.

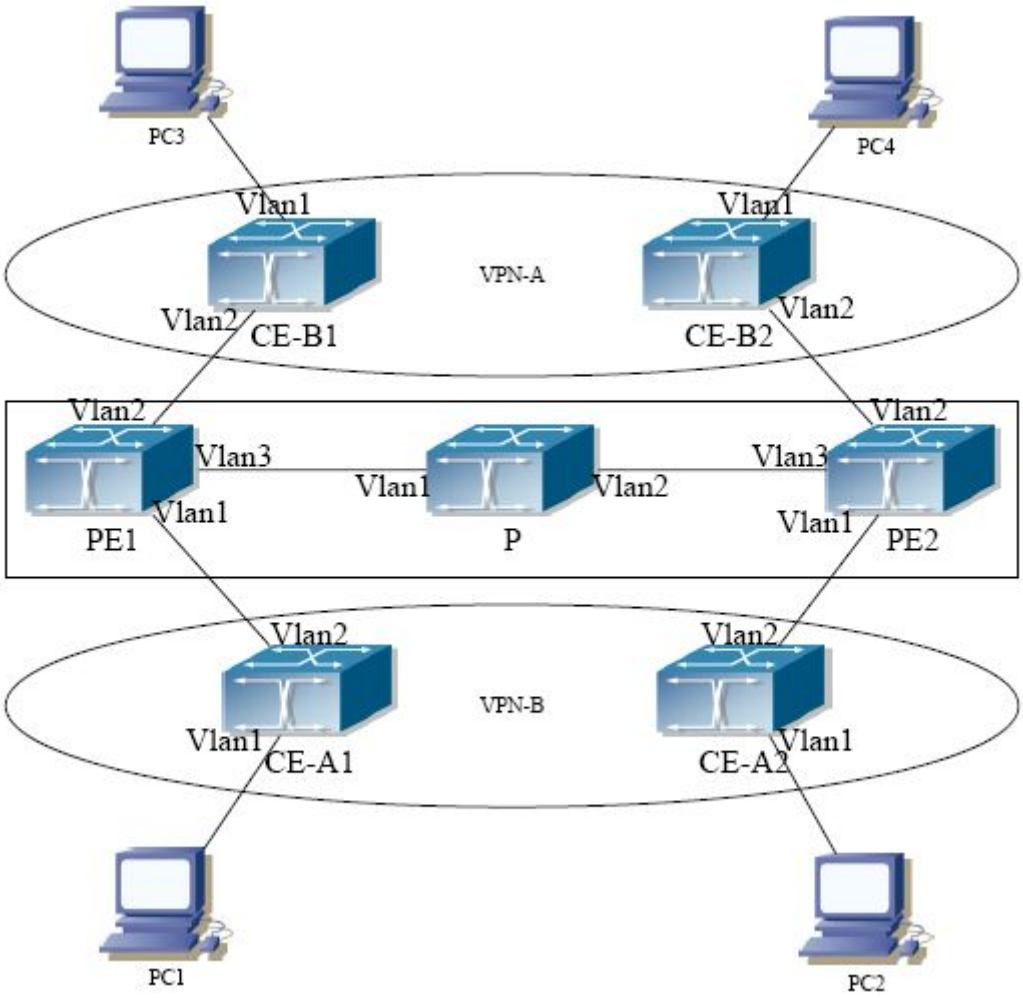


Figure 5-8 Example of MPLS VPN

As the figure shows, in a typical MPLS VPN application, the public network consists of PE1, P, and PE2, where MPLS is used for packet forwarding. VPN-A consists of CE-A1 and CE-A2, and VPN-B consists of CE-B1 and CE-B2. The two VPNs are isolated from each other. PE1 and PE2 are edge routers provided by the operator. CE-A1, CE-A2, CE-B1, and CE-B2 are user-side access switches. PC1 to PC4 represent network users. BGP runs in both the public and private networks. In the public network, VPN routing must be supported, and loopback interfaces are used for the connections.

Sample configurations are listed below.

Configuration of CE-A1:

```
CE-A1#config
CE-A1(config)#interface vlan 2
CE-A1(config-if-Vlan2)#ip address 192.168.101.2 255.255.255.0
CE-A1(config-if-Vlan2)#exit
CE-A1(config)#interface vlan 1
CE-A1(config-if-Vlan2)#ip address 10.1.1.1 255.255.255.0
CE-A1(config-if-Vlan2)#exit
CE-A1(config)#router bgp 60101
CE-A1(config-router)#neighbor 192.168.101.1 remote-as 100
CE-A1(config-router)#exit
```

Configuration of CE-A2:

```
CE-A2#config
CE-A2(config)#interface vlan 2
CE-A2(config-if-Vlan2)#ip address 192.168.102.2 255.255.255.0
CE-A2(config-if-Vlan2)#exit
CE-A2(config)#interface vlan 1
CE-A2(config-if-Vlan2)#ip address 10.1.2.1 255.255.255.0
CE-A2(config-if-Vlan2)#exit
CE-A2(config)#router bgp 60102
CE-A2(config-router)#neighbor 192.168.102.1 remote-as 100
CE-A2(config-router)#exit
```

Configuration of CE-B1:

```
CE-B1#config
CE-B1(config)#interface vlan 2
CE-B1(config-if-Vlan2)#ip address 192.168.201.2 255.255.255.0
CE-B1(config-if-Vlan2)#exit
CE-B1(config)#interface vlan 1
CE-B1(config-if-Vlan2)#ip address 20.1.1.1 255.255.255.0
CE-B1(config-if-Vlan2)#exit
CE-B1(config)#router bgp 60201
CE-B1(config-router)#neighbor 192.168.201.1 remote-as 100
CE-B1(config-router)#exit
```

Configuration of CE-B2:

```
CE-B2#config
CE-B2(config)#interface vlan 2
CE-B2(config-if-Vlan2)#ip address 192.168.202.2 255.255.255.0
CE-B2(config-if-Vlan2)#exit
CE-B2(config)#interface vlan 1
CE-B2(config-if-Vlan2)#ip address 20.1.2.1 255.255.255.0
CE-B2(config-if-Vlan2)#exit
CE-B2(config)#router bgp 60202
CE-B2(config-router)#neighbor 192.168.202.1 remote-as 100
CE-B2(config-router)#exit
```

Configuration of PE1:

```
PE1#config
PE1(config)#ip vrf VRF-A
PE1(config-vrf)#rd 100:10
PE1(config-vrf)#route-target both 100:10
```

```
PE1(config-vrf)#exit
PE1(config)#ip vrf VRF-B
PE1(config-vrf)#rd 100:20
PE1(config-vrf)#route-target both 100:20
PE1(config-vrf)#exit
PE1(config)#interface vlan 1
PE1(config-if-Vlan1)#ip vrf forwarding VRF-A
PE1(config-if-Vlan1)#ip address 192.168.101.1 255.255.255.0
PE1(config-if-Vlan1)#exit
PE1(config)#interface vlan 2
PE1(config-if-Vlan2)#ip vrf forwarding VRF-B
PE1(config-if-Vlan2)#ip address 192.168.201.1 255.255.255.0
PE1(config-if-Vlan2)#exit
PE1(config)#interface vlan 3
PE1(config-if-Vlan3)#ip address 202.200.1.2 255.255.255.0
PE1(config-if-Vlan3)#label-switching
PE1(config-if-Vlan3)#exit
PE1(config)#interface loopback 1
PE1(Config-if-Loopback1)# ip address 200.200.1.1 255.255.255.255
PE1(config-if-Vlan3)#exit
PE1(config)#router bgp 100
PE1(config-router)#neighbor 200.200.1.2 remote-as 100
PE1(config-router)#neighbor 200.200.1.2 update-source 200.200.1.1
PE1(config-router)#address-family vpnv4 unicast
PE1(config-router-af)#neighbor 200.200.1.2 activate
PE1(config-router-af)#exit-address-family
PE1(config-router)#address-family ipv4 vrf VRF-A
PE1(config-router-af)# neighbor 192.168.101.2 remote-as 60101
PE1(config-router-af)#exit-address-family
PE1(config-router)#address-family ipv4 vrf VRF-B
PE1(config-router-af)# neighbor 192.168.201.2 remote-as 60201
PE1(config-router-af)#exit-address-family
```

Configuration of PE2:

```
PE2#config
PE2(config)#ip vrf VRF-A
PE2(config-vrf)#rd 100:10
PE2(config-vrf)#route-target both 100:10
PE2(config-vrf)#exit
PE2(config)#ip vrf VRF-B
PE2(config-vrf)#rd 100:20
PE2(config-vrf)#route-target both 100:20
PE2(config-vrf)#exit
PE2(config)#interface vlan 1
PE2(config-if-Vlan1)#ip vrf forwarding VRF-A
PE2(config-if-Vlan1)#ip address 192.168.102.1 255.255.255.0
PE2(config-if-Vlan1)#exit
PE2(config)#interface vlan 2
PE2(config-if-Vlan2)#ip vrf forwarding VRF-B
PE2(config-if-Vlan2)#ip address 192.168.202.1 255.255.255.0
PE2(config-if-Vlan2)#exit
PE2(config)#interface vlan 3
PE2(config-if-Vlan3)#ip address 202.200.2.2 255.255.255.0
PE2(config-if-Vlan3)#label-switching
PE2(config-if-Vlan3)#exit
PE2(config)#interface loopback 1
PE2(Config-if-Loopback1)# ip address 200.200.1.2 255.255.255.255
PE2(config-if-Vlan3)#exit
PE2(config)#router bgp 100
PE2(config-router)#neighbor 200.200.1.1 remote-as 100
PE2(config-router)#address-family vpnv4 unicast
PE2(config-router-af)#neighbor 200.200.1.1 activate
```

```
PE2(config-router-af)#exit-address-family
PE2(config-router)#address-family ipv4 vrf VRF-A
PE2(config-router-af)# neighbor 192.168.102.2 remote-as 60102
PE2(config-router-af)#exit-address-family
PE2(config-router)#address-family ipv4 vrf VRF-B
PE2(config-router-af)# neighbor 192.168.202.2 remote-as 60202
PE2(config-router-af)#exit-address-family
```

The sample configurations above are the most typical. To enable communication between VRFs, modify the route targets. If the BGP AS numbers at both ends are the same, configure the **neighbor *<ip-addr>* as-override** command to avoid AS number conflicts.

Only the BGP-related configuration is listed above. To run LDP in the public network, refer to the LDP configuration example.

5.5.4 BGP Troubleshooting

When BGP is configured and deployed, physical connection or configuration errors may prevent BGP from working. Therefore, pay attention to the following points:

- ☞ First, ensure that the physical connections are correct.
- ☞ Second, ensure that the interface and link protocols are up (use the **show interface** command).
- ☞ Then, start BGP (use the **router bgp** command) and configure the IBGP and EBGP neighbors (use the **neighbor remote-as** command).
- ☞ Note that BGP itself cannot discover routes; it must import other routes to create BGP routes, and only imported routes can be advertised to IBGP and EBGP neighbors. The imported routes include direct routes, static routes, and IGP routes (RIP and OSPF). Use the **network** and **redistribute (BGP)** commands to import routes.
- ☞ Pay attention to the differences in behavior between IBGP and EBGP.
- ☞ After the configuration is complete, use the **show ip bgp summary** command to check the connection status of all neighbors, and use the **show ip bgp** command to view the BGP routing table.
- ☞ If the BGP routing problem still cannot be solved, use debug commands such as **debug ip bgp** packet/events, record the debug output for 3 minutes, and send it to EnGenius technical support.

5.6 IPv4 Black Hole Routing

5.6.1 Introduction to Black Hole Routing

Black hole routing is a special kind of static routing that drops all datagrams matching the route.

5.6.2 IPv4 Black Hole Routing Configuration Task

1. Configure IPv4 black hole routing

1. Configure IPv4 Black Hole Routing

Command	Explanation
Global Mode	
ip route {*<ip-prefix>* *<mask>*|*<ip-prefix>*/*<prefix-length>*} null0 [*<distance>*] **no ip route {*<ip-prefix>* *<mask>*|*<ip-prefix>*/*<prefix-length>*} null0**	Configures a static black hole route. The no command removes the specified black hole route.

5.6.3 Black Hole Routing Configuration Examples

Example 1: IPv4 black hole routing.

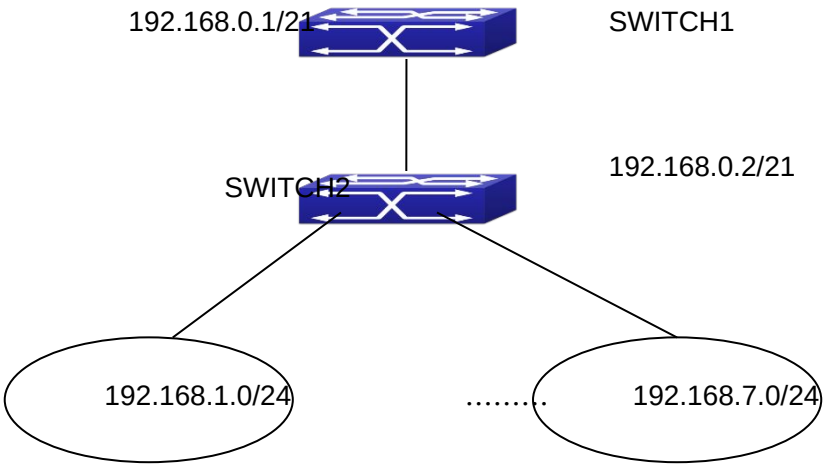


Figure 5-9 IPv4 Black Hole Routing Configuration Example

As shown in the figure, eight interfaces on Switch 2 are configured as Layer 3 VLAN interfaces for access. The network addresses are 192.168.1.0/24 ~ 192.268.7.0/24. A default route is configured on Switch 2 pointing to Switch 1, and a return route to network 192.168.0.0/21 is configured on Switch 1 pointing to Switch 2. Normally, this configuration works well. However, suppose one of the Layer 3 interfaces on Switch 2 goes down, for example, the interface for 192.168.1.0/24. When datagrams for this network arrive at VLAN1 on Switch 2, no route matches them, so the switch forwards them back to Switch 1 according to the default route. Switch 1 then forwards them back to Switch 2, creating a routing loop. To solve this problem, configure a black hole route on Switch 2:

```
ip route 192.168.0.0/21 null0 50
```

Switch 2 then drops the datagrams from interface VLAN1 that match the black hole route, and the routing loop is prevented.

The configuration steps are as follows:

```
Switch#config
Switch(config)#ip route 192.168.0.0/21 null0 50
```

5.6.4 Black Hole Routing Troubleshooting

When black hole routing is configured, it may not work because of problems such as an incorrect network address mask or an incorrect administrative distance. Pay attention to the following:

- ☞ IPv6 must be enabled before IPv6 black hole routing can work.
- ☞ It is recommended that the network mask of a black hole route be longer than that of normal routes, so that the black hole route does not interfere with other routes.
- ☞ When the network mask of a black hole route is the same as that of another route, it is recommended that the black hole route be given a lower distance.

If the problem cannot be solved with the methods above, run the **show ip route distance**, **show ip route fib**, and **show l3** commands, copy the output, and send it to EnGenius technical support.

5.7 GRE

5.7.1 Introduction to GRE Tunnels

GRE (General Routing-protocol Encapsulation) was submitted to the IETF by Cisco and NetSmiths in 1994 as RFC1701 and RFC1702. Today, most network vendors' devices support the GRE tunnel protocol. GRE specifies how to encapsulate one network protocol in another. A GRE tunnel is defined by the source and destination IP addresses of its two ends. It allows IP, IPX, and AppleTalk packets to be encapsulated in IP packets, and it supports all routing protocols (such as RIPv2 and OSPF). With GRE, a public IP network can be used to connect IPX or AppleTalk networks, private (reserved) addresses can be connected across the Internet, and the IP addresses of a corporate network can be hidden from the public network. GRE only encapsulates packets and provides no encryption against interception or attacks. Therefore, in practice it is usually used together with IPsec, which encrypts the data to provide strong security.

GRE is mainly used for encapsulating internal protocols and private addresses in enterprise networks. Because almost all enterprise networks use TCP/IP, there is little need for internal protocol encapsulation, and enterprises use GRE mainly to encapsulate private addresses. On the switch, GRE is mainly used for network protocol transition (including IPv6 over IPv4 and IPv4 over IPv6).

The implementation follows RFC1701, RFC1702, and RFC2784.

5.7.2 Basic GRE Tunnel Configuration

GRE tunnel configuration task list:

1. Configure the tunnel mode

1) Configure the tunnel mode as a GREv4 tunnel

2. Configure the source and destination addresses of the GRE tunnel

1) Configure an IPv4 address as the source address of the GRE tunnel

2) Configure an IPv4 address as the destination address of the GRE tunnel

3. Configure the GRE tunnel interface address

1) Configure the IPv4 address of the GRE tunnel interface

2) Configure the IPv6 address of the GRE tunnel interface

4. Configure a GRE tunnel as the egress interface of a static route

1) Configure a GRE tunnel as the egress interface of an IPv4 static route

2) Configure a GRE tunnel as the egress interface of an IPv6 static route

1. Configure the tunnel mode

Command	Explanation
Tunnel Interface Mode	
tunnel mode gre ip **no tunnel mode**	Configures the tunnel mode as a GREv4 tunnel. After a packet is encapsulated with GRE, it carries an IPv4 header and traverses the IPv4 network.

2. Configure the source and destination addresses of the GRE tunnel

Command	Explanation
Tunnel Interface Mode	
tunnel source *<ipv4-address>* **no tunnel source**	Configures an IPv4 address as the source address of the GRE tunnel.

tunnel destination *<ipv4-address>* **no tunnel destination**	Configures an IPv4 address as the destination address of the GRE tunnel.
Command	Explanation
Tunnel Interface Mode	
tunnel source {*<ipv6-address>* | *<ipv4-address>*} **no tunnel source**	Configures an IPv6 or IPv4 address as the source address of the GRE tunnel.
tunnel destination {*<ipv6-address>* | *<ipv4-address>*} **no tunnel destination**	Configures an IPv6 or IPv4 address as the destination address of the GRE tunnel.

3. Configure the GRE tunnel interface address

Command	Explanation
Tunnel Interface Mode	
ip address <*ipv4-address*> <*mask*> no ip address <*ipv4-address*> <*mask*>	Configures the IPv4 address of the GRE tunnel interface.
ipv6 address <*ipv6-address/prefix*> no ipv6 address <*ipv6-address/prefix*>	Configures the IPv6 address of the GRE tunnel interface.

4. Configure a GRE tunnel as the egress interface of a static route

Command	Explanation
Global Mode	
ip route <*ipv4-address/mask*> tunnel <*ID*> no ip route <*ipv4-address/mask*> tunnel <*ID*>	Configures a GRE tunnel as the egress interface of an IPv4 static route.
ipv6 route <*ipv6-address/prefix*> tunnel <*ID*> no ipv6 route <*ipv6-address/prefix*> tunnel <*ID*>	Configures a GRE tunnel as the egress interface of an IPv6 static route.

5.7.3 GRE Tunnel Example

Typical GRE Tunnel Example:

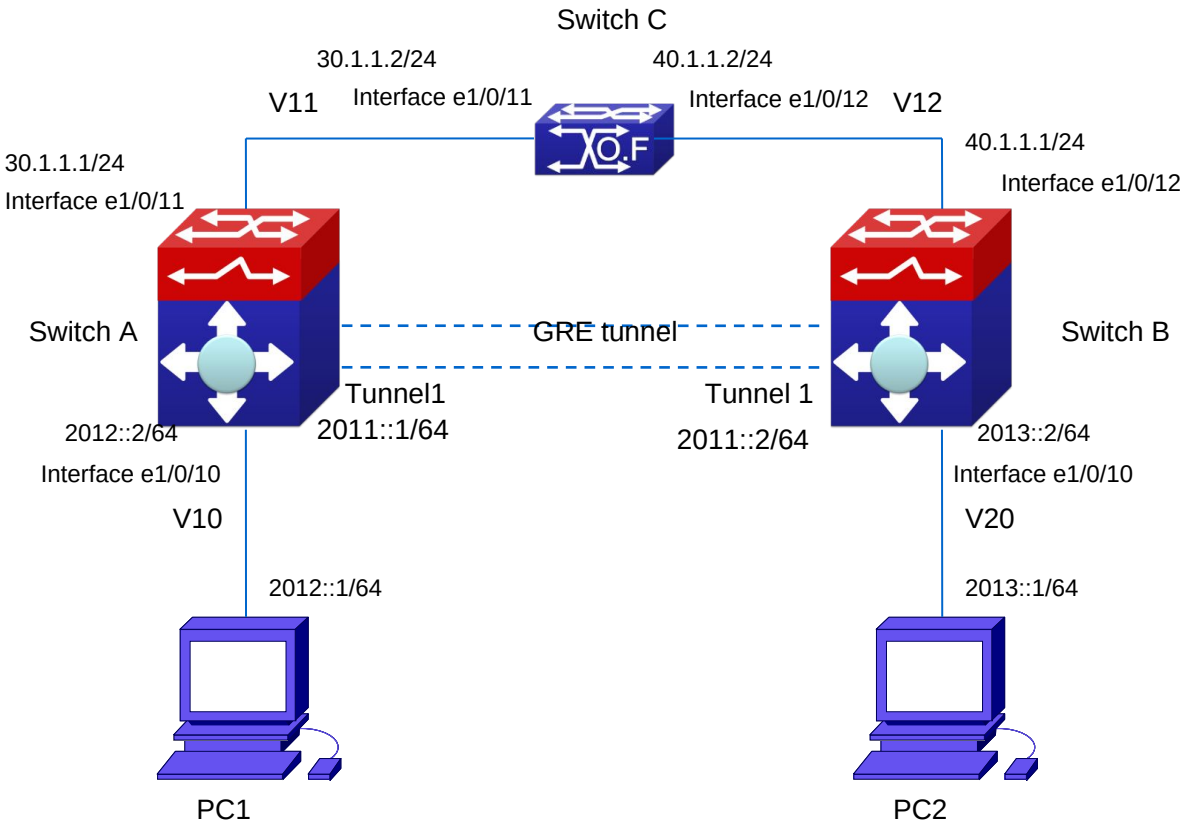


Figure 5-15 Typical IPv6 over IPv4 GRE Tunnel Network

Configuration summary:

- ☞ Configure the IPv4 network and ensure IPv4 connectivity.
- ☞ Configure the tunnel interface and the interfaces connected to the PCs.
- ☞ Configure the tunnel parameters and enable the tunnel interface.
- ☞ Enable OSPF to forward data between PC1 and PC2 through the tunnel.

Configuration steps

Note: The topology in this chapter may differ from your actual environment. To ensure that the configuration takes effect, make sure that the current configuration of the device does not conflict with the following configuration.

(1) Configuration of device A

1. Configuration steps

- ☞ Create interface VLAN 11 and configure its address.

```
SwitchA(config)#vlan 11
SwitchA(config-vlan11)#switchport interface ethernet 1/1/11
SwitchA(config-vlan11)#exit
SwitchA(config)#interface vlan 11
SwitchA(config-if-vlan11)#ip address 30.1.1.1/24
```

- ☞ Configure an IPv4 static route to Switch B through interface VLAN 11.

```
SwitchA(config)#ip route 40.1.1.1/24 30.1.1.2
```

- ☞ Configure the tunnel interface: source, destination, and type. After the tunnel is enabled, its source and destination addresses cannot be modified, unless the source is a Layer 3 interface.

```
SwitchA(config)#interface tunnel 1
```

```
SwitchA(config-if-tunnel1)# tunnel source 30.1.1.1
SwitchA(config-if-tunnel1)# tunnel destination 40.1.1.1
SwitchA(config-if-tunnel1)# tunnel mode gre ip
SwitchA#show gre tunnel
name         mode                source                  destination
Tunnel1      gre ip                  30.1.1.1                40.1.1.1
```

The GRE tunnel is configured successfully.

- ☞ Configure the IPv6 address of the tunnel interface. Only one interface address can be configured for a tunnel interface; this limitation also applies to other tunnels, such as configured, 6to4, and ISATAP tunnels.

Note: The tunnel must remain in the active state when an IPv4 address is configured; configuring an IPv6 address is different.

```
SwitchA (config-if-tunnel1)#ipv6 address 2011::1/64
```

- ☞ Create interface VLAN 10 and configure its address.

```
SwitchA(config)#vlan 10
SwitchA(config-vlan10)#switchport interface ethernet 1/1/10
SwitchA(config-vlan10)#exit
SwitchA(config)#interface vlan 10
SwitchA(config-if-vlan10)# ipv6 address 2012::2/64
SwitchA(config-if-vlan10)#exit
```

- ☞ Configure OSPF.

```
SwitchA(config)#router ospf
SwitchA(config-router)#router-id 1.1.1.1
SwitchA(config-router)#network 30.1.1.1/24 area 0
SwitchA(config-router)#exit
```

(2) Configuration of device B

1. Configuration steps

- ☞ Create interface VLAN 12 and configure its address.

```
SwitchA(config)#vlan 12
SwitchA(config-vlan12)#switchport interface ethernet 1/1/12
SwitchA(config-vlan12)#exit
SwitchA(config)#interface vlan 12
SwitchA(config-if-vlan12)#ip address 40.1.1.1/24
SwitchA(config-if-vlan12)#exit
SwitchA(config)#
```

- ☞ Configure an IPv4 static route to Switch A through interface VLAN 12.

```
SwitchA(config)#ip route 30.1.1.1/24 40.1.1.2
```

- ☞ Configure the tunnel interface: source, destination, and type.

```
SwitchA(config)#interface tunnel 1
SwitchA(config-if-tunnel1)# tunnel source 40.1.1.1
SwitchA(config-if-tunnel1)# tunnel destination 30.1.1.1
SwitchA(config-if-tunnel1)# tunnel mode gre ip
SwitchA#show gre tunnel
name         mode          source                  destination
Tunnel1       gre ip         40.1.1.1                  30.1.1.1
```

The GRE tunnel is configured successfully.

- ☞ Configure the IPv6 address of the tunnel interface. The interface address must be configured to run OSPF.

```
SwitchA (config-if-tunnel1)#ipv6 address 2011::2/64
```

- ☞ Create interface VLAN 20 and configure its address.

```
SwitchA(config)#vlan 20
```

```
SwitchA(config-vlan20)#switchport interface ethernet 1/1/10
SwitchA(config-vlan20)#exit
SwitchA(config)#interface vlan 20
SwitchA(config-if-vlan20)# ipv6 address 2013::2/64
SwitchA(config-if-vlan20)#exit
SwitchA(config)#
```

☞ Configure OSPF.

```
SwitchA(config)#router ospf
SwitchA(config-router)#router-id 1.1.1.2
SwitchA(config-router)#network 40.1.1.0/24 area 0
SwitchA(config-router)#exit
SwitchA(config)#
```

(3) Configuration of device C

1. Configuration steps

☞ Create interface VLAN 11 and configure its address.

```
SwitchA(config)#vlan 11
SwitchA(config-vlan11)#switchport interface ethernet 1/1/11
SwitchA(config-vlan11)#exit
SwitchA(config)#interface vlan 11
SwitchA(config-if-vlan11)#ip address 30.1.1.2/24
```

☞ Create interface VLAN 12 and configure its address.

```
SwitchA(config)#vlan 12
SwitchA(config-vlan12)#switchport interface ethernet 1/1/12
SwitchA(config-vlan12)#exit
SwitchA(config)#interface vlan 12
SwitchA(config-if-vlan12)#ip address 40.1.1.2/24
SwitchA(config-if-vlan12)#exit
```

(4) PC configuration

☞ Configure the IP address and default gateway of each PC.

PC1: IP address 2012::1/64, default gateway 2012::2

PC2: IP address 2013::1/64, default gateway 2013::2

5.7.4 GRE Tunnel Example Using a Loopback Group

Introduction to loopback groups

When cards of different types are installed in a device, the loopback function redirects traffic between the cards. For example, if both cards that support GRE tunnels and cards that do not are installed, the GRE tunnel data received by a card that does not support GRE tunnels is redirected to a card that does for processing. A loopback group uses the ports of one or more cards, and these ports must have no other configuration. To increase the redirection bandwidth between cards, add multiple ports to a loopback group for load balancing.

GRE tunnel example using a loopback group

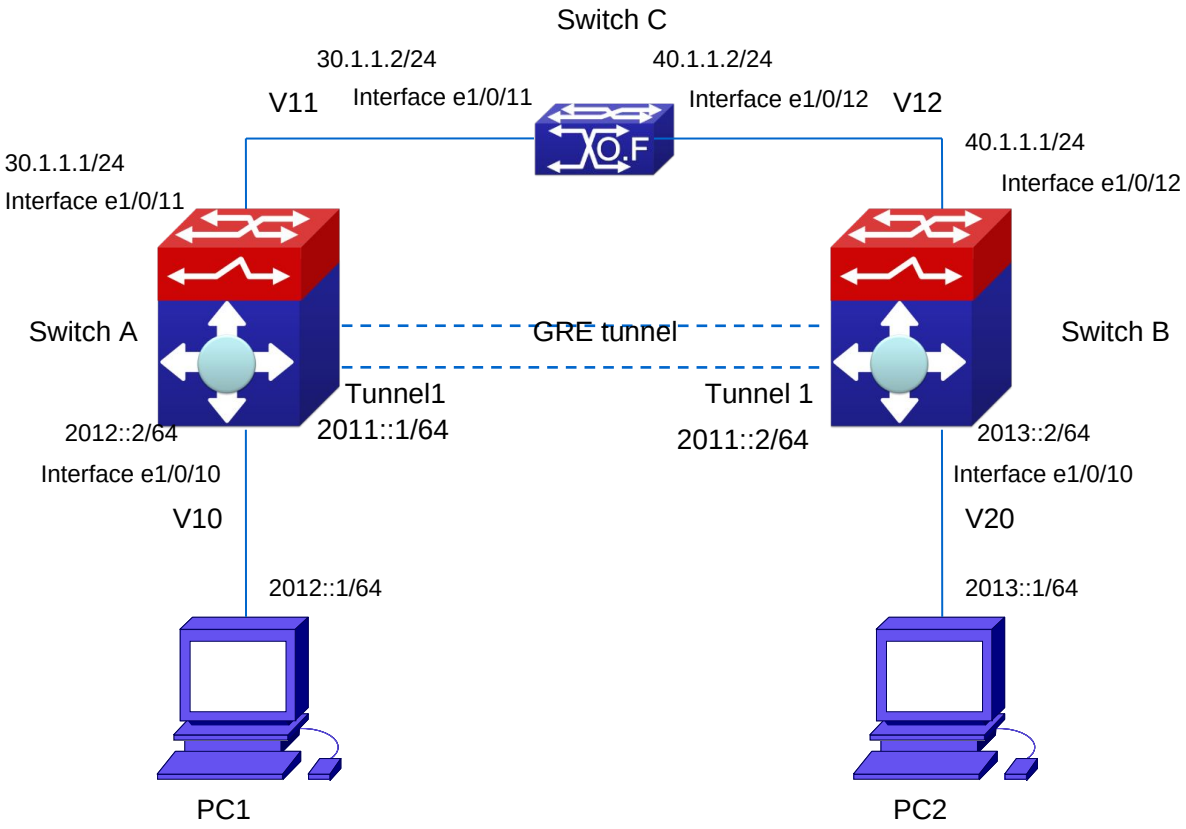


Figure 5-16 GRE Tunnel Using a Loopback Group Topology

Loopback group topology description

There is an IPv4 network between SwitchA and SwitchB, and PC1 and PC2 are in IPv6 networks, so PC1 must communicate with PC2 through a GRE tunnel across the IPv4 network between SwitchA and SwitchB. Cards of different types may be installed in SwitchA: card 1 does not support GRE tunnels and card 3 does, so the data received by card 1 must be redirected to card 3 through the loopback group function.

Configuration summary:

- ☞ Configure the IPv4 network and ensure IPv4 connectivity.
- ☞ Configure the tunnel interface and the interfaces connected to the PCs.
- ☞ Configure the tunnel parameters and enable the tunnel interface.
- ☞ Configure a loopback group, add port 1/1/12 of card 3 to the loopback group, and bind the tunnel to the loopback group.
- ☞ Enable OSPF to forward data between PC1 and PC2 through the tunnel.

Configuration steps

Note: The topology in this chapter may differ from your actual environment. To ensure that the configuration takes effect, make sure that the current configuration of the device does not conflict with the following configuration.

(1) Configuration of device A

1. Configuration steps

- ☞ Create interface VLAN 11 and configure its address.

```
SwitchA(config)#vlan 11
SwitchA(config-vlan11)#switchport interface ethernet 1/1/11
SwitchA(config-vlan11)#exit
```

```
SwitchA(config)#interface vlan 11
SwitchA(config-if-vlan11)#ip address 30.1.1.1/24
```

☞ Configure an IPv4 static route to Switch B through interface VLAN 11.

```
SwitchA(config)#ip route 40.1.1.1/24 30.1.1.2
```

☞ Configure the tunnel interface: source, destination, and type.

```
SwitchA(config)#interface tunnel 1
SwitchA(config-if-tunnel1)# tunnel source 30.1.1.1
SwitchA(config-if-tunnel1)# tunnel destination 40.1.1.1
SwitchA(config-if-tunnel1)# tunnel mode gre ip
SwitchA#show gre tunnel
name         mode                source                  destination
Tunnel1      gre ipv6               30.1.1.1                 40.1.1.1
```

The GRE tunnel is configured successfully.

☞ Configure the IPv6 address of the tunnel interface. The interface address must be configured to run OSPF.

```
SwitchA (config-if-tunnel1)#ipv6 address 2011::1/64
```

☞ Create interface VLAN 10 and configure its address.

```
SwitchA(config)#vlan 10
SwitchA(config-vlan10)#switchport interface ethernet 1/1/10
SwitchA(config-vlan10)#exit
SwitchA(config)#interface vlan 10
SwitchA(config-if-vlan10)# ipv6 address 2012::2/64
SwitchA(config-if-vlan10)#exit
```

☞ Configure the loopback group and bind the tunnel to it.

```
SwitchA (config)#loopback-group 1
SwitchA (config-if-ethernet1/1/12)#loopback-group 1
SwitchA (config-if-tunnel1)# loopback-group 1
```

☞ Configure OSPF.

```
SwitchA(config)#router ospf
SwitchA(config-router)#router-id 1.1.1.1
SwitchA(config-router)#network 30.1.1.0/24 area 0
SwitchA(config-router)#exit
```

(2) Configuration of device B

1. Configuration steps

☞ Create interface VLAN 12 and configure its address.

```
SwitchA(config)#vlan 12
SwitchA(config-vlan12)#switchport interface ethernet 1/1/12
SwitchA(config-vlan12)#exit
SwitchA(config)#interface vlan 12
SwitchA(config-if-vlan11)#ip address 30.1.1.2/24
SwitchA(config-if-vlan12)#exit
SwitchA(config)#
```

☞ Configure an IPv4 static route to Switch A through interface VLAN 12.

```
SwitchA(config)#ip route 30.1.1.1/24 40.1.1.2
```

☞ Configure the tunnel interface: source, destination, and type.

```
SwitchA(config)#interface tunnel 1
SwitchA(config-if-tunnel1)# tunnel source 40.1.1.1
SwitchA(config-if-tunnel1)# tunnel destination 30.1.1.1
SwitchA(config-if-tunnel1)# tunnel mode gre ip
SwitchA#show gre tunnel
name         mode          source                  destination
Tunnel1       gre ipv6       40.1.1.1                 30.1.1.1
```

The GRE tunnel is configured successfully.

☞ Configure the IPv6 address of the tunnel interface. The interface address must be configured to run OSPF.

```
SwitchA (config-if-tunnel1)#ipv6 address 2011::2/64
```

☞ Create interface VLAN 20 and configure its address.

```
SwitchA(config)#vlan 20
SwitchA(config-vlan20)#switchport interface ethernet 1/1/10
SwitchA(config-vlan20)#exit
SwitchA(config)#interface vlan 20
SwitchA(config-if-vlan20)# ipv6 address 2013::2/64
SwitchA(config-if-vlan20)#exit
SwitchA(config)#
```

☞ Configure OSPF.

```
SwitchA(config)#router ospf
SwitchA(config-router)#router-id 1.1.1.2
SwitchA(config-router)#network 40.1.1.0/24 area 0
SwitchA(config-router)#exit
SwitchA(config)#
```

(3) Configuration of device C

1. Configuration steps

☞ Create interface VLAN 11 and configure its address.

```
SwitchA(config)#vlan 11
SwitchA(config-vlan11)#switchport interface ethernet 1/1/11
SwitchA(config-vlan11)#exit
SwitchA(config)#interface vlan 11
SwitchA(config-if-vlan11)#ip address 30.1.1.2/24
```

☞ Create interface VLAN 12 and configure its address.

```
SwitchA(config)#vlan 12
SwitchA(config-vlan12)#switchport interface ethernet 1/1/12
SwitchA(config-vlan12)#exit
SwitchA(config)#interface vlan 12
SwitchA(config-if-vlan12)#ip address 40.1.1.2/24
SwitchA(config-if-vlan12)#exit
```

(4) PC configuration

☞ Configure the IP address and default gateway of each PC.

PC1: IP address 2012::1/64, default gateway 2012::2

PC2: IP address 2013::1/64, default gateway 2013::2

5.7.5 GRE Tunnel Troubleshooting

If a problem occurs when you use a GRE tunnel, check whether it is caused by any of the following:

☞ The configuration: whether the source and destination addresses of the tunnel and the tunnel mode (tunnel mode gre {ip | ipv6}) are configured correctly.

☞ The static route that uses the GRE tunnel interface as the next-hop interface.

☞ The connection between the switches. Use debug gre {packet | event | all} to check whether the switch receives and processes GRE packets correctly.

5.8 ECMP

5.8.1 Introduction to ECMP

ECMP (Equal-Cost Multi-Path routing) applies to networks with multiple different links to the same destination address. With traditional routing, only one link can be used to send packets to the destination; the other links are in the backup or invalid state, and switching between them takes some time in a static routing environment. ECMP, however, can use multiple links in such a network. It not only provides load balancing and increases transmission bandwidth, but also fully backs up the traffic of failed links without delay or packet loss.

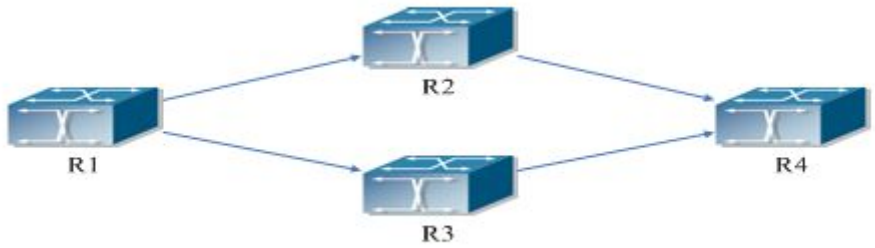


Figure 5-17 ECMP Application Environment

As shown in the figure, there are two paths from R1 to R4: R1-R2-R4 and R1-R3-R4. If the route type and cost are the same, two routes from R1 to R4 with different next hops are formed. If both routes are selected as best routes, they form equal-cost routes.

5.8.2 ECMP Configuration Task List

1. Configure the maximum number of equal-cost routes
2. Configure the load-balance mode for port groups

1. Configure the maximum number of equal-cost routes

Command	Explanation
Global Mode	
maximum-paths <1-8**>** **no maximum-paths**	Configures the maximum number of equal-cost routes.

2. Configure the load-balance mode for port groups

Command	Explanation
Global Mode	
load-balance {dst-src-mac | dst-src-ip | **dst-src-mac-ip }**	Sets the load-balance mode of the switch. The setting applies to both port groups and ECMP.

5.8.3 ECMP Typical Example

Figure 5-18 ECMP Application Environment

As shown in the figure, R1 connects to R2 and R3 with interface addresses 100.1.1.1/24 and 100.1.2.1/24. R2 and R3 connect to R1 with interface addresses 100.1.1.2/24 and 100.1.2.2/24. R4 connects to R2 and R3 with interface addresses 100.2.1.1/24 and 100.2.2.1/24. R2 and R3 connect to R4 with interface addresses 100.2.1.2/24 and 100.2.2.2/24. The loopback address of R4 is 5.5.5.5/32.

5.8.3.1 ECMP with Static Routes

```
R1(config)#ip route 5.5.5.5/32 100.1.1.2
R1(config)#ip route 5.5.5.5/32 100.1.2.2
```

Run show ip route on R1. The following is displayed:

```
R1(config)#show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter
area
       * - candidate default
C       1.1.1.1/32 is directly connected, Loopback1  tag:0
S       5.5.5.5/32 [1/0] via 100.1.1.2, Vlan100  tag:0
                   [1/0] via 100.1.2.2, Vlan200  tag:0
C       100.1.1.0/24 is directly connected, Vlan100  tag:0
C       100.1.2.0/24 is directly connected, Vlan200 tag:0
C       127.0.0.0/8 is directly connected, Loopback  tag:0
Total routes are : 6 item(s)
```

5.8.3.2 ECMP with OSPF

R1 configuration:

```
R1(config)#interface Vlan100
R1(Config-if-Vlan100)# ip address 100.1.1.1 255.255.255.0
R1(config)#interface Vlan200
R1(Config-if-Vlan200)# ip address 100.1.2.1 255.255.255.0
R1(config)#interface loopback 1
R1(Config-if-loopback1)# ip address 1.1.1.1 255.255.255.255
R1(config)#router ospf 1
R1(config-router)# ospf router-id 1.1.1.1
R1(config-router)# network 100.1.1.0/24 area 0
R1(config-router)# network 100.1.2.0/24 area 0
```

R2 configuration:

```
R2(config)#interface Vlan100
R2(Config-if-Vlan100)# ip address 100.1.1.2 255.255.255.0
R2(config)#interface Vlan200
R2(Config-if-Vlan200)# ip address 100.2.1.2 255.255.255.0
R2(config)#interface loopback 1
R2(Config-if-loopback1)# ip address 2.2.2.2 255.255.255.255
R2(config)#router ospf 1
R2(config-router)# ospf router-id 2.2.2.2
R2(config-router)# network 100.1.1.0/24 area 0
R2(config-router)# network 100.2.1.0/24 area 0
```

R3 configuration:

```
R3(config)#interface Vlan100
R3(Config-if-Vlan100)# ip address 100.1.2.2 255.255.255.0
R3(config)#interface Vlan200
R3(Config-if-Vlan200)# ip address 100.2.2.2 255.255.255.0
R3(config)#interface loopback 1
R3(Config-if-loopback1)# ip address 3.3.3.3 255.255.255.255
R3(config)#router ospf 1
R3(config-router)# ospf router-id 3.3.3.3
R3(config-router)# network 100.1.2.0/24 area 0
R3(config-router)# network 100.2.2.0/24 area 0
```

R4 configuration:

```
R4(config)#interface Vlan100
R4(Config-if-Vlan100)# ip address 100.2.1.1 255.255.255.0
```

```
R4(config)#interface Vlan200
R4(Config-if-Vlan200)# ip address 100.2.2.1 255.255.255.0
R4(config)#interface loopback 1
R4(Config-if-loopback1)# ip address 5.5.5.5 255.255.255.255
R4(config)#router ospf 1
R4(config-router)# ospf router-id 4.4.4.4
R4(config-router)# network 100.2.1.0/24 area 0
R4(config-router)# network 100.2.2.0/24 area 0
```

Run show ip route on R1. The following is displayed:

```
R1(config)#show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter
area
       * - candidate default
C       1.1.1.1/32 is directly connected, Loopback1  tag:0
O       5.5.5.5/32 [110/3] via 100.1.1.2, Vlan100, 00:00:05  tag:0
                   [110/3] via 100.1.2.2, Vlan200, 00:00:05  tag:0
C       100.1.1.0/24 is directly connected, Vlan100  tag:0
C       100.1.2.0/24 is directly connected, Vlan200  tag:0
O       100.2.1.0/24 [110/2] via 100.1.1.2, Vlan100, 00:02:25  tag:0
O       100.2.2.0/24 [110/2] via 100.1.2.2, Vlan200, 00:02:25  tag:0
C       127.0.0.0/8 is directly connected, Loopback  tag:0
Total routes are : 8 item(s)
```

5.8.4 ECMP Troubleshooting

ECMP may not work properly because of physical connection or configuration errors. Note the following:

- When ECMP is used, set the load-balance mode to dst-src-ip or dst-src-mac-ip to ensure correct load balancing of packets.

5.9 BFD

5.9.1 Introduction to BFD

BFD (Bidirectional Forwarding Detection) provides a mechanism for quickly detecting and monitoring link connectivity in a network. To improve network performance, protocol neighbors must detect communication failures quickly so that communication can be restored through backup paths as soon as possible.

BFD provides a general-purpose, standardized, medium-independent, and protocol-independent fast failure detection mechanism. It uniformly and quickly detects failures of the bidirectional forwarding paths between two network devices for upper-layer protocols, such as routing protocols and Multiprotocol Label Switching (MPLS). BFD establishes a session between two network devices to monitor the bidirectional forwarding paths between them on behalf of upper-layer protocols. BFD has no discovery mechanism; an upper-layer protocol notifies BFD to establish a session. After a session is established, if no BFD control packet is received from the peer within the detection time, BFD notifies the upper-layer protocol of the failure, and the protocol takes appropriate action.

5.9.2 BFD Configuration Task List

1. Configure basic BFD functions
2. Configure BFD for RIP (RIPng)

3. Configure BFD for static routes (IPv6)

4. Configure BFD for VRRP (VRRPv3)

1. Configure basic BFD functions

Command	Explanation
Global Mode	
bfd mode{active | passive} **no bfd mode**	Configures the BFD mode used before a BFD session is established. The default is active mode. The no command restores active mode.
bfd authentication key <*1-255*> text <*WORD*> **no bfd authentication key <*1-255*>**	Configures a key and a plain-text authentication string for BFD. The no command deletes the configured key.
bfd authentication key <*1-255*> md5 <*WORD*> **no bfd authentication key**	Configures a key and an MD5-encrypted authentication string for BFD. The no command deletes the configured key.
Interface Mode	
bfd interval <*value1*> min_rx <*value2*> multiplier <*value3*> **no bfd interval**	Configures the minimum transmission interval of BFD control packets and the session detection multiplier. The no command restores the default detection multiplier.
bfd min-echo-receive-interval <*value*> **no bfd min-echo-receive-interval**	Configures the minimum receive interval for BFD control packets. The no command restores the default value.
bfd echo **no bfd echo**	Enables BFD echo. The no command disables the function.
bfd echo-source-ip <*ipv4-address*> **no bfd echo-source-ip**	Configures the source address of echo packets used to detect link faults. The no command deletes the configured source address of echo packets.
bfd echo-source-ipv6 <*ipv6-address*> **no bfd echo-source-ipv6**	Configures the IPv6 source address of echo packets used to detect link faults. The no command deletes the configured source address of echo packets.
bfd authentication key <*1-255*> **no bfd authentication key**	Enables BFD authentication and configures the key for the interface. The no command disables BFD authentication.

2. Configure BFD for RIP (RIPng)

Command	Explanation
Interface Mode	
rip bfd enable **no rip bfd enable**	Enables BFD for RIP on the specified interface. The no command disables BFD for

	RIP.
ipv6 rip bfd enable **no ipv6 rip bfd enable**	Enables BFD for RIPng on the specified interface. The no command cancels the configuration.

3. Configure BFD for static routes (IPv6)

Command	Explanation
Global Mode	
ip route {vrf <name> <ipv4-address> | <ipv4-address>} mask <nexthop> bfd **no ip route {vrf <name> <ipv4-address> | <ipv4-address>} mask <nexthop> bfd**	Configures BFD for the static route. The no command cancels the configuration.
ipv6 route {vrf <name> <ipv6-address> | <ipv6-address>} prefix <nexthop> bfd **no ipv6 route {vrf <name> <ipv6-address> | <ipv6-address>} prefix <nexthop> bfd**	Configures BFD for the static IPv6 route. The no command cancels the configuration.

4. Configure BFD for VRRP (VRRPv3)

Command	Explanation
VRRP (v3) Group Configuration Mode	
bfd enable **no bfd enable**	Enables BFD for VRRP (VRRPv3) and enables BFD detection on this group. The no command disables the function.

5.9.3 BFD Examples

5.9.3.1 Example: Linking BFD with Static Routes

Example:

Configure a static route to 14.1.1.0/24 on Switch A and a static route to 15.1.1.0/24 on Switch B, and enable BFD detection on both switches. When the link between Switch A and Switch B fails, BFD detects the failure immediately.

Figure 5-10

```
Configuration procedure:
Switch A:
Switch#config
Switch(config)#interface vlan 12
Switch(config-if-vlan12)#ip address 12.1.1.1 255.255.255.0
Switch(config)#interface vlan 15
Switch(config-if-vlan15)#ip address 15.1.1.1 255.255.255.0
Switch(config)#ip route 14.1.1.0 255.255.255.0 12.1.1.2 bfd
Switch B:
Switch#config
Switch(config)#interface vlan 12
Switch(config-if-vlan12)#ip address 12.1.1.2 255.255.255.0
Switch(config)#interface vlan 14
Switch(config-if-vlan15)#ip address 14.1.1.1 255.255.255.0
Switch(config)#ip route 15.1.1.0 255.255.255.0 12.1.1.1 bfd
```

When the link between Switch B and the Layer 2 switch fails, Switch A immediately detects the change on Switch B, and the static route becomes inactive.

5.9.3.2 Example: Linking BFD with RIP Routes

Example:

Switch A and Switch B are connected and run RIP, and BFD is enabled on both. When the link between Switch A and Switch B fails, BFD detects the failure immediately.

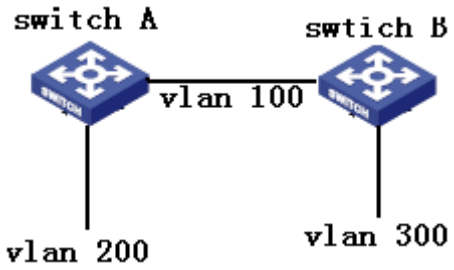


Figure 5-11

```
Configuration procedure:
Switch A:
Switch#config
Switch(config)#bfd mode active
Switch(config)#interface vlan 100
Switch(config-if-vlan100)#ip address 10.1.1.1 255.255.255.0
Switch(config)#interface vlan 200
Switch(config-if-vlan200)#ip address 20.1.1.1 255.255.255.0
Switch(config)#router rip
Switch (config-router)#network vlan 100
Switch (config-router)#network vlan 200
Switch(config)#interface vlan 100
Switch(config-if-vlan100) #rip bfd enable
Switch B:
Switch#config
Switch(config)#bfd mode passive
Switch(config)#interface vlan 100
Switch(config-if-vlan100)#ip address 10.1.1.2 255.255.255.0
Switch(config)#interface vlan 300
Switch(config-if-vlan300)#ip address 30.1.1.1 255.255.255.0
Switch(config)#router rip
Switch (config-router)#network vlan 100
Switch (config-router)#network vlan 300
Switch(config)#interface vlan 100
Switch(config-if-vlan100) #rip bfd enable
```

When the link between Switch A and Switch B fails, BFD detects the failure immediately and notifies RIP to delete the learned routes.

5.9.3.3 Example: Linking BFD with VRRP

Example:

When the master fails, the backup cannot become the master until the configured timeout expires. The timeout is generally three to four seconds, so the switchover is slow. To solve this problem, VRRP uses BFD to probe the state of the master. If the master fails, the backup can become the new master within 100 ms.

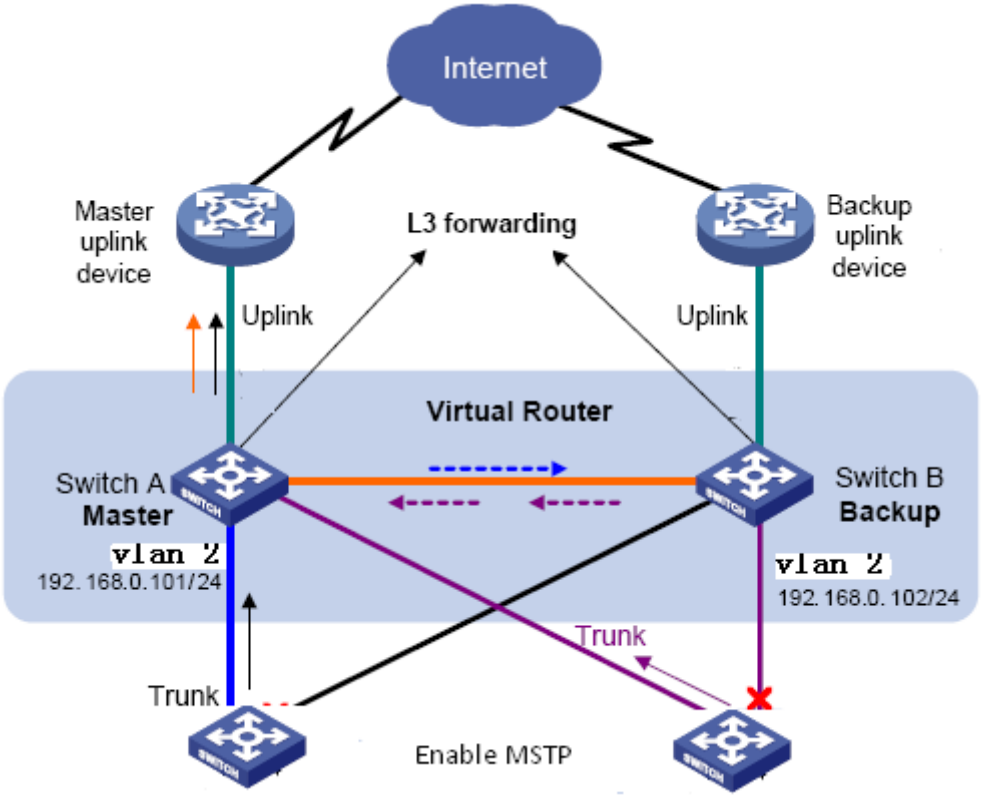


Figure 5-12

```
Configuration procedure:
# Configure Switch A
Switch#config
Switch(config)#bfd mode active
Switch(config)#interface vlan 2
Switch(config-ip-vlan2)#ip address 192.16.0.101 255.255.255.0
Switch(config)#router vrrp 1
Switch(config-router)#virtual-ip 192.168.0.10
Switch(config-router)#interface vlan 1
Switch(config-router)#enable
Switch(config-router)#bfd enable
# Configure Switch B
Switch#config
Switch(config)#bfd mode passive
Switch(config)#interface vlan 2
Switch(config-ip-vlan2)#ip address 192.16.0.102 255.255.255.0
Switch(config)#router vrrp 1
Switch(config-router)#virtual-ip 192.168.0.10
Switch(config-router)#interface vlan 1
Switch(config-router)#enable
Switch(config-router)#bfd enable
```

5.9.4 BFD Troubleshooting

If a problem occurs with the BFD function, check whether it is caused by any of the following:

- ☞ Whether the routing protocol neighbor is established successfully. If no routing protocol neighbor is established, BFD cannot perform detection.
- ☞ For linkage with static routes, whether the configured source IP is correct. If IP connectivity between the two peers fails, BFD cannot perform detection.

☞ For linkage with VRRP, whether the VRRP group is established successfully. If no VRRP group is established, BFD cannot perform detection.

5.10 BGP GR

5.10.1 Introduction to GR

As networks develop, higher availability is required. HA (High Availability) addresses how to keep forwarding packets without affecting traffic when the control plane of a router does not work normally.

Usually, when a router does not work normally, its routing protocol neighbors detect that the neighbor relationship has gone down, and it comes back up soon after. This process is called neighbor flapping. Neighbor flapping causes route flapping, which eventually results in routing black holes or in traffic bypassing the restarted router. As a result, network availability drops quickly.

To achieve high availability, upper-layer routing protocols must support GR (Graceful Restart). GR ensures that packets are processed and forwarded correctly when the control plane fails.

GR reduces route flapping and control-plane resource consumption and improves network stability. This document describes BGP GR, which restarts BGP without affecting the forwarding process and keeps forwarding packets along the correct path.

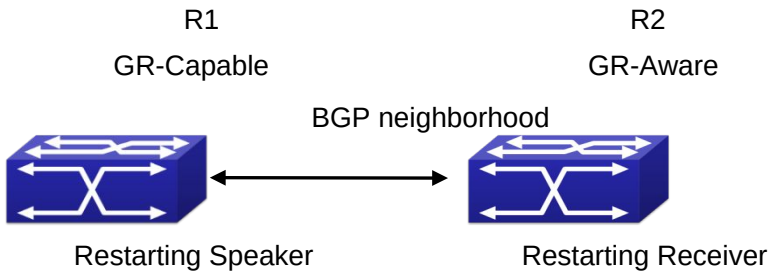


Figure 5-22 Application Environment for GR

GR requires cooperation between a GR-capable router and a GR-aware router. The restarting router is called the Restarting Speaker (or GR-Restarter), and its neighbor is called the Receiving Speaker (or GR-Helper). The Restarting Speaker is a GR-capable router, and the Receiving Speaker is a GR-aware router. Suppose that routers R1 and R2 have established a BGP neighbor relationship, as shown in the figure above. The GR process is as follows:

Restarting Speaker (GR-Restarter):

1. R1 and R2 negotiate the GR capability through OPEN messages when the original BGP neighbor relationship is established.

2. When R1 restarts, its routes are kept on the interface board and continue to guide forwarding.

3. R1 re-establishes a TCP connection with R2 and sets the Restart state to 1 in the BGP OPEN message to indicate that it has restarted. It also informs the neighbor of the restart time (which is less than the hold time in OPEN messages) and of the GR types it supports.

4. After R1 has correctly established the connection with R2, it can receive and process update messages, and it starts the selection deferral timer.

5. R1 delays the calculation of local BGP routes until it receives End-of-RIB from all GR-aware BGP neighbors or until the local selection deferral timer expires.

6. R1 calculates routes and sends route updates, and then sends End-of-RIB to its neighbors.

Receiving Speaker (GR-Helper):

1. R2 negotiates the GR capability with R1 when the original BGP neighbor relationship is established; R1 is a GR-capable router.

2. When R1 restarts, R2 may detect that the TCP connection between R1 and R2 is broken, or it may not detect the previous state before the TCP connection is re-established. If R2 does not detect it, go to step 4; otherwise, go to step 3.

3. R2 keeps the routes sent by R1, marks them as stale, and starts the Restart Timer.

4. R2 closes the old TCP connection and continues to process the new TCP connection. It keeps the routes sent by R1, marks them as stale, and starts the Restart Timer.

5. R2 establishes a new neighbor relationship with the restarted router, deletes the Restart Timer, and starts the Stale Path Timer.

6. If, before the new neighbor relationship is established, the Restart Timer expires, the Restart flag is not 1, or the AFI/SAFI address family has no related support information, R2 clears the retained routes.

7. R2 sends route updates to the restarted router and then sends the End-of-RIB marker.

8. If the Stale Path Timer expires, R2 clears the retained routes.

5.10.2 GR Configuration Task List

1. Configure whether the GR capability is supported

2. Configure whether a specific neighbor supports the GR capability

3. Configure the restart time

4. Configure the restart time for a neighbor

5. Configure the stale path time for BGP GR

6. Configure the selection deferral time for BGP GR

1. Configure whether the GR capability is supported

Command	Description
BGP Router Mode	
bgp graceful-restart **no bgp graceful-restart**	Enables BGP support for GR.

2. Configure whether a specific neighbor supports the GR capability

Command	Description
BGP Unicast Address Family Mode and VRF Address Family Mode	
neighbor (A.B.C.D | X:X::X:X | WORD) capability graceful-restart **no neighbor (A.B.C.D | X:X::X:X | WORD) capability graceful-restart**	Sets a flag for the neighbor so that GR parameters are carried in the OPEN messages sent to it.

3. Configure the restart time

Command	Description
BGP Router Mode	

bgp graceful-restart restart-time <1-3600> **no bgp graceful-restart restart-time <1-3600>**	Configures the BGP GR restart time. The Receiving Speaker starts a timeout timer for the neighbor and uses the restart time as the timeout. The restart time specifies the maximum time from when the Receiving Speaker detects the restart until it receives an OPEN message. If the Receiving Speaker does not receive an OPEN message within this time, it deletes the stale routes saved for the neighbor.

4. Configure the restart time for a neighbor

Command	Description
BGP Unicast Address Family Mode and VRF Address Family Mode	
neighbor (A.B.C.D | X:X::X:X | WORD) restart-time <1-3600> **no neighbor (A.B.C.D | X:X::X:X | WORD) restart-time <1-3600>**	Configures the restart time for the neighbor. The no command restores the default time.

5. Configure the stale path time for BGP GR

Command	Description
BGP Router Mode	
bgp graceful-restart stale-path-time <1-3600> **no bgp graceful-restart stale-path-time <1-3600>**	Configures the BGP GR stale path time. The default is 360s, which is much longer than the restart time and the selection deferral time, because between receiving the OPEN message and receiving EOR, the Receiving Speaker sends the initial route updates and waits until the initial route updates are completely received.

6. Configure the selection deferral time for BGP GR

Command	Description
BGP Router Mode	
bgp selection-deferral-time <1-3600> **no bgp selection-deferral-time <1-3600>**	Specifies the maximum time that the Restarting Speaker waits, from receiving OPEN messages until receiving EOR, before it starts route selection. If the Restarting Speaker does not receive EOR within this time, it starts route selection.

5.10.3 Typical GR Example

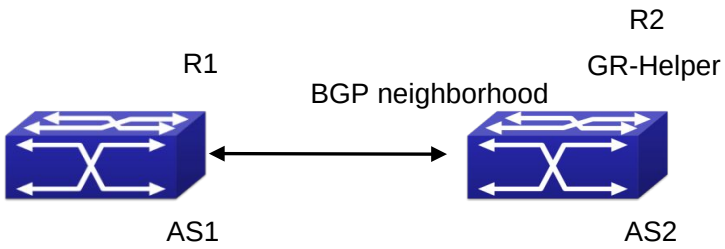


Figure 5-23 Application Environment for GR

As shown in Figure 5-23, R1 and R2 establish a BGP neighbor relationship. When the connection is broken, BGP on R2 enters helper mode, keeps the route entries from R1, and starts the restart-time timer. During this time, if R2 receives OPEN messages from R1 or the timer expires, the routes

marked as stale on R2 are deleted. When the neighbor relationship is re-established, R1 starts the selection timer and waits until R2 sends EOR messages or the timer expires, and then R1 performs route selection. After R2 receives the OPEN messages from R1, it starts the stale route timer; it deletes the timer and the stale routes when it receives EOR from R1 or when the timer expires.

R1: interface VLAN 12, IP address 12.1.1.1

R2: interface VLAN 12, IP address 12.1.1.2

R1 configuration:

```
R1#config
R1(config)#vlan 12
R1(config-vlan12)#int vlan 12
R1(config-if-vlan12)#ip address 12.1.1.1 255.255.255.0
R1(config-if-vlan12)#exit
R1(config)#router bgp 1
R1(config-router)#neighbor 12.1.1.2 remote-as 2
R1(config-router)#neighbor 12.1.1.2 capability graceful-restart
R1(config-router)#bgp selection-deferral-time 120
R1(config-router)#bgp graceful-restart restart-time 60
R1(config-router)#bgp graceful-restart stale-path-time 180
R1(config-router)#exit
```

R2 configuration:

```
R2#config
R2(config)#vlan 12
R2(config-vlan12)#int vlan 12
R2(config-if-vlan12)#ip address 12.1.1.2 255.255.255.0
R2(config-if-vlan12)#exit
R2(config)#router bgp 2
R2(config-router)#neighbor 12.1.1.1 remote-as 1
R2(config-router)#neighbor 12.1.1.1 capability graceful-restart
R2(config-router)#bgp selection-deferral-time 120
R2(config-router)#bgp graceful-restart restart-time 60
R2(config-router)#bgp graceful-restart stale-path-time 180
R2(config-router)#exit
```

5.11 OSPF GR

5.11.1 Introduction to OSPF GR

OSPF Graceful Restart (OSPF GR) keeps data forwarded correctly and prevents interruption of critical service traffic when a routing protocol restarts or a Layer 3 switch performs an active/standby master switchover. It is a high availability technology.

Currently, high-end Layer 3 switches usually separate control from forwarding. The control module that calculates routes is on the master control board, while the data forwarding module is on the line cards. As a result, restarting the master control board does not affect data forwarding on the line cards. Devices that support GR are therefore generally chassis devices with two master control boards.

Because standard OSPF (RFC2328) does not support GR, traffic interruption and route flapping occur when the routing protocol restarts or an active/standby master switchover occurs for any reason. For example, as shown in the figure below, when S1 performs a switchover, the neighbor relationship between S1 and S2 is lost. S2 then sends a Router-LSA to S3 and S4 that does not include the link between S1 and S2. After receiving the LSA, S3 and S4 recalculate routes, and the result does not include the link between S1 and S2. After S1 completes the switchover, it re-establishes the neighbor relationship with S2 and synchronizes the database, which causes S2, S3, and S4 to recalculate routes again. The switchover of S1 therefore causes route flapping, which is unacceptable in some networks with high performance requirements.

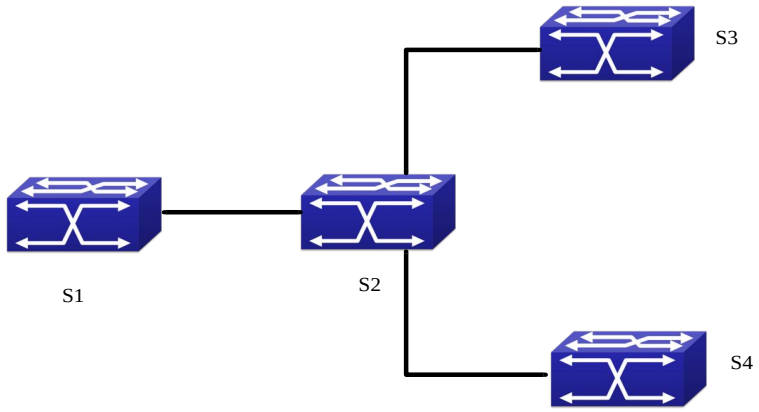


Figure 5-24 Typical Application Scenario

OSPF GR, described in RFC3623, was designed for this situation. Its basic idea is that if the network topology remains stable during the switchover and the Layer 3 switch can keep the same forwarding table, its neighbors can maintain their relationships with it, so that the switch stays on the forwarding path. If S1 and S2 support and enable GR, the line cards of S1 keep forwarding traffic and S2 maintains its relationship with S1. The network topology seen by S3 and S4 does not change, and no route recalculation is needed. This ensures traffic forwarding and avoids route flapping.

According to their roles in the GR process, Layer 3 switches are divided into GR restarters and GR helpers. A GR restarter is a Layer 3 switch that performs an active/standby master switchover or restarts the protocol, and a GR helper is a Layer 3 switch that helps the GR restarter. In the above example, S1 is the GR restarter and S2 is the GR helper.

The advantages of OSPF GR are as follows:

- Increases network reliability
- Reduces the impact of route flapping on the network
- Reduces the impact on traffic and avoids packet loss during the switchover

5.11.2 OSPF GR Configuration

OSPF GR configuration task list:

1. Enable GR for OSPF
2. Configure the grace period for the OSPF GR restarter (optional)
3. Configure the policy for the OSPF GR helper (optional)

1. Enable GR for OSPF

Command	Description
OSPF Router Mode	
capability restart graceful **no capability restart**	Enables GR for the specified OSPF process.

2. Configure the grace period for the OSPF GR restarter (optional)

Command	Description
Global Mode	
ospf graceful-restart grace-period <*integer*> **no ospf restart grace-period**	Configures the grace period for the GR restarter (the switch that performs the switchover or restarts the protocol). The no command restores the default value.

3. Configure the policy for the OSPF GR helper (optional)

Command	Description

Global Mode	
ospf graceful-restart helper max-grace-period *<integer>* **no ospf graceful-restart helper**	Configures the maximum grace period supported by the helper (one of the GR helper policies). The no command deletes all configured helper policies.
ospf graceful-restart helper never **no ospf graceful-restart helper**	Configures the switch so that it cannot become an OSPF GR helper (one of the GR helper policies). The no command deletes all configured helper policies.

5.11.3 OSPF GR Example

Example:

There are four switches, S1 to S4 (each has two master control boards and supports OSPF GR). OSPF is enabled on them to achieve the following:

1. S1 keeps forwarding traffic during the switchover, and S2 to S4 experience no route flapping, so network traffic continues uninterrupted.

2. S1 must complete the switchover and protocol restart within 120s; otherwise, S2 exits GR and recalculates routes.

3. S1 does not act as an OSPF GR helper (S1 does not help S2 perform GR; it recalculates routes when S2 performs a switchover or restarts OSPF).

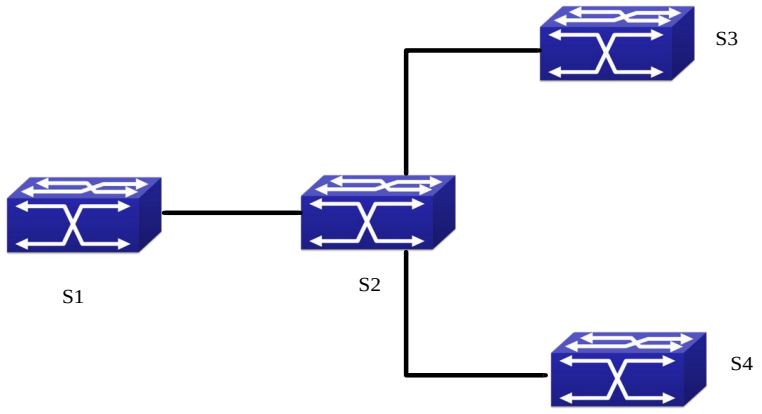


Figure 5-25 Typical Application

Steps: Because OSPF GR is enabled on the switch by default, only the OSPF GR parameters and helper policy need to be configured. (Only the OSPF GR configuration is shown; the topology configuration is omitted.)

```
S1
S1(config)#ospf graceful-restart grace-period 120
S1(config)# ospf graceful-restart helper never
S2
S2(config)# ospf graceful-restart helper max-grace-period 120
```

5.11.4 OSPF GR Troubleshooting

If you have trouble using OSPF GR, check the following:

- Whether the GR restarter supports OSPF GR and has two main control boards; make sure that GR is not disabled.
- Whether the network topology changes during the OSPF GR process. If it changes, the switch may exit GR and restart OSPF.

- Make sure that all neighbors of the GR restarter support GR.
- Do not modify the OSPF configuration during GR.

Chapter 6 Multicast Protocol Configuration

6.1 Multicast

This chapter describes the configuration of IPv4 multicast protocols.

6.1.1 Introduction to Multicast

Different transmission modes can be used when packets (including data, voice, and video) are destined for a small group of users in the network. One option is unicast mode, which sets up a separate data transmission path for each user. Another option is broadcast mode, which sends messages to all users in the network, whether they need them or not. For example, if 200 users in a network want to receive the same packet, the traditional solution is either to send the packet 200 times separately by unicast, so that every user who needs the data receives it, or to send the data to the entire domain by broadcast, so that users who need the data can get it directly from the network. Both modes waste a great deal of valuable bandwidth, and broadcast mode also compromises security and confidentiality.

IP multicast technology solves this problem. The multicast source sends a message only once, the multicast routing protocol builds a routing tree for the multicast data packets, and the packets are replicated and distributed only at branch points, as far down the tree as possible. In this way, packets are delivered accurately and efficiently to every user who needs them.

Note that a multicast source does not need to join a multicast group. It sends data to one or more multicast groups but is not necessarily a receiver of those groups. More than one source can send packets to a multicast group at the same time. Some routers in the network may not support multicast; in that case, a multicast router can encapsulate multicast packets into unicast IP packets in tunnel mode and send them to the next multicast router, which removes the unicast IP header and continues multicast forwarding. This avoids major changes to the network structure. The main advantages of multicast are:

1. Higher efficiency: reduces network traffic and lightens the load on servers and CPUs
2. Better performance: reduces redundant traffic
3. Distributed applications: enables multipoint applications

6.1.2 Multicast Address

The destination address of a multicast message is a class D IP address in the range 224.0.0.0 to 239.255.255.255. A class D address cannot appear in the source IP address field of an IP message. In unicast data transmission, a data packet is routed from the source address to the destination address hop by hop. In an IP multicast environment, however, the destination is a group rather than a single host, identified by a group address. Receivers join a group; once they have joined, the data sent to the group address is delivered to them immediately, and all members of the group receive the data packets. Multicast group membership is dynamic: hosts can join and leave a multicast group at any time.

A multicast group can be permanent or temporary. Some multicast group addresses are officially assigned; these are called permanent multicast groups. A permanent multicast group keeps its IP address fixed, but its membership can change. A permanent multicast group can have any number of members, including zero. IP multicast addresses that are not reserved for permanent multicast groups can be used by temporary multicast groups.

224.0.0.0 to 224.0.0.255 are reserved multicast addresses (permanent group addresses): 224.0.0.0 is reserved but not assigned, and the other addresses are used by routing protocols. 224.0.1.0 to

238.255.255.255 are multicast addresses available to users (temporary group addresses) and are valid throughout the network. 239.0.0.0 to 239.255.255.255 are locally administered multicast addresses, which are valid only within a specific local domain. Commonly used reserved multicast addresses are listed below:

Base address (reserved)

224.0.0.1 All hosts

224.0.0.2 All multicast routers

224.0.0.3 Unassigned

224.0.0.4 DVMRP routers

224.0.0.5 OSPF routers

224.0.0.6 OSPF DRs

224.0.0.7 ST routers

224.0.0.8 ST hosts

224.0.0.9 RIP-2 routers

224.0.0.10 IGRP routers

224.0.0.11 Mobile agents

224.0.0.12 DHCP Server/Relay Agent

224.0.0.13 All PIM routers

224.0.0.14 RSVP Encapsulation

224.0.0.15 All CBT routers

224.0.0.16 Designated SBM

224.0.0.17 All SBMs

224.0.0.18 VRRP

224.0.0.22 IGMP

When Ethernet transmits unicast IP messages, the destination MAC address is the MAC address of the receiver. When multicast packets are transmitted, however, the destination is no longer a specific receiver but a group with changing members, so a multicast MAC address is used. The multicast MAC address corresponds to the multicast IP address. IANA (Internet Assigned Numbers Authority) specifies that the upper 25 bits of a multicast MAC address are 0x01005e and the lower 23 bits are the lower 23 bits of the multicast IP address.

Because only 23 of the lower 28 bits of an IP multicast address are mapped into the MAC address, 32 IP multicast addresses map to the same MAC address.

6.1.3 IP Multicast Packet Transmission

In multicast mode, the source host sends packets to the host group identified by the multicast group address in the destination address field of the IP packet. Unlike unicast, a multicast data packet must be forwarded out of multiple interfaces to reach all receivers, so multicast forwarding is more complex than unicast forwarding.

To ensure that all multicast packets reach the router over the shortest path, the receiving interface of each multicast packet must be checked against the unicast routing table. This check, RPF (Reverse Path Forwarding), is the basis of forwarding for most multicast routing protocols. A multicast router uses the source address of the received packet to query the unicast routing table or an independent multicast routing table and determines whether the ingress interface of the packet is on the shortest path from the receiver to the source address. If a shortest path tree is used, the source address is the address of the source host that sends the multicast data packets; if a shared tree is used, the source address is the address of the root of the shared tree. When a multicast data packet reaches the

router, it is forwarded according to the multicast forwarding entry if the RPF check passes; otherwise, it is discarded.

6.1.4 IP Multicast Application

IP multicast technology effectively solves the problem of one-to-many transmission. It delivers data efficiently from one point to multiple points, saves a great deal of network bandwidth, and reduces network load. The multicast capability of the network also makes it easy to offer new value-added services. In information services such as live online broadcasting, network TV, distance learning, telemedicine, and real-time video/audio conferencing, the following applications can be provided:

1) Multimedia and streaming media applications
2) Data warehousing and financial applications (such as stock quotes)
3) Any one-to-many data distribution application

As multimedia services in IP networks keep growing, multicast has tremendous market potential and will become increasingly widespread.

6.2 PIM-DM

6.2.1 Introduction to PIM-DM

PIM-DM (Protocol Independent Multicast, Dense Mode) is a dense-mode multicast routing protocol suitable for small networks, in which multicast group members are relatively dense.

The PIM-DM working process can be summarized as neighbor discovery, flooding and pruning, and grafting.

1. Neighbor Discovery

When a PIM-DM router starts, it uses Hello messages to discover neighbors. Network nodes running PIM-DM use Hello messages to contact each other. PIM-DM Hello messages are sent periodically.

2. Flooding and Pruning

PIM-DM assumes that all hosts on the network are ready to receive multicast data. When a multicast source S begins sending data to multicast group G, a router that receives the multicast packet first performs an RPF check based on the unicast routing table. If the check passes, the router creates an (S, G) entry and forwards the multicast packet to all downstream PIM-DM nodes on the network (flooding). If the RPF check fails, that is, the multicast packet arrived on an incorrect interface, the packet is discarded. After this process, every node in the PIM-DM multicast domain has an (S, G) entry. If a downstream node has no multicast group members, it sends a Prune message to its upstream node to stop it from forwarding data for this multicast group. On receiving the Prune message, the upstream node removes the corresponding interface from the outgoing interface list of its (S, G) multicast forwarding entry. In this way, an SPT (Shortest Path Tree) rooted at source S is built. Pruning is initiated by the leaf routers.

This process is called flooding and pruning. Each pruned node also maintains a timeout: when the prune state times out, the router restarts the flooding and pruning process. PIM-DM flooding and pruning is performed periodically.

3. RPF Check

Using the RPF check, PIM-DM uses the existing unicast routing table to build a multicast forwarding tree rooted at the data source. When a multicast packet arrives, the router first determines whether it arrived on the correct path. If the arrival interface is the interface toward the multicast source according to unicast routing, the packet is considered to have come from the correct path; otherwise, the packet is discarded as redundant. The unicast routing information used for this check can come

from any unicast routing protocol, such as RIP or OSPF; PIM-DM does not depend on any specific unicast routing protocol.

4. Assert Mechanism

Suppose multicast routers A and B on the same LAN segment each have a receiving route, and both forward to the LAN the multicast data packets sent by multicast source S. The downstream multicast router C then receives two identical copies of each multicast packet. When a router detects this situation, it uses the Assert mechanism to select a single forwarder: Assert packets are exchanged to select the optimal forwarding path. If two or more paths have the same priority and cost, the node with the higher IP address becomes the upstream neighbor of the (S, G) entry and is responsible for forwarding the (S, G) multicast packets.

5. Graft

When a pruned downstream node needs to resume receiving traffic, it sends a Graft packet to notify its upstream node to restore multicast data forwarding.

6.2.2 PIM-DM Configuration Task List

1. Enable PIM-DM (required)
2. Configure static multicast routing entries (optional)
3. Configure additional PIM-DM parameters (optional)
   a) Configure the interval for PIM-DM hello messages
   b) Configure the interval for state-refresh messages
   c) Configure the boundary interfaces
   d) Configure the management boundary
4. Disable PIM-DM

1. Enable PIM-DM

To configure PIM-DM on a Layer 3 switch, enable PIM multicast routing globally, and then enable PIM-DM on specific interfaces.

Command	Explanation
Global Mode	
ip pim multicast-routing **no ip pim multicast-routing**	Enables PIM-DM globally for all interfaces. To make PIM-DM work on a specific interface, the following interface command is also required.

Then enable PIM-DM on the interface.

Command	Explanation
Interface Mode	
ip pim dense-mode	Enables PIM-DM on the specified interface (required).

2. Configure static multicast routing entries

Command	Explanation
Global Mode	
ip mroute *<A.B.C.D>* *<A.B.C.D>* *<ifname>* *<.ifname>* **no ip mroute *<A.B.C.D>* *<A.B.C.D>* [*<ifname>* *<.ifname>*]**	Configures a static multicast routing entry. The no command deletes the specified entry.

3. Configure additional PIM-DM parameters

a) Configure the interval for PIM-DM hello messages

Command	Explanation

Interface Mode	
ip pim hello-interval < *interval*> **no ip pim hello-interval**	Configures the interval for sending PIM-DM hello messages. The no command restores the default interval.

b) Configure the interval for state-refresh messages

Command	Explanation
Interface Mode	
ip pim state-refresh origination-interval **no ip pim state-refresh origination-interval**	Configures the interval for sending PIM-DM state-refresh messages. The no command restores the default value.

c) Configure the boundary interfaces

Command	Explanation
Interface Mode	
ip pim bsr-border **no ip pim bsr-border**	Configures the interface as a PIM-DM boundary. BSR messages are neither sent nor received on a boundary interface, and the network connected to the interface is treated as a directly connected network. The no command removes the configuration.

d) Configure the management boundary

Command	Explanation
Interface Mode	
ip pim scope-border <*1-99* >| **<*acl_name*>** **no ip pim scope-border**	Configures a PIM-DM management boundary on the interface and applies an ACL to it. By default, 239.0.0.0/8 is the management group scope. If an ACL is configured, the scope permitted by the ACL is the management group scope. The no command removes the configuration.

4. Disable PIM-DM

Command	Explanation
Interface Mode	
no ip pim dense-mode	Disables PIM-DM on the interface.
Global Mode	
no ip pim multicast-routing	Disables PIM-DM globally.

6.2.3 PIM-DM Configuration Examples

As shown in the following figure, add the Ethernet ports of Switch A and Switch B to the corresponding VLANs, and enable PIM-DM on each VLAN interface.

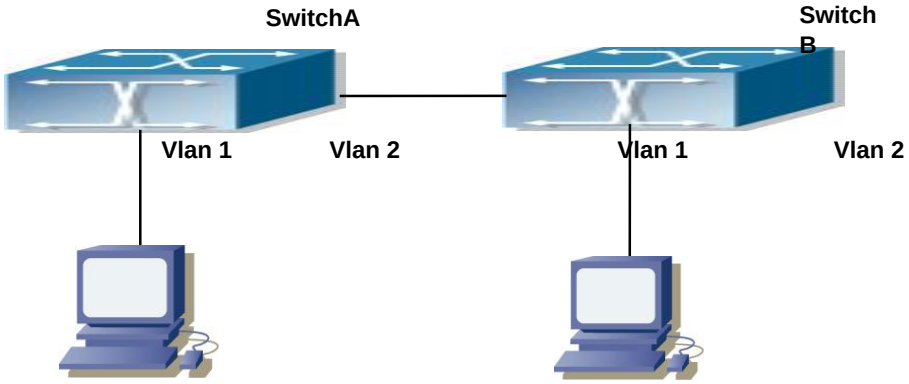


Figure 6-1 Typical PIM-DM Environment

The configuration procedure for SwitchA and SwitchB is as follows:

(1) Configure SwitchA:

```
Switch(config)#ip pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)# ip address 10.1.1.1 255.255.255.0
Switch(Config-if-Vlan1)# ip pim dense-mode
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan2
Switch(Config-if-Vlan2)# ip address 12.1.1.1 255.255.255.0
Switch(Config-if-Vlan2)# ip pim dense-mode
```

(2) Configure SwitchB:

```
Switch(config)#ip pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)# ip address 12.1.1.2 255.255.255.0
Switch(Config-if-Vlan1)# ip pim dense-mode
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)# ip address 20.1.1.1 255.255.255.0
Switch(Config-if-Vlan2)# ip pim dense-mode
```

Also make sure the unicast routing protocol is configured so that all devices can communicate with each other at the network layer and routes are updated dynamically through the unicast routing protocol.

6.2.4 PIM-DM Troubleshooting

When PIM-DM is configured and used, it may fail to operate normally because of physical connection problems or incorrect configuration. Check the following:

- Make sure the physical connections are correct.
- Make sure the interface and line protocol are up (use the show interface command).
- Make sure PIM is enabled in Global Mode (use ipv6 pim multicast-routing).
- Make sure PIM-DM is enabled on the interface (use the ipv6 pim dense-mode command).
- Multicast protocols use unicast routing for the RPF check, so make sure unicast routing is correct first.

If the PIM-DM problem persists after all these checks, run debug commands such as debug pim, copy the debug output for 3 minutes, and contact EnGenius technical support.

6.3 PIM-SM

6.3.1 Introduction to PIM-SM

PIM-SM (Protocol Independent Multicast, Sparse Mode) is a sparse-mode multicast routing protocol. It is mainly used in large networks where group members are relatively sparse and widely distributed. Unlike the flooding and pruning of dense mode, PIM-SM assumes that no host needs to receive multicast data packets. A PIM-SM router forwards multicast data packets to a host only when the host explicitly requests them.

By configuring an RP (Rendezvous Point) and a BSR (Bootstrap Router), PIM-SM advertises multicast information to all PIM-SM routers and builds an RPT (RP-rooted shared tree) using the Join/Prune messages of the routers. This reduces the network bandwidth used by data packets and control messages and lowers the processing overhead of the routers. Multicast data flows along the shared tree to the network segments where multicast group members are located. When the data traffic reaches a certain level, the multicast data stream can be switched to a source-based shortest path tree (SPT) to reduce network delay. PIM-SM does not depend on any specific unicast routing protocol; it performs the RPF check using the existing unicast routing table.

1. How PIM-SM Works

The main PIM-SM processes are neighbor discovery, RP shared tree (RPT) generation, multicast source registration, and SPT switchover. Neighbor discovery works the same way as in PIM-DM and is not described here.

(1) RP Shared Tree (RPT) Generation

When a host joins multicast group G, the leaf router directly connected to the host learns from IGMP messages that there is a receiver for group G. It determines the RP for group G and sends a Join message upstream toward the RP. Every router on the path from the leaf router to the RP creates a (*, G) entry, which applies to messages from any source to the multicast group. When the RP receives messages sent to group G, the messages travel along the established path to the leaf router and reach the host. In this way, an RPT rooted at the RP is built.

(2) Multicast Source Registration

When multicast source S sends a multicast packet to multicast group G, the PIM-SM multicast router directly connected to the source encapsulates the multicast packet in a Register message and unicasts it to the corresponding RP. If there is more than one PIM-SM multicast router on a network segment, the DR (Designated Router) is responsible for sending the multicast packets.

(3) SPT Switchover

When a multicast router finds that the rate of multicast packets from the RP with destination address G exceeds a threshold, it sends a Join message to the next upstream node toward the source, which switches the traffic from the RPT to the SPT.

2. Preparation Before PIM-SM Configuration

(1) Configure Candidate RPs

A PIM-SM network can have multiple RPs (candidate RPs). Each C-RP (Candidate RP) is responsible for forwarding multicast packets whose destination addresses are in a certain range. Configuring multiple candidate RPs provides RP load sharing. RPs are not divided into master and slave. After receiving the candidate RP information advertised by the BSR, all multicast routers use the same algorithm to determine the RP for each multicast group.

Note that one RP can serve multiple multicast groups or all multicast groups. However, at any moment each multicast group can correspond to only one RP; it cannot correspond to multiple RPs at the same time.

(2) Configure the BSR

The BSR is the management center of a PIM-SM network. It collects the messages sent by the candidate RPs and broadcasts them.

Only one BSR can exist in a network, but multiple C-BSRs (Candidate BSRs) can be configured. If the BSR fails, another C-BSR can take over. The C-BSRs elect the BSR automatically.

6.3.2 PIM-SM Configuration Task List

1. Enable PIM-SM (required)
2. Configure static multicast routing entries (optional)
3. Configure additional PIM-SM parameters (optional)

(1) Configure PIM-SM interface parameters

   1) Configure the interval for PIM-SM hello messages
   2) Configure the hold time for PIM-SM hello messages
   3) Configure ACL for PIM-SM neighbors
   4) Configure the interface as a PIM-SM boundary interface
   5) Configure the interface as a PIM-SM management boundary

(2) Configure global PIM-SM parameters

   1) Configure the switch as a candidate BSR
   2) Configure the switch as a candidate RP
   3) Configure static RP
   4) Configure the cache time of kernel multicast routes

4. Disable PIM-SM

1. Enable PIM-SM

To enable PIM-SM on a Layer 3 switch, enable PIM in Global Mode, and then enable PIM-SM on specific interfaces in Interface Mode.

Command	Explanation
Global Mode	
ip pim multicast-routing	Enables PIM-SM globally for all interfaces (required). To make PIM-SM work on a specific interface, the following interface command is also required.

Then enable PIM-SM on the interface.

Command	Explanation
Interface Mode	
ip pim sparse-mode	Enables PIM-SM on the interface (required).

2. Configure static multicast routing entries

Command	Explanation
Global Mode	
ip mroute ***<A.B.C.D> <A.B.C.D> <ifname> <.ifname>*** **no ip mroute** ***<A.B.C.D> <A.B.C.D>*** **[***<ifname> <.ifname>***]**	Configures a static multicast routing entry. The no command deletes the specified static multicast routing entry.

3. Configure additional PIM-SM parameters

(1) Configure PIM-SM interface parameters

   1) Configure the interval for PIM-SM hello messages

Command	Explanation
Interface Mode	
ip pim hello-interval <*interval*> **no ip pim hello-interval**	Configures the interval for sending PIM-SM hello messages. The no command restores the default interval.

2) Configure the hold time for PIM-SM hello messages

Command	Explanation
Interface Mode	
ip pim hello-holdtime <*value*> **no ip pim hello-holdtime**	Configures the value of the holdtime field in PIM-SM hello messages. The no command restores the default hold time.

3) Configure ACL for PIM-SM neighbors

Command	Explanation
Interface Mode	
ip pim neighbor-filter{<*access-list-number*> } **no ip pim neighbor-filter{<*access-list-number*> }**	Configures an ACL to filter PIM-SM neighbors. If the session with a neighbor is denied by the ACL, an established session is torn down immediately and no new session is set up.

4) Configure the interface as a PIM-SM boundary interface

Command	Explanation
Interface Mode	
ip pim bsr-border **no ip pim bsr-border**	Configures the interface as a PIM-SM boundary. BSR messages are neither sent nor received on a boundary interface, and the network connected to the interface is treated as a directly connected network. The no command removes the configuration.

5) Configure the interface as a PIM-SM management boundary

Command	Explanation
Interface Mode	
ip pim scope-border <*1-99* > | **<*acl_name*>** **no ip pim scope-border**	Configures a PIM-SM management boundary on the interface and applies an ACL to it. By default, 239.0.0.0/8 is the management group scope. If an ACL is configured, the scope permitted by the ACL is the management group scope. acl_name must be a standard IPv4 ACL name. The no command removes the configuration.

(2) Configure global PIM-SM parameters

1) Configure the switch as a candidate BSR

Command	Explanation
Global Mode	
ip pim bsr-candidate {vlan <*vlan-id*>| <*ifname*>}[<*mask-length*>] **[<*priority*>]** **no ip pim bsr-candidate**	Configures the switch as a PIM-SM candidate BSR so that it can compete with other candidate BSRs to become the BSR. The **no ip pim bsr-candidate** command removes the candidate BSR configuration.

2) Configure the switch as a candidate RP

Command	Explanation

Global Mode	
ip pim rp-candidate { vlan <*vlan-id*>| lookback<*index*> <*ifname*>} [<*A.B.C.D*>][<*priority*>] **no ip pim rp-candiate**	Configures the switch as a PIM-SM candidate RP so that it can compete with other candidate RPs to become the RP. The **no ip pim rp-candidate** command removes the candidate RP configuration.

3) Configure static RP

Command	Explanation
Global Mode	
ip pim rp-address <*A.B.C.D*> [<*A.B.C.D/M*>] **no ip pim rp-address <*A.B.C.D*> {<*all*>|<*A.B.C.D/M*>}**	Configures a static RP globally or for a multicast address range. The no command removes the static RP configuration.

4) Configure the cache time of kernel multicast routes

Command	Explanation
Global Mode	
ip multicast unresolved-cache aging-time <value> **no ip multicast unresolved-cache aging-time**	Configures the cache time of kernel multicast routes. The no command restores the default value.

4. Disable PIM-SM

Command	Explanation
Interface Mode	
no ip pim sparse-mode | no ip pim multicast-routing(Global configuration mode)	Disables PIM-SM.

6.3.3 PIM-SM Configuration Examples

As shown in the following figure, add the Ethernet ports of SwitchA, SwitchB, SwitchC, and SwitchD to the corresponding VLANs, and enable PIM-SM on each VLAN interface.

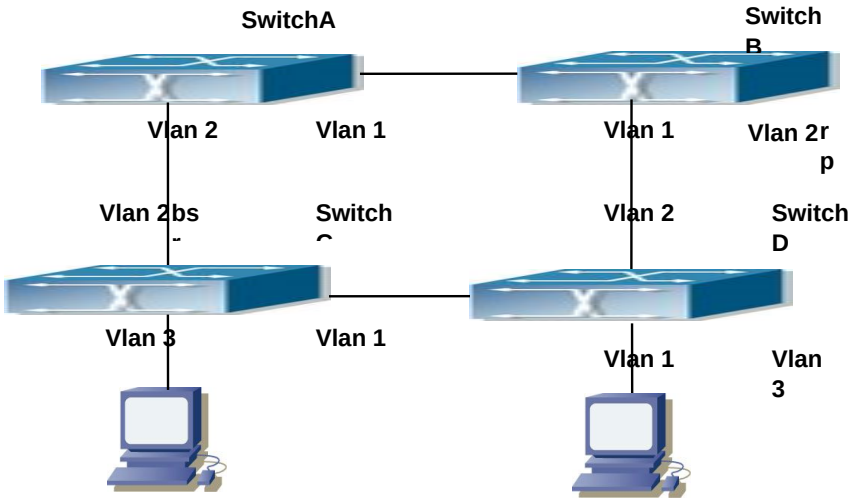


Figure 6-2 Typical PIM-SM Environment

The configuration procedure for SwitchA, SwitchB, SwitchC, and SwitchD is as follows:

(1) Configure SwitchA:

```
Switch(config)#ip pim multicast-routing
Switch(config)#interface vlan 1
```

```
Switch(Config-if-Vlan1)# ip address 12.1.1.1 255.255.255.0
Switch(Config-if-Vlan1)# ip pim sparse-mode
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)# ip address 13.1.1.1 255.255.255.0
Switch(Config-if-Vlan2)# ip pim sparse-mode
```

(2) Configure SwitchB:

```
Switch(config)#ip pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)# ip address 12.1.1.2 255.255.255.0
Switch(Config-if-Vlan1)# ip pim sparse-mode
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)# ip address 24.1.1.2 255.255.255.0
Switch(Config-if-Vlan2)# ip pim sparse-mode
Switch(Config-if-Vlan2)# exit
Switch(config)# ip pim rp-candidate vlan2
```

(3) Configure SwitchC:

```
Switch(config)#ip pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)# ip address 34.1.1.3 255.255.255.0
Switch(Config-if-Vlan1)# ip pim sparse-mode
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)# ip address 13.1.1.3 255.255.255.0
Switch(Config-if-Vlan2)# ip pim sparse-mode
Switch(Config-if-Vlan2)#exit
Switch(config)#interface vlan 3
Switch(Config-if-Vlan3)# ip address 30.1.1.1 255.255.255.0
Switch(Config-if-Vlan3)# ip pim sparse-mode
Switch(Config-if-Vlan3)# exit
Switch(config)# ip pim bsr-candidate vlan2 30 10
```

(4) Configure SwitchD:

```
Switch(config)#ip pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)# ip address 34.1.1.4 255.255.255.0
Switch(Config-if-Vlan1)# ip pim sparse-mode
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)# ip address 24.1.1.4 255.255.255.0
Switch(Config-if-Vlan2)# ip pim sparse-mode
Switch(Config-if-Vlan2)#exit
Switch(config)#interface vlan 3
Switch(Config-if-Vlan3)# ip address 40.1.1.1 255.255.255.0
Switch(Config-if-Vlan3)# ip pim sparse-mode
```

Also make sure the unicast routing protocol is configured so that all devices can communicate with each other at the network layer and routes are updated dynamically through the unicast routing protocol.

6.3.4 PIM-SM Troubleshooting

When PIM-SM is configured and used, it may fail to operate normally because of physical connection problems or incorrect configuration. Check the following:

- Make sure the physical connections are correct.
- Make sure the interface and line protocol are up (use the show interface command).
- Make sure PIM is enabled in Global Mode (use ip pim multicast-routing).

- Make sure PIM-SM is configured on the interface (use ip pim sparse-mode).
- Multicast protocols use unicast routing for the RPF check, so make sure unicast routing is correct first.
- PIM-SM requires an RP and a BSR. First use show ip pim bsr-router to check whether BSR information exists. If it does not, check whether there is a unicast route to the BSR.
- Use the show ip pim rp-hash command to check whether the RP information is correct. If there is no RP information, check unicast routing.

If the PIM-SM problem persists after all these checks, run debug commands such as debug pim or debug pim BSR, copy the debug output for 3 minutes, and contact EnGenius technical support.

6.4 MSDP

6.4.1 Introduction to MSDP

MSDP (Multicast Source Discovery Protocol) is a protocol used to learn information about multicast sources in other PIM-SM domains. An RP configured with MSDP advertises information about the multicast sources in its domain to all other MSDP peers through SA messages. In this way, multicast source information in one PIM-SM domain is propagated to other domains. MSDP uses an inter-domain source tree instead of the shared tree. The intra-domain multicast routing protocol must be PIM-SM.

- RP workflow in PIM-SM

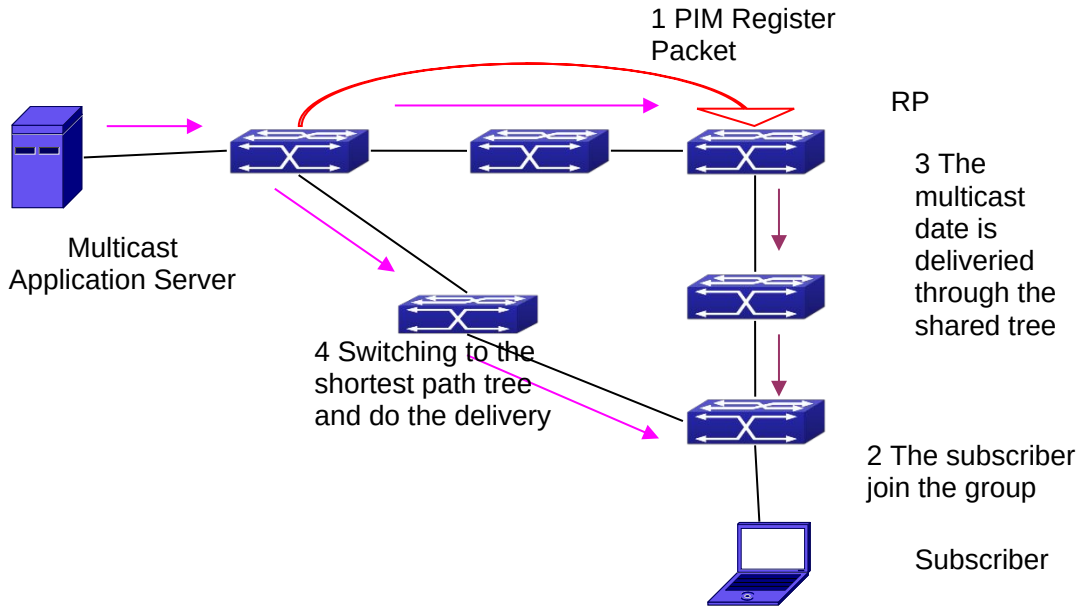


Figure 6-13

6.4.2 MSDP Configuration Task List

1. Configure basic MSDP functions
   1) Enable MSDP (required)
   2) Configure MSDP peers (required)
   3) Configure the connect-source interface
   4) Configure static RPF peers
   5) Configure the originator RP

6) Configure the TTL value

2. Configure MSDP peers
   1) Configure the connect-source interface
   2) Configure the description of MSDP peers
   3) Configure the AS number
   4) Configure the MSDP mesh group
   5) Configure the maximum size of the cache
3. Configure SA packet delivery
   1) Configure filter policies for creating SA packets
   2) Configure filter rules for receiving and forwarding SA packets
   3) Configure SA request packets
   4) Configure filter policies for SA request packets
4. Configure SA cache parameters
   1) Configure the SA packet cache
   2) Configure the aging time of SA cache entries
   3) Configure the maximum size of the cache

6.4.3 Basic MSDP Configuration

All commands in this section are configured on RPs in a PIM-SM domain. These RPs act as MSDP peers of each other.

6.4.3.1 Prerequisites for MSDP Configuration

Before configuring basic MSDP functions, complete the following tasks:

- Configure at least one unicast routing protocol to provide connectivity within and between domains.
- Configure PIM-SM to implement multicast within the domain.

Before configuring basic MSDP functions, prepare the following information:

- IP addresses of the MSDP peers
- Filter policy tables

Note: MSDP cannot be used together with Anycast RP; however, Anycast RP can be configured based on MSDP.

6.4.3.2 Enable MSDP

MSDP must be enabled before other MSDP functions can be configured.

1. Enable MSDP
2. Configure MSDP

1. Enable MSDP

Command	Explanation
Global Mode	
router msdp **no router msdp**	Enables MSDP. The no command disables MSDP globally.

2. Configure MSDP parameters

Command	Explanation

MSDP Configuration Mode	
connect-source <*interface-type*> <*interface-number*> **no connect-source**	Configures the connect-source interface for MSDP peers. The no command removes the configured connect-source interface.
default-rpf-peer ***<peer-address> [rp-policy <acl-list-number>|<word>]*** ***no* default-rpf-peer**	Configures a static RPF peer. The no command removes the configured RPF peer.
originating-rp <*interface-type*> <*interface-number*> **no originating-rp**	Configures the originator RP. The no command removes the configured originator RP.
ttl-threshold <*ttl*> **no ttl-threshold**	Configures the TTL value. The no command removes the configured TTL value.

6.4.4 MSDP Peer Configuration

6.4.4.1 Create an MSDP Peer

Command	Explanation
MSDP Configuration Mode	
peer <*peer-address*> **no peer <*peer-address*>**	Creates an MSDP peer. The no command deletes the configured MSDP peer.

6.4.4.2 Configure MSDP Peer Parameters

Command	Explanation
MSDP Peer Configuration Mode	
connect-source <*interface-type*> <*interface-number*> **no connect-source**	Configures the connect-source interface for the MSDP peer. The no command removes the configured connect-source interface.
description <*text*> **no description**	Configures the description of the MSDP peer. The no command removes the configured description.
remote-as <*as-num*> **no remote-as <*as-num*>**	Configures the AS number of the MSDP peer. The no command removes the configured AS number of the MSDP peer.
mesh-group <*name*> **no mesh-group <*name*>**	Adds the MSDP peer to the specified mesh group. The no command removes the MSDP peer from the specified mesh group.

6.4.5 MSDP Packet Delivery Configuration

Command	Explanation
MSDP Configuration Mode	
redistribute [list <*acl-list-number* |*acl-name*>] **no redistribute**	Configures the filter rules for creating SA packets. The no command removes the configured filter rules.
MSDP Configuration Mode or MSDP Peer Configuration Mode	
sa-filter（**in|out**）**[list <*acl-number* | *acl-name*> | rp-list <*rp-acl-number* | *rp-acl-name*>]** **no sa-filter**（**in|out**）**[[list <*acl-number* |**	Configures the filter rules for receiving and forwarding SA packets. The no command removes the configured rules.

***acl-name*>	rp-list <*rp-acl-number*	*rp-acl-name*>]**	
MSDP Peer Configuration Mode	
sa-request **no sa-request**	Enables sending SA request packets. The no command disables sending SA request packets.
MSDP Configuration Mode	
**sa-request-filter [list <*access-list-number*	*access-list-name*>]** **no sa-request-filter [list <*access-list-number*	*access-list-name*>]**	Configures the filter rules for receiving SA request packets. The no command removes the configured filter rules for SA request packets.

6.4.6 SA Cache Parameter Configuration

Command	Explanation
MSDP Configuration Mode	
cache-sa-state **no cache-sa-state**	Enables the SA packet cache. The no command disables the SA packet cache.
MSDP Configuration Mode	
cache-sa-holdtime <*150-3600*> **no cache-sa-holdtime**	Configures the aging time of entries in the SA cache. The no command restores the default aging time.
MSDP Configuration Mode or MSDP Peer Configuration Mode	
cache-sa-maximum <*sa-limit*> **no cache-sa-maximum**	Configures the maximum size of the SA cache. The no command restores the default SA cache size.

6.4.7 MSDP Configuration Examples

Example 1: Basic MSDP functions.

Multicast process:

1. Suppose the multicast server sends multicast datagrams to 224.1.1.1.
2. The designated router (DR) connected to the multicast server encapsulates the multicast datagrams in Register packets and sends them to the RP (RP1) in the local domain.
3. The RP decapsulates the packets and sends them to all domain members through the shared tree. Members in the domain can be configured to be on the shared tree or not.
4. At the same time, the source RP in the domain generates an SA (Source Active) message and sends it to its MSDP peer, RP2.
5. If there is another member in the same domain as an MSDP peer, named RP3, RP3 distributes the multicast datagrams encapsulated in the SA messages to the members of the shared tree and sends Join messages toward the multicast source. That is, the RP creates an (S, G) entry and sends (S, G) Join messages hop by hop, so that an SPT rooted at the multicast source is built across the PIM-SM domains.

If there are no members in the same domain as MSDP peer RP2, RP2 neither creates an (S, G) entry nor joins the SPT rooted at the multicast source.

6. After the reverse path is set up, multicast datagrams from the source are delivered directly to RP3, and the RP forwards them along the shared tree. The router closest to the domain members can then decide whether to switch to the SPT.

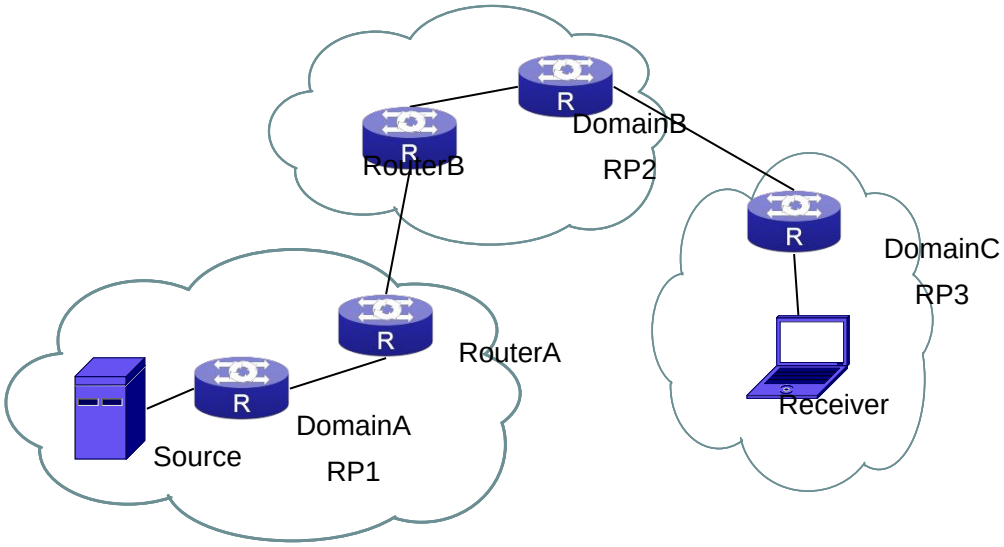


Figure 6-4 Network Topology for MSDP Peers

Configuration tasks:

Prerequisites:

Enable a unicast routing protocol and PIM on every router, and make sure that inter-domain routing and intra-domain multicast work properly.

Suppose multicast server S in Domain A provides a multicast program at 224.1.1.1, and host R in Domain C subscribes to this program. Before MSDP is configured, R cannot receive the program. After the following configuration, R can receive the program from S.

RP1 in Domain A:

```
Switch#config
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ip address 10.1.1.1 255.255.255.0
Switch(Config-if-Vlan1)#exit
Switch(config)#router msdp
Switch(router-msdp)#peer 10.1.1.2
```

Router A in Domain A:

```
Switch#config
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ip address 10.1.1.2  255.255.255.0
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)#ip address 20.1.1.2  255.255.255.0
Switch(Config-if-Vlan2)#exit
Switch(config)#router msdp
Switch(router-msdp)#peer 10.1.1.1
Switch(msdp-peer)#exit
Switch(router-msdp)#peer 20.1.1.1
```

Router B in Domain B:

```
Switch#config
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)#ip address 20.1.1.1  255.255.255.0
Switch(Config-if-Vlan2)#exit
Switch(Config)#interface vlan 3
```

```
Switch(Config-if-Vlan3)#ip address 30.1.1.1  255.255.255.0
Switch(Config-if-Vlan3)#exit
Switch(config)#router msdp
Switch(router-msdp)#peer 20.1.1.2
Switch(msdp-peer)#exit
Switch(router-msdp)#peer 30.1.1.2
```

RP2 in Domain B:

```
Switch#config
Switch(config)#interface vlan 3
Switch(Config-if-Vlan3)#ip address 30.1.1.2  255.255.255.0
Switch(config)#interface vlan 4
Switch(Config-if-Vlan4)#ip address 40.1.1.2  255.255.255.0
Switch(Config-if-Vlan4)#exit
Switch(config)#router msdp
Switch(router-msdp)#peer 30.1.1.1
Switch(config)#router msdp
Switch(router-msdp)#peer 40.1.1.1
```

RP3 in Domain C:

```
Switch(config)#interface vlan 4
Switch(Config-if-Vlan1)#ip address 40.1.1.1  255.255.255.0
Switch(Config-if-Vlan1)#exit
Switch(config)#router msdp
Switch(router-msdp)#peer 40.1.1.2
```

Example 2: MSDP mesh group application.

A mesh group can be used to reduce the flooding of SA messages. Fully meshed peers in the same domain can be configured as a mesh group. All members of a mesh group use the same group name.

As shown in the following figures, when a mesh group is configured for the four fully meshed peers in the same domain, the flooding of SA messages is greatly reduced.

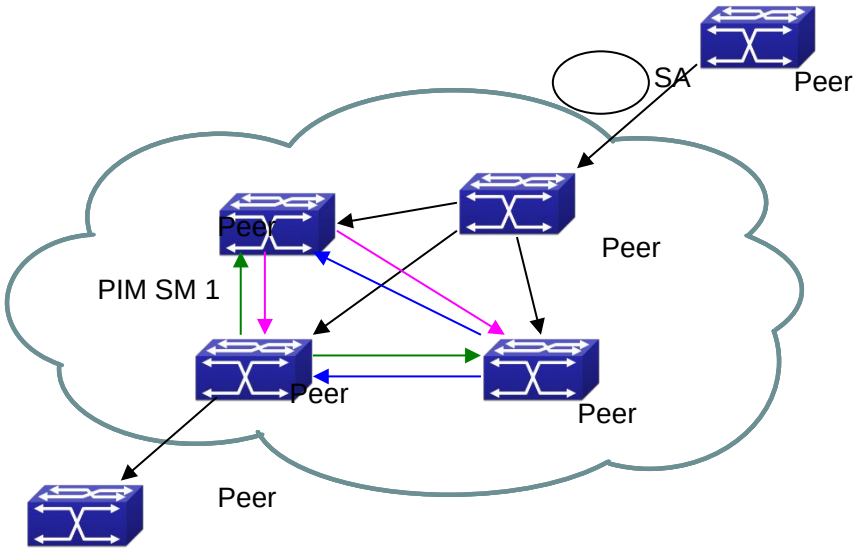


Figure 6-5 Flooding of SA Messages

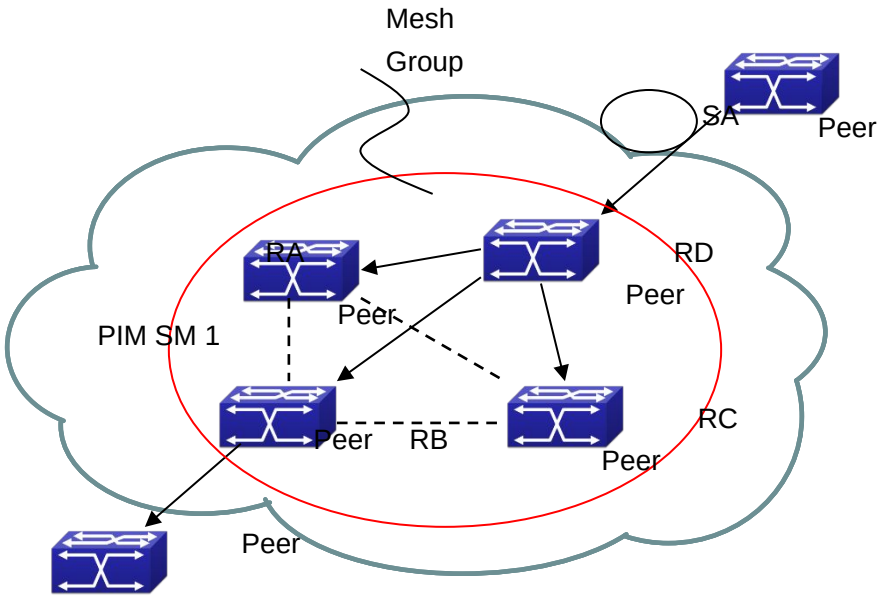


Figure 6-6 Flooding of SA Messages with a Mesh Group

Configuration steps:

Router A:

```
Switch#config
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ip address 10.1.1.1 255.255.255.0
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)#ip address 20.1.1.1  255.255.255.0
Switch(Config-if-Vlan2)#exit
Switch(config)#interface vlan 3
Switch(Config-if-Vlan3)#ip address 30.1.1.1  255.255.255.0
Switch(Config-if-Vlan3)#exit
Switch(config)#router msdp
Switch(router-msdp)#peer 10.1.1.2
Switch(router-msdp)#mesh-group test-1
Switch(msdp-peer)#exit
Switch(router-msdp)#peer 20.1.1.4
Switch(router-msdp)#mesh-group test-1
Switch(msdp-peer)#exit
Switch(router-msdp)#peer 30.1.1.3
Switch(router-msdp)#mesh-group test-1
Switch(msdp-peer)#exit
```

Router B:

```
Switch#config
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ip address 10.1.1.2  255.255.255.0
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan 4
Switch(Config-if-Vlan4)#ip address 40.1.1.2  255.255.255.0
Switch(Config-if-Vlan4)#exit
Switch(config)#interface vlan 6
Switch(Config-if-Vlan6)#ip address 60.1.1.2  255.255.255.0
Switch(Config-if-Vlan6)#exit
Switch(config)#router msdp
Switch(router-msdp)#peer 10.1.1.1
Switch(router-msdp)#mesh-group test-1
Switch(msdp-peer)#exit
Switch(router-msdp)#peer 40.1.1.4
```

```
Switch(router-msdp)#mesh-group test-1
Switch(msdp-peer)#exit
Switch(router-msdp)#peer 60.1.1.3
Switch(router-msdp)#mesh-group test-1
```

Router C:

```
Switch#config
Switch(config)#interface vlan 4
Switch(Config-if-Vlan4)#ip address 40.1.1.4  255.255.255.0
Switch(Config-if-Vlan4)#exit
Switch(config)#interface vlan 5
Switch(Config-if-Vlan5)#ip address 50.1.1.4  255.255.255.0
Switch(Config-if-Vlan5)#exit
Switch(config)#interface vlan 6
Switch(Config-if-Vlan6)#ip address 60.1.1.4  255.255.255.0
Switch(Config-if-Vlan6)#exit
Switch(config)#router msdp
Switch(router-msdp)#peer 20.1.1.1
Switch(router-msdp)#mesh-group test-1
Switch(msdp-peer)#exit
Switch(router-msdp)#peer 40.1.1.4
Switch(router-msdp)#mesh-group test-1
Switch(msdp-peer)#exit
Switch(router-msdp)#peer 60.1.1.2
Switch(router-msdp)#mesh-group test-1
```

Router D:

```
Switch#config
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)#ip address 20.1.1.4 255.255.255.0
Switch(Config-if-Vlan2)#exit
Switch(config)#interface vlan 4
Switch(Config-if-Vlan1)#ip address 40.1.1.4  255.255.255.0
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan 5
Switch(Config-if-Vlan5)#ip address 50.1.1.4  255.255.255.0
Switch(Config-if-Vlan5)#exit
Switch(config)#router msdp
Switch(router-msdp)#peer 20.1.1.1
Switch(router-msdp)#mesh-group test-1
Switch(msdp-peer)#exit
Switch(router-msdp)#peer 40.1.1.2
Switch(router-msdp)#mesh-group test-1
Switch(msdp-peer)#exit
Switch(router-msdp)#peer 50.1.1.3
Switch(router-msdp)#mesh-group test-1
```

6.4.8 MSDP Troubleshooting

MSDP may fail to work because of physical link failures or configuration errors. Check the following to make MSDP work:

- Make sure the physical links work properly.
- Make sure intra-domain and inter-domain routing works.
- Make sure PIM-SM is used as the intra-domain routing protocol in every domain and is configured correctly.
- Make sure MSDP is enabled and the link status of each MSDP peer is up.
- Use the **show msdp global** command to check whether the MSDP configuration is correct.

If the MSDP problem persists after these checks, run the **debug msdp** command, collect the debug messages for three minutes, and contact EnGenius technical support.

6.5 Anycast RP

6.5.1 Introduction to Anycast RP

Anycast RP is a PIM-based technology that provides RP redundancy so that service recovers as quickly as possible when an RP becomes unavailable.

The core idea of Anycast RP is that the same RP address, configured throughout the network, exists on multiple multicast servers. (Typically, each device providing Anycast RP uses a loopback interface and configures the RP address on it with the longest mask.) Unicast routing ensures that PIM routers can always find the nearest RP, which provides a shorter and faster path to the RP in a large network. When the RP in use becomes unavailable, unicast routing ensures that PIM routers quickly find a path to a new RP, so multicast service is restored in time. However, multiple RPs introduce a new problem: if the multicast source and the receivers register with different RPs, some receivers cannot receive data from the multicast source (register messages naturally go only to the nearest RP). To keep all RPs synchronized, Anycast RP specifies that the RP nearest to the multicast source forwards the source register messages to all other RPs, so that all receivers that joined any RP can find the multicast source.

PIM-based Anycast RP is implemented as follows: each switch configured with Anycast RP maintains an Anycast RP list and uses another address as an identifier to recognize the other RPs. When an Anycast RP device receives a register message, it forwards the message to the other Anycast RP devices, using its own address as the source address, to notify them of the original destination.

6.5.2 Anycast RP Configuration Task List

1. Enable Anycast RP v4
2. Configure Anycast RP v4

1. Enable Anycast RP v4

Command	Explanation
Global Mode	
ip pim anycast-rp **no ip pim anycast-rp**	Enables the Anycast RP function (required). The no command disables the Anycast RP function globally.

2. Configure Anycast RP v4

(1) Configure the candidate RP

Command	Explanation
Global Mode	
ip pim rp-candidate {vlan<*vlan-id*> | loopback<*index*> |<*ifname*>} [<*A.B.C.D*>] [<*priority*>] **no ip pim rp-candidate**	PIM-SM allows a loopback interface to be a candidate RP (required). Note that Anycast RP can use a loopback interface or a regular Layer 3 VLAN interface as the candidate RP. To make sure that PIM routers in the network can find the RP, the candidate RP interface must be added to the routing (advertised by the unicast routing protocol). The no command removes the candidate RP configuration on this router.

(2) Configure the self-rp-address (the RP address of this router)

Command	Explanation
Global Mode	
ip pim anycast-rp self-rp-address A.B.C.D **no ip pim anycast-rp self-rp-address**	Configures the self-rp-address of this router (as an RP). This address uniquely identifies this router when it communicates with other RPs. The **self-rp-address** has two functions: 1 When this router (as an RP) receives a register message unicast from the DR, it forwards the register message to all other RPs in the network to notify them of the (S, G) source state. When forwarding the register message, this router changes its source address to the self-rp-address. 2 When this router (as an RP) receives a register message unicast from another RP, that is, a register message whose destination is the self-rp-address of this router, it creates the (S, G) state and sends back a Register-Stop message whose destination address is the source address of the register message. Note: The self-rp-address must be the address of a Layer 3 interface on this router, although it can be configured before the interface exists. The self-rp-address must be unique. The no command removes the self-rp-address that this router (as an RP) uses to communicate with other RPs.

(3) Configure the other-rp-address (communication addresses of other RPs)

Command	Explanation
Global Mode	
ip pim anycast-rp <*anycast-rp-addr*> <*other-rp-addr*> **no ip pim anycast-rp <*anycast-rp-addr*> <*other-rp-addr*>**	Configures the anycast-rp-addr on this router (as an RP). This unicast address is the RP address configured on multiple RPs in the network and matches the address of the candidate RP interface (or loopback interface). The **anycast-rp-addr** works as follows: 1 Multiple anycast-rp-addr addresses can be configured, but only the one that matches the currently configured candidate RP address takes effect. Only then do the other-rp-addresses associated with this anycast-rp-addr take effect. 2 The anycast-rp-addr can be configured before the corresponding interface exists. Configures, on this router (as an RP), the other-rp-addresses of the other RPs that communicate with it. This unicast address identifies another RP and is used for communication with the local router. The **other-rp-address** has two functions: 1 When this router (as an RP) receives a register message unicast from a DR, it forwards the message to the other RPs in

	the network to notify all RPs of the (S, G) source state. When forwarding, the router changes the destination address of the register message to the other-rp-address. 2 Multiple other-rp-addresses can be configured for one anycast-rp-addr. When a register message from a DR is received, it is forwarded to each of these RPs in turn. The no command removes an other-rp-address that communicates with this router.

6.5.3 Anycast RP Configuration Examples

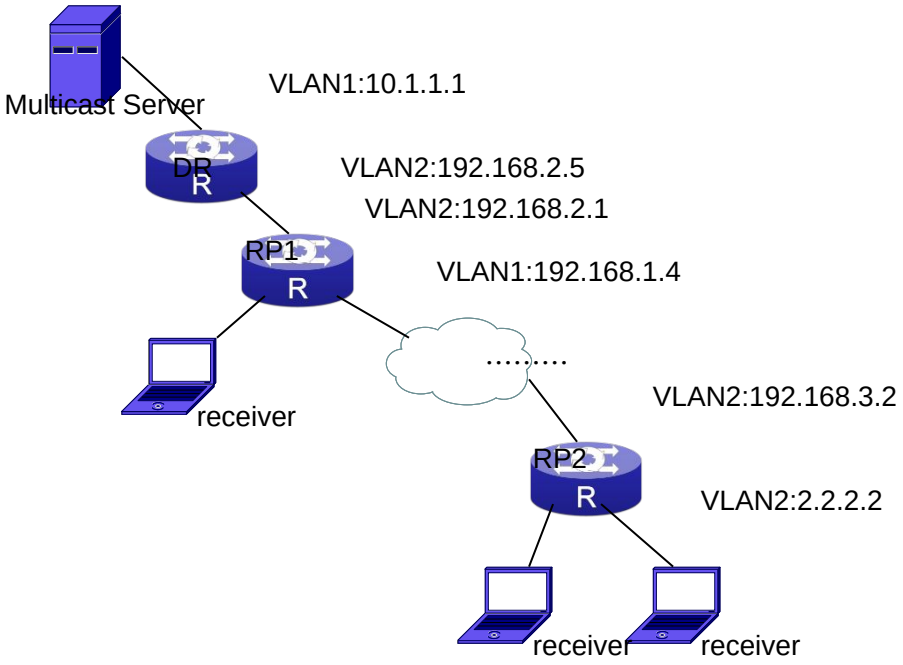


Figure 6-7 Anycast RP v4 Function of the Router

As shown in the figure, the network runs PIM-SM and has two routers that support Anycast RP: RP1 and RP2. When multicast data from the multicast source server reaches the DR, the DR unicasts a source register message to the nearest RP according to unicast routing, which is RP1 in this example. When RP1 receives the register message from the DR, it distributes the data along the shared tree to the receivers that have joined it, and it also forwards the register message to RP2 so that all receivers that have joined RP2 can find the multicast source. RP1, configured with Anycast RP, maintains an Anycast list that contains the unicast addresses of all other RPs in the network. When RP1 receives the register message, it forwards it to RP2 using its self-rp-address as the source address. The cloud in the figure represents the PIM-SM network between RP1 and RP2.

Configuration steps:

RP1 Configuration:

```
Switch#config
Switch(config)#interface loopback 1
Switch(Config-if-Loopback1)#ip address 1.1.1.1 255.255.255.255
Switch(Config-if-Loopback1)#exit
Switch(config)#ip pim rp-candidate loopback1
Switch(config)#ip pim bsr-candidate vlan 1
Switch(config)#ip pim multicast-routing
Switch(config)#ip pim anycast-rp
Switch(config)#ip pim anycast-rp self-rp-address 192.168.2.1
Switch(config)#ip pim anycast-rp 1.1.1.1 192.168.3.2
```

RP2 Configuration:

```
Switch#config
Switch(config)#interface loopback 1
Switch(Config-if-Loopback1)#ip address 1.1.1.1 255.255.255.255
Switch(Config-if-Loopback1)#exit
Switch(config)#ip pim rp-candidate loopback1
Switch(config)#ip pim multicast-routing
Switch(config)#ip pim anycast-rp
Switch(config)#ip pim anycast-rp self-rp-address 192.168.3.2
Switch(config)#ip pim anycast-rp 1.1.1.1 192.168.2.1
```

6.5.4 ANYCAST RP Troubleshooting

Anycast RP may fail to work because of physical connection problems, configuration errors, or other faults. Check the following:

- Make sure the physical connections are correct.
- Make sure PIM-SM operates normally.
- Make sure Anycast RP is enabled in Global Mode.
- Make sure the self-rp-address is configured correctly in Global Mode.
- Make sure the other-rp-address is configured correctly in Global Mode.
- Make sure all interface routes are added correctly, including the loopback interface used as the RP.
- Use the **show ip pim anycast rp status** command to check whether the Anycast RP configuration is correct.

If the Anycast RP problem persists after these checks, run debug commands such as **debug pim anycast-rp**, copy the debug output for three minutes, and contact EnGenius technical support.

6.6 PIM-SSM

6.6.1 Introduction to PIM-SSM

Source-Specific Multicast (PIM-SSM) is a multicast service model in which a multicast session is identified by both the multicast group address and the multicast source address. In SSM, hosts can join multicast groups as efficiently as in traditional PIM-SM, but the shared tree and RP management of PIM-SM are not needed. SSM builds an SPT for (S, G), where G is the multicast group address and S is the address of the multicast source that sends datagrams to G. An (S, G) pair is called an SSM channel. SSM is best suited for one-to-many multicast applications, such as online sports video channels and news channels. By default, the SSM multicast group address range is 232.0.0.0 to 232.255.255.255; this range can be extended as needed.

6.6.2 PIM-SSM Configuration Task List

Command	Explanation
Global Mode	
ip multicast ssm {default|range *<access-list-number >*} **no ip multicast ssm**	Configures the PIM-SSM address range. The no command removes the configuration.

6.6.3 PIM-SSM Configuration Examples

As shown in the figure, the Ethernet ports of SwitchA, SwitchB, SwitchC, and SwitchD are configured in separate VLANs. PIM-SSM is enabled globally, and PIM-SM or PIM-DM is enabled on the VLAN interfaces. PIM-SM is used in this example.

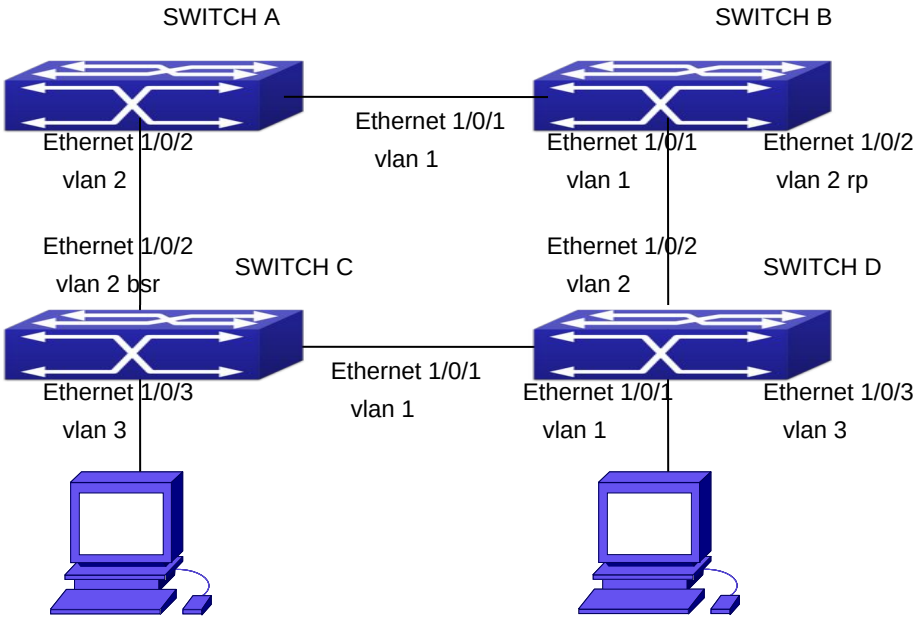


Figure 6-8 Typical PIM-SSM Environment

The configurations of SwitchA, SwitchB, SwitchC, and SwitchD are as follows.

(1) Configure SwitchA.

```
Switch(config)#ip pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-If-Vlan1)# ip pim sparse-mode
Switch(Config-If-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-If-Vlan2)# ip pim sparse-mode
Switch(Config-If-Vlan2)#exit
Switch(config)#access-list 1 permit 224.1.1.1 0.0.0.255
Switch(config)#ip multicast ssm range 1
```

(2) Configure SwitchB.

```
Switch(config)#ip pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-If-Vlan1)# ip pim sparse-mode
Switch(Config-If-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-If-Vlan2)# ip pim sparse-mode
Switch(Config-If-Vlan2)# exit
Switch(config)# ip pim rp-candidate vlan2
Switch(config)#access-list 1 permit 224.1.1.1 0.0.0.255
Switch(config)#ip multicast ssm range 1
```

(3) Configure SwitchC.

```
Switch(config)#ip pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-If-Vlan1)# ip pim sparse-mode
Switch(Config-If-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-If-Vlan2)# ip pim sparse-mode
```

```
Switch(Config-If-Vlan2)#exit
Switch(config)#interface vlan 3
Switch(Config-If-Vlan3)# ip pim sparse-mode
Switch(Config-If-Vlan3)# exit
Switch(config)# ip pim bsr-candidate vlan2 30 10
Switch(config)#access-list 1 permit 224.1.1.1 0.0.0.255
Switch(config)#ip multicast ssm range 1
```

(4) Configure SwitchD.

```
Switch(config)#ip pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-If-Vlan1)# ip pim sparse-mode
Switch(Config-If-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-If-Vlan2)# ip pim sparse-mode
Switch(Config-If-Vlan2)#exit
Switch(config)#interface vlan 3
Switch(Config-If-Vlan3)# ip pim sparse-mode
Switch(Config-If-Vlan3)#exit
Switch(config)#access-list 1 permit 224.1.1.1 0.0.0.255
Switch(config)#ip multicast ssm range 1
```

6.6.4 PIM-SSM Troubleshooting

When PIM-SSM is configured and used, it may fail to operate normally because of physical connection problems or incorrect configuration. Check the following:

- Make sure the physical connections are correct.
- Make sure the interface and line protocol are up (use the **show interface** command).
- Make sure PIM is enabled in Global Mode (use **ip pim multicast-routing**).
- Make sure PIM-SM is configured on the interface (use **ip pim sparse-mode**).
- Make sure SSM is configured in Global Mode.
- Multicast protocols use unicast routing for the RPF check, so make sure unicast routing is correct first.

If the PIM-SSM problem persists after all these checks, run debug commands such as **debug pim event/debug pim packet**, copy the debug output for 3 minutes, and contact EnGenius technical support.

6.7 DVMRP

6.7.1 Introduction to DVMRP

DVMRP (Distance Vector Multicast Routing Protocol) is a dense-mode multicast routing protocol. In a manner similar to RIP, it builds a forwarding broadcast tree for each source, and through dynamic Prune/Graft it builds a truncated broadcast tree, that is, the shortest path tree to the source, for each source.

The main features of DVMRP are:

1. Routing exchanges that provide reverse path check information are based on distance vectors (similar to RIP).
2. Routing updates are exchanged periodically (every 60 seconds by default).
3. The TTL upper limit is 32 hops (compared with 16 for RIP).
4. Routing updates include the network mask and support CIDR.

Compared with unicast routing, multicast routing is reverse routing: what matters is where packets come from, not where they go. The information in the DVMRP routing table is therefore used to determine whether an incoming multicast packet was received on the correct interface. If not, the packet is discarded to prevent multicast loops.

The check that determines whether a packet arrived on the correct interface is called the RPF check. When multicast data packets arrive on an interface, the switch looks up the DVMRP routing table to determine the reverse path to the source network. If the arrival interface is the interface used to send unicast messages to the source, the reverse path check passes and the packets are forwarded out of all downstream interfaces. Otherwise, a fault has probably occurred, and the multicast packet is discarded.

Because not all switches support multicast, DVMRP supports tunnel-based multicast communication. A tunnel is a method of sending multicast data between DVMRP switches that are separated by switches that do not support multicast routing. Multicast data packets are encapsulated in unicast data packets and sent directly to the next switch that supports multicast. DVMRP treats tunnel interfaces and ordinary physical interfaces in the same way.

If two or more switches are connected to a multi-access network, more than one copy of a data packet may be sent to the subnet. Therefore, a designated forwarder must be selected. DVMRP achieves this through its routing exchange mechanism: when two switches on the multi-access network exchange routing information, each learns the other's routing distance to the source network, and the switch with the shortest distance to the source network becomes the designated forwarder for the subnet. If the distances are equal, the switch with the lowest IP address is selected.

After DVMRP is enabled on an interface, the switch multicasts Probe messages on that interface to other DVMRP switches to discover neighbors and learn each other's capabilities. If no Probe message is received from a neighbor before the neighbor times out, the neighbor is considered lost.

In DVMRP, source network routing information is exchanged in basically the same way as in RIP: routing report messages are sent periodically between DVMRP neighbors (every 60 seconds by default). The routing information in the DVMRP routing table is used to build the source distribution tree, that is, to determine through which neighbor the source sending the multicast packets is reached. The interface to this neighbor is called the upstream interface. A routing report includes the source network address (with network mask) and the hop count used as the routing metric.

To forward correctly, every DVMRP switch needs to know which downstream switches need to receive multicast packets from a specific source network through it. After receiving packets from a specific source, a DVMRP switch first broadcasts these multicast packets out of all downstream interfaces, that is, interfaces with other DVMRP switches that depend on it for that source. After receiving a Prune message from a downstream switch on an interface, it prunes that switch. A DVMRP switch uses poison reverse to tell its upstream switch for a specific source, "I am your downstream." The switch implements poison reverse by adding infinity (32) to the routing distance of the specific source that it advertises back upstream. Valid distance values are therefore 1 to 2 × infinity (32) - 1, that is, 1 to 63: 1 to 63 means the source network is reachable, 32 means the source network is unreachable, and 33 to 63 means the switch that generated the report depends on the upstream router to receive multicast packets from the specific source.

6.7.2 DVMRP Configuration Task List

1．Enable or disable DVMRP globally (required)

2．Enable or disable DVMRP on an interface (required)

3．Configure DVMRP sub-parameters (optional)

Configure DVMRP interface parameters

1) Configure the delay for sending report messages on a DVMRP interface and the number of messages sent each time

2) Configure the metric of a DVMRP interface
3) Configure whether DVMRP can establish neighbor relationships with DVMRP routers that do not support Prune/Graft

4．Configure DVMRP tunnel

1. Enable DVMRP globally

The basic configuration for running DVMRP on a Layer 3 switch is simple. First, enable DVMRP globally.

Command	Explanation
Global Mode	
[no] ip dvmrp multicast-routing	Enables DVMRP globally (required). The **no ip dvmrp multicast-routing** command disables DVMRP globally.

2. Enable DVMRP on the interface

After enabling DVMRP globally, enable DVMRP on the corresponding interfaces.

Command	Explanation
Interface Mode	
ip dvmrp **no ip dvmrp**	Enables DVMRP on the interface. The **no ip dvmrp** command disables DVMRP on the interface.

3. Configure DVMRP sub-parameters

(1) Configure DVMRP interface parameters

1) Configure the delay for sending report messages on a DVMRP interface and the number of messages sent each time
2) Configure the metric of a DVMRP interface
3) Configure whether DVMRP can establish neighbor relationships with DVMRP routers that do not support Prune/Graft

Command	Explanation
Interface Mode	
ip dvmrp output-report-delay <*delay_val*> [<*burst_size*>] **no ip dvmrp output-report-delay**	Configures the delay for sending DVMRP report messages on the interface and the number of messages sent each time. The **no ip dvmrp output-report-delay** command restores the default value.
ip dvmrp metric <*metric_val*> **no ip dvmrp metric**	Configures the metric of DVMRP report messages on the interface. The **no ip dvmrp metric** command restores the default value.
ip dvmrp reject-non-pruners **no ip dvmrp reject-non-pruners**	Configures the interface to refuse neighbor relationships with DVMRP routers that do not support pruning/grafting. The **no ip dvmrp reject-non-pruners** command allows such neighbor relationships again.

4. Configure a DVMRP tunnel

Command	Explanation
Global Mode	
ip dvmrp tunnel <*index*> <*src-ip*> <*dst-ip*> **no ip dvmrp tunnel {<*index*> |<*src-ip*> <*dst-ip*>}**	Configures a DVMRP tunnel. The **no ip dvmrp tunnel {<*index*> |<*src-ip*> <*dst-ip*>}** command deletes a DVMRP tunnel.

6.7.3 DVMRP Configuration Examples

As shown in the following figure, add the Ethernet ports of Switch A and Switch B to the corresponding VLANs, and enable DVMRP on each VLAN interface.

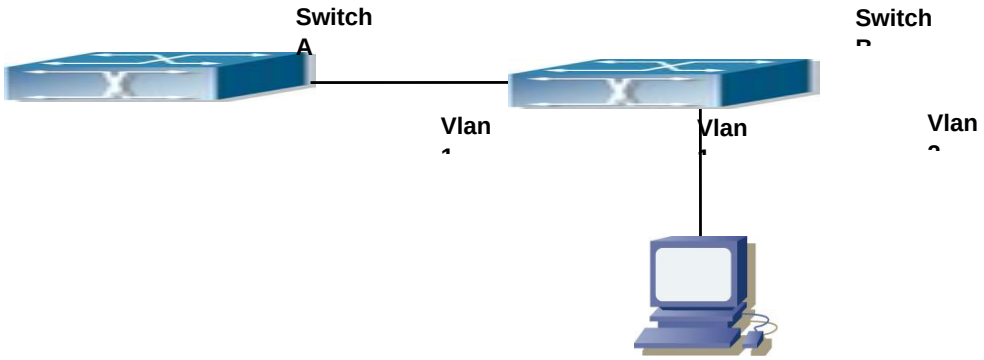


Figure 6-9 DVMRP Network Topology

The configuration procedure for SwitchA and SwitchB is as follows:

(1) Configure SwitchA:

```
Switch (config)#ip dvmrp multicast-routing
Switch (config)#interface vlan 1
Switch(Config-if-Vlan1)# ip address 10.1.1.1 255.255.255.0
Switch(Config-if-Vlan1)# ip dvmrp enable
```

(2) Configure SwitchB:

```
Switch (config)#ip dvmrp multicast-routing
Switch (config)#interface vlan 1
Switch(Config-if-Vlan1)# ip address 12.1.1.2 255.255.255.0
Switch(Config-if-Vlan1)# ip dvmrp enable
Switch(Config-if-Vlan1)#exit
Switch (config)#interface vlan 2
Switch(Config-if-Vlan2)# ip address 20.1.1.1 255.255.255.0
Switch(Config-if-Vlan2)# ip dvmrp
```

DVMRP does not depend on a unicast routing protocol, so no unicast routing protocol needs to be configured. This is a difference from PIM-DM and PIM-SM.

6.7.4 DVMRP Troubleshooting

When DVMRP is configured and used, it may fail to operate normally because of physical connection problems or incorrect configuration. Check the following:

- First, make sure the physical connections are correct.
- Next, make sure the interface and line protocol are up (use the **show interface** command).
- Check whether the correct IP address is configured on the interface (use the **ip address** command).
- Then enable DVMRP on the interface (use the **ip dvmrp** command and the **ip dv multicast-routing** command).
- Multicast protocols use unicast routing for the RPF check, so make sure unicast routing is correct first. (DVMRP uses its own unicast routing table; use the **show ip dvmrp route** command to view it.)

If the DVMRP problem persists after all these checks, run debug commands such as debug DVMRP, copy the debug output for 3 minutes, and contact EnGenius technical support.

6.8 DCSCM

6.8.1 Introduction to DCSCM

DCSCM (Destination control and source control multicast) technology covers three aspects: Multicast Packet Source Controllable, Multicast User Controllable, and Service-Oriented Priority Strategy Multicast.

The Multicast Packet Source Controllable technology of Security Controllable Multicast works as follows:

1．On an edge switch, if source-controlled multicast is configured, only multicast data from the specified sources to the specified groups can pass.

2．On an RP switch in the core of a PIM-SM network, a REGISTER message from a source or group outside the specified ones is answered directly with REGISTER_STOP, and no entry is created. (This is implemented in the PIM-SM module.)

The Multicast User Controllable technology of Security Controllable Multicast is based on controlling the IGMP report messages sent by users, so the controlled modules are IGMP Snooping and IGMP. There are three control methods: control based on the VLAN + MAC address of the sender, control based on the IP address of the sender, and control based on the ingress port of the messages. IGMP Snooping can use all three methods at the same time, whereas IGMP, which operates at Layer 3, can control only by the IP address of the sender.

The Service-Oriented Priority Strategy Multicast of Security Controllable technology works as follows: for multicast data within a specified range, the user-specified priority is set at the access (join) side, so that the data is sent with a higher priority on trunk ports. This ensures that the data is forwarded with the user-specified priority throughout the network.

6.8.2 DCSCM Configuration Task List

1．Source Control Configuration

2．Destination Control Configuration

3．Multicast Strategy Configuration

1．**Source Control Configuration**

Source control configuration has three parts. The first is to enable source control globally. The source control command is as follows:

Command	Explanation
Global Mode	
[no] ip multicast source-control (Required)	Enables source control globally. The **no ip multicast source-control** command disables source control globally. Note that after source control is enabled globally, all multicast packets are discarded by default. No other source control configuration can be made until source control is enabled globally, and source control cannot be disabled until all configured rules are removed.

The second part is to configure source control rules. They are configured in the same way as ACLs and use ACL numbers 5000-5099; each rule number can hold 10 rules. Note that the rules are ordered: the rule configured first is placed first. Once a rule is matched, the subsequent rules do not take effect, so a global permit rule must be placed at the end. The commands are as follows:

Command	Explanation
Global Mode	

[no] access-list <*5000-5099*> {deny|permit} ip {{<*source*> <*source-wildcard*>}|{host-source <*source-host-ip*>}|any-source} {{<*destination*> <*destination-wildcard*>}|{host-destination <*destination-host-ip*>}|any-destination}	Configures a source control rule. The rule does not take effect until it is applied to a specified port. The no command deletes the specified rule.

The third part is to apply the configured rules to specified ports.

Note: The configured rules occupy hardware table entries. Configuring too many rules can cause configuration failures because the hardware table becomes full, so use rules that are as simple as possible. The configuration commands are as follows:

Command	Explanation
Port Mode	
[no] ip multicast source-control access-group <*5000-5099*>	Applies source control rules to the port. The no command removes the configuration.

2．**Destination Control Configuration**

Like source control configuration, destination control configuration has three steps.

First, enable destination control globally. Because destination control must prevent unauthorized users from receiving multicast data, after global destination control is configured the switch does not broadcast the multicast data it receives. Therefore, avoid connecting two or more other Layer 3 switches in the same VLAN on a switch on which destination control is enabled. The configuration commands are as follows:

Command	Explanation
Global Mode	
[no] multicast destination-control（**required**）	Enables IPv4 and IPv6 destination control globally. The no command disables destination control globally. All other destination control configurations take effect only after it is enabled globally. Destination control rules are configured next, in a similar way.

Next, configure the multicast destination control profile rule list, using profile-id numbers 1-50.

Command	Explanation
Global Mode	
profile-id <1-50> {deny|permit} {{<source/M> }|{host-source <source-host-ip> (range <2-65535>|)}|any-source} {{<destination/M>}|{host-destination <destination-host-ip> (range <2-255>|)}|any-destination} **no profile-id <1-50>**	Configures a destination control profile rule. The no command deletes it.

Then configure destination control rules. They are similar to source control rules, except that ACL numbers 6000-7999 are used.

Command	Explanation
Global Mode	

[no] access-list <6000-7999> {{{add | delete} profile-id WORD} | {{deny|permit} (ip) {{<source/M> }|{host-source <source-host-ip> (range <2-65535>|)}|any-source} {{<destination/M>}|{host-destination <destination-host-ip> (range <2-255>|)}|any-destination}}	Configures a destination control rule. The rule does not take effect until it is applied to a source IP address, a VLAN-MAC, or a port. The no command deletes the specified rule.

Finally, apply the rules to a specified source IP address, source VLAN-MAC, or port. Note that, because of the conditions described above, these rules can be used only when IGMP Snooping is enabled globally. If IGMP Snooping is not enabled, only source IP rules can be used, under the IGMP protocol. The configuration commands are as follows:

Command	Explanation
Port Mode	
[no] ip multicast destination-control access-group *<6000-7999>*	Applies destination control rules to the port. The no command removes the configuration.
Global Mode	
[no] ip multicast destination-control *<1-4094>* *<macaddr>* access-group <6000-7999>	Applies destination control rules to the specified VLAN-MAC. The no command removes the configuration.
[no] ip multicast destination-control *<IPADDRESS/M>* access-group *<6000-7999>*	Applies destination control rules to the specified IP address/network mask. The no command removes the configuration.

3．**Multicast Strategy Configuration**

Multicast Strategy assigns a priority to specified multicast data to deliver the service level that specific users require. Note that multicast data receives this special treatment throughout the network only when it is transmitted on trunk ports. The configuration consists of a single command, which sets the priority for the specified multicast traffic:

Command	Explanation
Global Mode	
[no] ip multicast policy *<IPADDRESS/M>* *<IPADDRESS/M>* cos *<priority>*	Configures the multicast strategy by specifying a priority for sources and groups in the specified range. The priority range is <0-7>.

6.8.3 DCSCM Configuration Examples

1．Source Control

To prevent an edge switch from sending multicast data arbitrarily, configure the edge switch so that only the switch on port Ethernet1/0/5 is allowed to send multicast data, and only to group 225.1.2.3. The uplink switch connected to port Ethernet1/0/10 can send multicast data without any limit. The configuration is as follows:

```
EC(config)#access-list 5000 permit ip any host 225.1.2.3
EC(config)#access-list 5001 permit ip any any
EC(config)#ip multicast source-control
EC(config)#interface ethernet1/0/5
EC(Config-If-Ethernet1/0/5)#ip multicast source-control access-group 5000
EC(config)#interface ethernet1/0/10
EC(Config-If-Ethernet1/0/10)#ip multicast source-control access-group 5001
```

2．Destination Control

To prevent users in the 10.0.0.0/8 network segment from joining groups in 238.0.0.0/8, configure the following:

First, enable IGMP Snooping in the VLAN where the users are located (VLAN 2 in this example).

```
EC(config)#ip igmp snooping
EC(config)#ip igmp snooping vlan 2
```

Then configure the destination control access list and apply it to the specified IP addresses.

```
Switch(config)#access-list 6000 deny ip any 238.0.0.0 0.255.255.255
Switch(config)#access-list 6000 permit ip any any
Switch(config)#multicast destination-control
Switch(config)#ip multicast destination-control 10.0.0.0/8 access-group 6000
```

In this way, users in this network segment can join only groups outside 238.0.0.0/8.

Alternatively, configure the destination control access list by adding a profile.

```
Switch (config)#profile-id 1 deny ip any 238.0.0.0 0.255.255.255
Switch (config)#access-list 6000 add profile-id 1
Switch (config)#multicast destination-control
Switch (config)#ip multicast destination-control 10.0.0.0/8 access-group 6000
```

3．Multicast Strategy

Server 210.1.1.1 distributes important multicast data to group 239.1.2.3. Configure its access switch as follows:

```
Switch(config)#ip multicast policy 210.1.1.1/32 239.1.2.3/32 cos 4
```

In this way, the multicast stream has a priority of 4 when it passes through this switch to other switches. (This is a relatively high priority; the only higher one is normally used for protocol data. If a higher priority is set and there is too much multicast data, the switch protocols may behave abnormally.)

6.8.4 DCSCM Troubleshooting

The DCSCM module works similarly to ACLs, and problems are usually caused by improper configuration. Read the descriptions above carefully. If you still cannot determine the cause of the problem, send your configuration and the expected behavior to EnGenius technical support.

6.9 IGMP

6.9.1 Introduction to IGMP

IGMP (Internet Group Management Protocol) is the protocol in the TCP/IP protocol suite responsible for IP multicast group membership management. It is used to establish and maintain multicast group memberships between IP hosts and their neighboring multicast switches. IGMP does not propagate or maintain group membership information among multicast switches; that is done by the multicast routing protocols. All hosts participating in multicast must implement IGMP.

Hosts participating in IP multicast can join and leave multicast groups at any location, at any time, and without a limit on the total number of members. A multicast switch does not need to, and cannot, keep track of the memberships of all hosts. It only needs to know whether there are receivers (group members) of a multicast group on the network segment connected to each interface. A host only needs to keep track of the multicast groups it has joined.

IGMP is asymmetric between hosts and routers. A host responds to IGMP query messages from multicast switches with membership report messages. A switch sends membership query messages periodically and, based on the received responses, determines whether hosts on its subnet have joined specific groups. When it receives a report that a host has left a group, it sends a group-specific query (IGMP version 2) to determine whether any members of that group remain.

There are currently three versions of IGMP: IGMP version 1 (defined in RFC 1112), IGMP version 2 (defined in RFC 2236), and IGMP version 3 (defined in RFC 3376).

The main improvements of IGMP version 2 over version 1 are:

1. Querier election on shared network segments

A shared network segment is a segment with more than one multicast switch. Because all switches running IGMP on such a segment receive membership reports from hosts, only one switch needs to send membership query messages, so an election mechanism is needed to select one switch as the querier. In IGMP version 1, the querier is selected by the multicast routing protocol. IGMP version 2 improves this by specifying that when there is more than one multicast switch on the same network segment, the multicast switch with the lowest IP address is elected as the querier.

2. IGMP version 2 adds a Leave Group mechanism

In IGMP version 1, a host leaves a multicast group silently without notifying any multicast switch, so the multicast switch can detect that a member has left only when the group response times out. In version 2, when a host decides to leave a multicast group, it sends a Leave message if it was the host that responded to the latest membership query.

3. IGMP version 2 adds group-specific queries

In IGMP version 1, a query from a multicast switch covers all multicast groups on the network segment; this is called a general query. IGMP version 2 adds group-specific queries in addition to general queries. The destination IP address of a group-specific query is the IP address of the multicast group, and the group address field of the message is also the IP address of the multicast group. This prevents hosts that are members of other multicast groups from sending responses.

4. IGMP version 2 adds the Maximum Response Time field

IGMP version 2 adds the Maximum Response Time field to dynamically adjust how quickly hosts respond to group queries.

The main feature of version 3 is that it allows a host to choose to receive traffic from, or reject traffic from, specific sources, which is the basis of SSM (Source-Specific Multicast). For example, when a host sends a report of INCLUDE{10.1.1.1, 10.1.1.2} for group G, the host asks the router to forward traffic from 10.1.1.1 and 10.1.1.2; when a host sends a report of EXCLUDE{192.168.1.1} for group G, the host wants traffic from all sources of group G except 192.168.1.1. This is a major difference from earlier versions of IGMP.

The main improvements of IGMP version 3 over IGMP versions 1 and 2 are:

1. The state to be maintained is the group and source list, not only the group as in IGMPv2.
2. Interoperation with IGMPv1 and IGMPv2 is defined in the IGMPv3 state.
3. The IP service interface is modified to allow source lists to be specified.
4. The querier includes its Robustness Variable and Query Interval in query messages so that non-queriers can synchronize these variables.
5. The Max Response Time in query messages has an exponential range; the maximum value increases from 25.5 seconds in v2 to 53 minutes, which can be used on high-capacity links.
6. To improve robustness, hosts retransmit State-Change messages.
7. Additional data is defined to accommodate future extensions.
8. Reports are sent to 224.0.0.22 to help IGMP Snooping on Layer 2 switches.
9. A report can contain more than one group record, so that the complete current state can be reported in fewer packets.
10. Hosts no longer perform report suppression, which simplifies implementation and allows explicit membership tracking.
11. In query messages, the new router-side suppression process (S flag) modifies the robustness mechanism of IGMPv2.

6.9.2 IGMP Configuration Task List

1．Enable IGMP (required)

2．Configure IGMP sub-parameters (optional)

(1) Configure IGMP group parameters

1) Configure IGMP group filtering conditions

2) Configure the interface to join an IGMP group

3) Configure the interface to join an IGMP static group

(2) Configure IGMP query parameters

1) Configure the interval for sending IGMP query messages

2) Configure the maximum response time of IGMP query

3) Configure the IGMP query timeout

(3) Configure the IGMP version

3．Disable IGMP

1. **Enable IGMP**

There is no specific command for enabling IGMP on a Layer 3 switch. Enabling any multicast protocol on an interface automatically enables IGMP.

Command	Explanation
Global Mode	
ip dvmrp multicast-routing | ip pim multicast-routing	Enabling a multicast protocol globally is a prerequisite for enabling IGMP (required). The **no ip dvmrp multicast-routing | no ip pim multicast-routing** commands disable the multicast protocol and IGMP.
Command	Explanation
Interface Mode	
ip dvmrp enable| ip pim dense-mode | ip pim sparse-mode	Enables IGMP (required). The corresponding **no ip dvmrp enable| no ip pim dense-mode | no ip pim sparse-mode** commands disable IGMP.

2. **Configure IGMP sub-parameters**

(1) Configure IGMP group parameters

1) Configure IGMP group filtering conditions

2) Configure the interface to join an IGMP group

3) Configure the interface to join an IGMP static group

Command	Explanation
Interface Mode	
ip igmp access-group {<*acl_num* | *acl_name*>} **no ip igmp access-group**	Configures the IGMP group filtering conditions on the interface. The **no ip igmp access-group** command removes the filtering conditions.
ip igmp join-group <*A.B.C.D* > **no ip igmp join-group <*A.B.C.D* >**	Configures the interface to join an IGMP group. The **no ip igmp join-group <*A.B.C.D* >** command removes the interface from the group.
ip igmp static-group <*A.B.C.D* > **no ip igmp static-group <*A.B.C.D*>**	Configures the interface to join an IGMP static group. The **no ip igmp static-group <*A.B.C.D* >** command removes the interface from the group.

(2) Configure IGMP query parameters

1) Configure the interval for sending IGMP query messages
2) Configure the maximum response time of IGMP query
3) Configure the IGMP query timeout

Command	Explanation
Interface Mode	
ip igmp query-interval <*time_val*> **no ip igmp query-interval**	Configures the interval for sending periodic IGMP query messages. The **no ip igmp query-interval** command restores the default value.
ip igmp query-max-response-time <*time_val*> **no ip igmp query-max-response-time**	Configures the maximum response time for IGMP queries on the interface. The **no ip igmp query-max-response-time** command restores the default value.
ip igmp query-timeout <*time_val*> **no ip igmp query-timeout**	Configures the IGMP query timeout on the interface. The **no ip igmp query-timeout** command restores the default value.

(3) Configure the IGMP version

Command	Explanation
Interface Mode	
ip igmp version <*version*> **no ip igmp version**	Configures the IGMP version on the interface. The **no ip igmp version** command restores the default value.

3. **Disable IGMP**

Command	Explanation
Interface Mode	
no ip dvmrp | no ip pim dense-mode | no ip pim sparse-mode | no ip dvmrp multicast-routing | no ip pim multicast-routing	Disables IGMP.

6.9.3 IGMP Configuration Examples

As shown in the following figure, add the Ethernet ports of Switch A and Switch B to the corresponding VLANs, and enable PIM-DM on each VLAN interface.

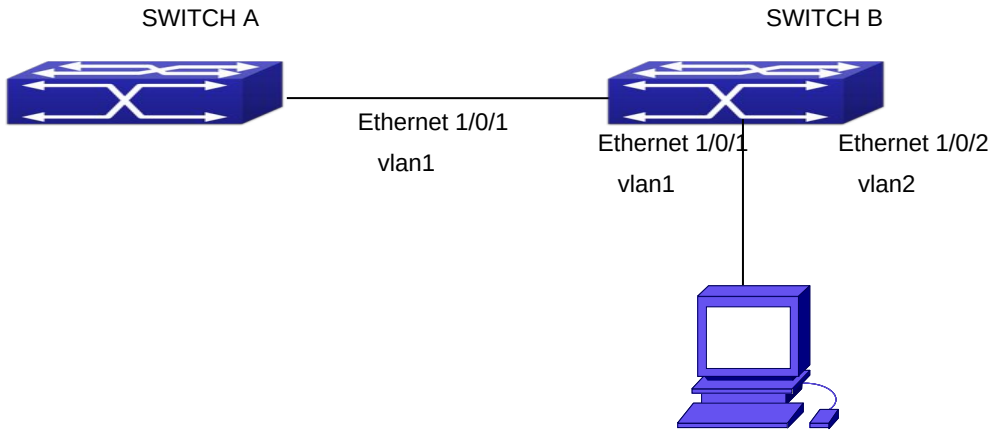


Figure 6-10 IGMP Network Topology

The configuration procedure for SwitchA and SwitchB is as follows:

(1) Configure SwitchA:

```
Switch(config)#ip pim multicast-routing
Switch (config)#interface vlan 1
Switch(Config-if-Vlan1)#ip address 12.1.1.1 255.255.255.0
Switch(Config-if-Vlan1)#ip pim dense-mode
```

(2) Configure SwitchB:

```
Switch(config)#ip pim multicast-routing
Switch(config)#interface vlan1
Switch(Config-if-Vlan1)#ip address 12.1.1.2 255.255.255.0
Switch(Config-if-Vlan1)#ip pim dense-mode
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan2
Switch(Config-if-Vlan1)#ip address 20.1.1.1 255.255.255.0
Switch(Config-if-Vlan2)#ip pim dense-mode
Switch(Config-if-Vlan2)#ip igmp version 3
```

6.9.4 IGMP Troubleshooting

When IGMP is configured and used, it may fail to operate normally because of physical connection problems or incorrect configuration. Check the following:

- First, make sure the physical connections are correct.
- Next, make sure the interface and line protocol are up (use the show interface command).
- Then make sure a multicast protocol is enabled on the interface.
- Multicast protocols use unicast routing for the RPF check, so make sure unicast routing is correct first.

6.10 IGMP Snooping

6.10.1 Introduction to IGMP Snooping

IGMP (Internet Group Management Protocol) is a protocol used in IP multicast. Multicast-enabled network devices (such as routers) use IGMP to query host membership, and hosts joining a multicast group use it to inform the router that they want to receive packets for a certain multicast address. All these operations are done by exchanging IGMP messages. The router sends IGMP host membership query messages to the all-hosts multicast address (224.0.0.1). A host that wants to join a multicast group replies to the multicast address of that group with an IGMP host membership report message.

IGMP Snooping is also called IGMP listening. With IGMP Snooping, the switch prevents multicast traffic from flooding and forwards multicast traffic only to the ports associated with multicast devices. The switch listens to the IGMP messages exchanged between the multicast router and hosts, maintains a multicast group forwarding table based on what it learns, and forwards multicast packets according to this table.

The switch supports IGMP Snooping and can also send queries itself, so it can be used in IP multicast networks.

6.10.2 IGMP Snooping Configuration Task List

1. Enable IGMP Snooping

2. Configure IGMP Snooping

1. Enable IGMP Snooping

Command	Explanation
Global Mode	
ip igmp snooping **no ip igmp snooping**	Enables IGMP Snooping. The no command disables IGMP Snooping.

2. Configure IGMP Snooping

Command	Explanation
Global Mode	
ip igmp snooping vlan <*vlan-id*> **no ip igmp snooping vlan <*vlan-id*>**	Enables IGMP Snooping for the specified VLAN. The no command disables IGMP Snooping for the specified VLAN.
ip igmp snooping proxy **no ip igmp snooping proxy**	Enables the IGMP Snooping proxy function. The no command disables the function.
ip igmp snooping vlan < *vlan-id* > limit {group <*g_limit*> | source <*s_limit*>} **no ip igmp snooping vlan < *vlan-id* > limit**	Configures the maximum number of groups in the VLAN and the maximum number of sources in each group. The “no **ip igmp snooping vlan <*vlan-id*> limit”** command removes this configuration.
ip igmp snooping vlan <1-4094> interface (ethernet | port-channel|) IFNAME limit {group <1-65535>| source <1-65535>} strategy (replace | drop) **no ip igmp snooping vlan <1-4094> interface (ethernet | port-channel|) IFNAME limit group source strategy**	Configures the maximum number of groups that can be joined on the IGMP Snooping port and the maximum number of sources in each group, as well as the action taken when the limit is reached ("replace" or "drop"). The no command removes the limits.
ip igmp snooping vlan <*vlan-id*> l2-general-querier **no ip igmp snooping vlan <*vlan-id*> l2-general-querier**	Configures the VLAN as a Layer 2 general querier. It is recommended to configure one Layer 2 general querier on each segment. The “no **ip igmp snooping vlan <*vlan-id*> l2-general-querier”command** removes this configuration.
ip igmp snooping vlan <*vlan-id*> l2-general-querier-version <version>	Configures the version of the general queries sent by the Layer 2 general querier.
ip igmp snooping vlan <*vlan-id*> l2-general-querier-source <source>	Configures the source address of the general queries sent by the Layer 2 general querier.
ip igmp snooping vlan <*vlan-id*> mrouter-port interface <*interface –name*> **no ip igmp snooping vlan <*vlan-id*> mrouter-port interface <*interface –name*>**	Configures a static mrouter port for the VLAN. The no command removes this configuration.
ip igmp snooping vlan <*vlan-id*> mrouter-port learnpim **no ip igmp snooping vlan <*vlan-id*> mrouter-port learnpim**	Enables the specified VLAN to learn mrouter ports from PIM packets. The no command disables the function.
ip igmp snooping vlan <*vlan-id*> mrpt <*value* > **no ip igmp snooping vlan <*vlan-id*> mrpt**	Configures the aging time of the mrouter port. The “no **ip igmp snooping vlan <*vlan-id*> mrpt”** command restores the default value.
ip igmp snooping vlan <*vlan-id*> query-	Configures the query interval. The “no **ip**

interval <*value*> **no ip igmp snooping vlan <*vlan-id*> query-interval**	**igmp snooping vlan <*vlan-id*> query-interval"** command restores the default value.
ip igmp snooping vlan <*vlan-id*> immediately-leave **no ip igmp snooping vlan <*vlan-id*> immediately-leave**	Enables IGMP fast leave for the specified VLAN. The "no **ip igmp snooping vlan <*vlan-id*> immediate-leave"** command disables IGMP fast leave.
ip igmp snooping vlan <*vlan-id*> query-mrsp <*value*> **no ip igmp snooping vlan <*vlan-id*> query-mrsp**	Configures the maximum query response time. The "no **ip igmp snooping vlan <*vlan-id*> query-mrsp"** command restores the default value.
ip igmp snooping vlan <*vlan-id*> query-robustness <*value*> **no ip igmp snooping vlan <*vlan-id*> query-robustness**	Configures the query robustness variable. The "no **ip igmp snooping vlan <*vlan-id*> query-robustness"** command restores the default value.
ip igmp snooping vlan <*vlan-id*> suppression-query-time <*value*> **no ip igmp snooping vlan <*vlan-id*> suppression-query-time**	Configures the query suppression time. The "no **ip igmp snooping vlan <*vlan-id*> suppression-query-time"** command restores the default value.
ip igmp snooping vlan <*vlan-id*> static-group <*A.B.C.D*> [source <*A.B.C.D*>] interface [ethernet | port-channel] <*IFNAME*> **no ip igmp snooping vlan <*vlan-id*> static-group <*A.B.C.D*> [source <*A.B.C.D*>] interface [ethernet | port-channel] <*IFNAME*>**	Configures a static group on the specified port of the VLAN. The no command removes this configuration.
ip igmp snooping vlan <*vlan-id*> report source-address <*A.B.C.D*> **no ip igmp snooping vlan <*vlan-id*> report source-address**	Configures the source address of forwarded IGMP packets. The no command removes the configured source address.
ip igmp snooping vlan <*vlan-id*> specific-query-mrsp <value> **no ip igmp snooping vlan <*vlan-id*> specific-query-mrspt**	Configures the maximum query response time for group-specific or source-specific queries. The no command restores the default value.

6.10.3 IGMP Snooping Examples

Scenario 1: IGMP Snooping

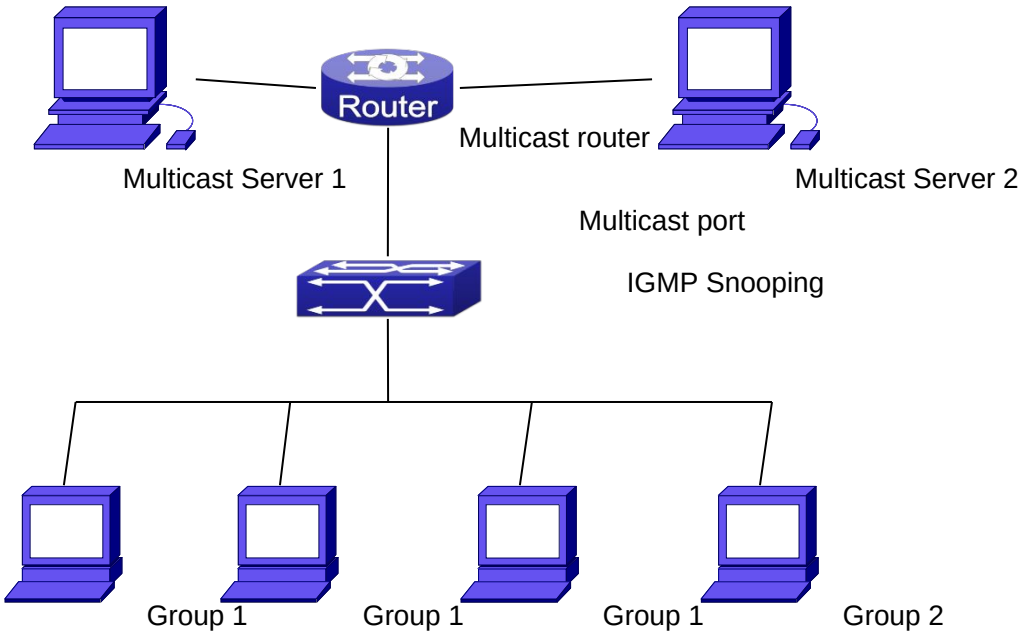


Figure 6-11 Enabling IGMP Snooping

Example: As shown in the figure above, VLAN 100 is configured on the switch and includes ports 1, 2, 6, 10, and 12. Four hosts are connected to ports 2, 6, 10, and 12, and the multicast router is connected to port 1. IGMP Snooping is disabled by default both globally and in VLANs. To enable IGMP Snooping in VLAN 100, first enable IGMP Snooping globally in Global Mode and in VLAN 100, and then set port 1 of VLAN 100 as the mrouter port.

The configuration steps are as follows:

```
Switch(config)#ip igmp snooping
Switch(config)#ip igmp snooping vlan 100
Switch(config)#ip igmp snooping vlan 100 mrouter interface ethernet 1/0/1
```

Multicast Configuration

Suppose the multicast server provides two programs using multicast addresses Group1 and Group2. Three of the four hosts running multicast applications, on ports 2, 6, and 10, play program 1, while the host on port 12 plays program 2.

IGMP Snooping listening result:

The multicast table built by IGMP Snooping in VLAN 100 contains ports 1, 2, 6, and 10 for Group1 and ports 1 and 12 for Group2.

All four hosts receive the program of their choice: ports 2, 6, and 10 do not receive program 2 traffic, and port 12 does not receive program 1 traffic.

Scenario 2: L2-general-querier

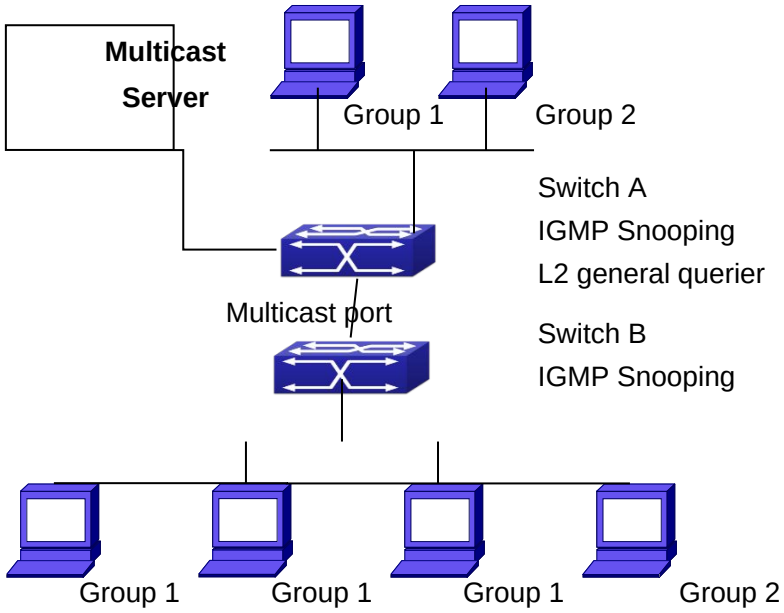


Figure 6-12 Switches as IGMP Queriers

The configuration of Switch2 is the same as that of the switch in Scenario 1, and SwitchA replaces the multicast router of Scenario 1. Assume VLAN 60 is configured on SwitchA and includes ports 1, 2, 10, and 12. Port 1 connects to the multicast server, and port 2 connects to Switch2. To send queries at regular intervals, enable IGMP Snooping globally and the Layer 2 general querier in VLAN 60.

The configuration steps are as follows:

```
SwitchA#config
SwitchA(config)#ip igmp snooping
SwitchA(config)#ip igmp snooping vlan 60
SwitchA(config)#ip igmp snooping vlan 60 L2-general-querier
SwitchB#config
SwitchB(config)#ip igmp snooping
SwitchB(config)#ip igmp snooping vlan 100
SwitchB(config)#ip igmp snooping vlan 100 mrouter interface ethernet 1/0/1
```

Multicast Configuration

Same as Scenario 1.

IGMP Snooping listening result:

Similar to scenario 1

Scenario 3: Working with Layer 3 multicast protocols

The switch used in Scenario 1 is replaced by a router, and the rest of the configuration remains the same. The multicast and IGMP Snooping configurations are the same as in Scenario 1. Configure PIM-SM on the router and enable PIM-SM on VLAN 100 (use the same PIM mode as the connected multicast router).

The configuration is as follows:

```
switch#config
switch(config)#ip pim multicast-routing
switch(config)#interface vlan 100
switch(config-if-vlan100)#ip pim sparse-mode
```

IGMP Snooping does not install entries when a Layer 3 multicast protocol is enabled. It performs only the following tasks:

- ☞ Removes the Layer 2 multicast entries.
- ☞ Provides query functions to Layer 3, with VLAN, S, and G as parameters.
- ☞ Resumes installing Layer 2 multicast entries when Layer 3 IGMP is disabled.

The ports can be determined by looking up the Layer 3 IPMC entries. This ensures that IGMP Snooping works together with Layer 3 multicast protocols.

6.10.4 IGMP Snooping Troubleshooting

IGMP Snooping may fail to run properly because of physical connection problems or configuration errors. Check the following:

- ☞ Make sure the physical connections are correct.
- ☞ Enable IGMP Snooping in Global Mode (use **ip igmp snooping**).
- ☞ Enable IGMP Snooping on the VLAN in Global Mode (use **ip igmp snooping vlan <vlan-id>**).
- ☞ Make sure one VLAN in the same segment is configured as the Layer 2 general querier, or make sure a static mrouter port is configured.
- ☞ Use the **show ip igmp snooping vlan <vid>** command to check the IGMP Snooping information.

6.11 IGMP Proxy

6.11.1 Introduction to IGMP Proxy

IGMP/MLD proxy, introduced in RFC 4605, is a simplified multicast protocol that runs on edge devices. Edge devices running IGMP/MLD proxy do not need to run complex multicast routing protocols such as PIM or DVMRP; instead, they interwork with the multicast-enabled network through IGMP/MLD proxy. This simplifies the implementation of multicast on edge devices.

IGMP/MLD proxy works between the multicast router and the clients, acting as both a multicast host and a multicast router. Upstream and downstream ports must be specified in the IGMP/MLD proxy configuration. The host protocol runs on upstream ports, and the router protocol runs on downstream ports. The switch collects the Join and Leave messages received on downstream ports and forwards them to the multicast router through upstream ports.

IGMP Proxy cannot be configured together with PIM or DVMRP.

6.11.2 IGMP Proxy Configuration Task List

1. Enable IGMP Proxy
2. Configure IGMP Proxy upstream and downstream ports on different interfaces
3. Configure IGMP Proxy parameters

1. Enable IGMP Proxy

Command	Explanation
Global Mode	
ip igmp proxy **no ip igmp proxy**	Enables IGMP Proxy. The **no ip igmp proxy** command disables this function.

2. Configure IGMP Proxy upstream and downstream ports on different interfaces

Command	Explanation
Interface Mode	

ip igmp proxy upstream **no ip igmp proxy upstream**	Enables the IGMP Proxy upstream function. The **no ip igmp proxy upstream** command disables this function.
ip igmp proxy downstream **no ip igmp proxy downstream**	Enables the IGMP Proxy downstream function. The **no ip igmp proxy downstream** command disables this function.

3. Configure IGMP Proxy parameters

Command	Explanation
Global Mode	
ip igmp proxy limit {group *<1-500>*| source *<1-500>*} **no ip igmp proxy limit**	Configures the maximum number of groups that upstream ports can join and the maximum number of sources in a single group. The no command restores the default values.
ip igmp proxy unsolicited-report interval *<1-5>* **no ip igmp proxy unsolicited-report interval**	Configures the interval at which upstream ports send unsolicited reports. The no command restores the default configuration.
ip igmp proxy unsolicited-report robustness *<2-10>* **no ip igmp proxy unsolicited-report robustness**	Configures the number of times upstream ports retransmit unsolicited reports. The no command restores the default value.
ip igmp proxy aggregate **no ip igmp proxy aggregate**	Enables non-querier downstream ports to aggregate IGMP operations. The no command restores the default configuration.
ip multicast ssm range *<1-99>* **ip multicast ssm default** **no ip mulitcast ssm**	Configures the address range of IGMP Proxy SSM multicast groups. The no command removes the configuration.
ip igmp proxy multicast-source **no ip igmp proxy multicast-source**	Configures the port as a downstream port for multicast sources. The no command disables the configuration.

6.11.3 IGMP Proxy Examples

Example 1: IGMP Proxy

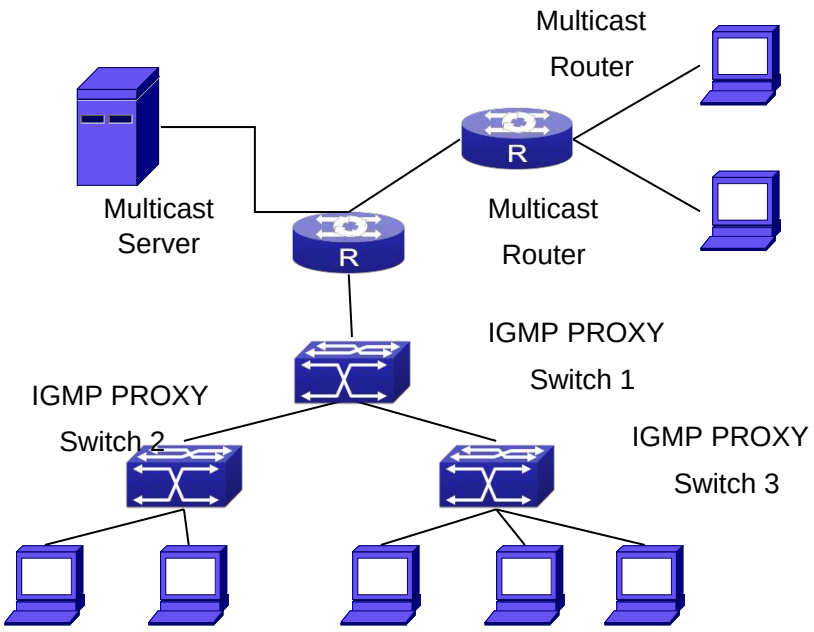


Figure 6-13 IGMP Proxy Topology

As shown in the figure above, the switches act as IGMP Proxy devices in a tree topology. Each switch distributes the multicast data flow received on its upstream port to its downstream ports, while IGMP membership reports flow from downstream ports to upstream ports. Each of the three IGMP Proxy-enabled switches in the tree has one port connected toward the multicast router and at least one port connected to hosts or to the upstream ports of other IGMP Proxy-enabled switches.

The configuration steps are as follows:

```
Switch#config
Switch(config)#ip igmp proxy
Switch(Config)#interface vlan 1
Switch(Config-if-Vlan1)#ip igmp proxy upstream
Switch(Config)#interface vlan 2
Switch(Config-if-Vlan2)#ip igmp proxy downstream
```

Multicast Configuration:

Suppose the multicast server provides a program through 224.1.1.1, and some hosts at the edge of the network subscribe to it. The IGMP multicast members report to the downstream ports of the IGMP Proxy-enabled Switch 2 and Switch 3. Switch 2 and Switch 3 aggregate the group membership information and send it through their upstream ports. Switch 1 receives the group membership information on its downstream ports, forwards it to the multicast router through its upstream port, and delivers the multicast data flow through its downstream ports.

Example2: IGMP Proxy for multicast sources on downstream ports

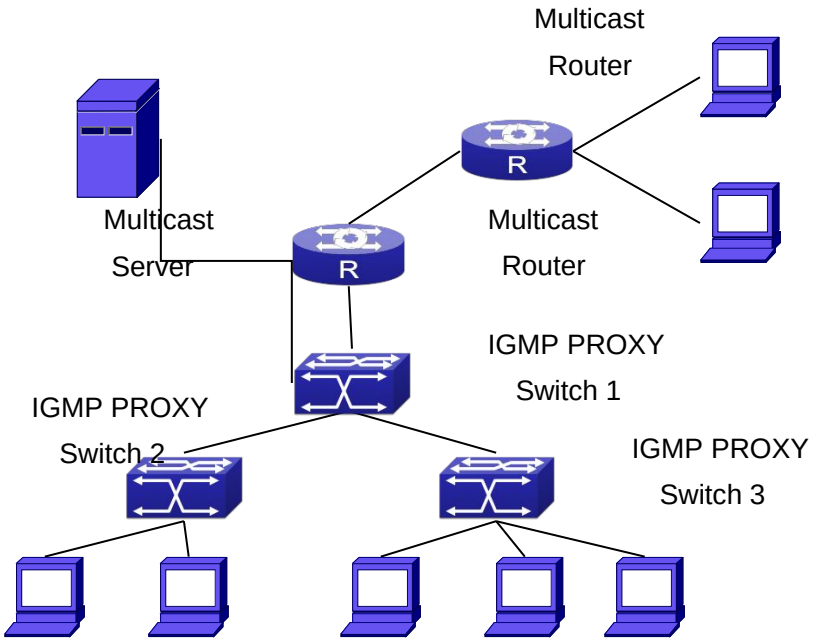


Figure 6-14 IGMP Proxy for Multicast Sources on Downstream Ports

As shown in the figure above, IGMP Proxy-enabled switches are connected to the network in a tree topology. The multicast source server connects to a downstream port of Switch1, and the multicast data flow is distributed through the upstream port and the other downstream ports. Each of the three IGMP Proxy-enabled switches in the tree has one port connected toward the multicast router and at least one port connected to hosts or to the upstream ports of other IGMP Proxy-enabled switches.

The configuration steps are as follows:

IGMP Proxy Switch1 configuration:

```
Switch#config
Switch(config)#ip igmp proxy
Switch(Config)#interface vlan 1
Switch(Config-if-Vlan1)#ip igmp proxy upstream
Switch(Config)#interface vlan 2
Switch(Config-if-Vlan2)#ip igmp proxy downstream
Switch(Config-if-Vlan2)#ip igmp proxy multicast-source
```

Router1 configuration:

```
Switch#config
Switch(config)#ip pim multicast
Switch(Config)#interface vlan 1
Switch(Config-if-Vlan1)#ip pim sparse-mode
Switch(Config-if-Vlan1)#ip pim bsr-border
```

Multicast Configuration:

Suppose the server provides a program through multicast address 224.1.1.1, and some hosts at the edge of the network subscribe to it. The hosts report their IGMP group memberships to Switch 2 and Switch 3 through downstream ports. Switch 2 and Switch 3 aggregate the reports and forward them to Switch 1, which forwards the information to the multicast router. When the multicast data flow arrives, the IGMP Proxy-enabled switches distribute it through upstream and downstream ports. When the multicast router receives the multicast data flow from the IGMP Proxy, it treats the multicast source as directly connected and determines the DR and originator roles. The multicast data flow is then distributed according to PIM.

6.11.4 IGMP Proxy Troubleshooting

IGMP Proxy may fail to run properly because of physical connection problems or configuration errors. Check the following:

- ☞ Make sure the physical connections are correct.
- ☞ Enable IGMP Proxy in Global Mode (use **ip igmp proxy**).
- ☞ Make sure one upstream port and at least one downstream port are configured in Interface Mode (use **ip igmp proxy upstream**, **ip igmp proxy downstream**).
- ☞ Use the **show ip igmp proxy** command to check whether the IGMP Proxy information is correct.

If the IGMP Proxy problem persists, run debug IGMP Proxy and other debug commands, copy the debug messages for three minutes, and contact EnGenius technical support.

6.12 Multicast VLAN

6.12.1 Introduction to Multicast VLAN

With the conventional multicast subscription method, when users in different VLANs request the same multicast traffic, each VLAN carries its own copy of the traffic, which wastes a great deal of bandwidth. By configuring a multicast VLAN, adding switch ports to it, and enabling IGMP Snooping/MLD Snooping, users in different VLANs share the same multicast VLAN. The multicast traffic exists only within the multicast VLAN, which saves bandwidth. Because the multicast VLAN is completely separated from the user VLANs, both security and bandwidth requirements are met. After the multicast VLAN is configured, multicast traffic is sent to users continuously.

6.12.2 Multicast VLAN Configuration Task List

1. Enable the multicast VLAN
2. Configure IGMP Snooping
3. Configure MLD Snooping

1. Enable the multicast VLAN

Command	Explanation
VLAN Mode	
multicast-vlan **no multicast-vlan**	Enables the multicast VLAN function on the VLAN. The **no multicast-vlan** command disables the multicast VLAN function on the VLAN.
multicast-vlan association <*vlan-list*> **no multicast-vlan association <*vlan-list*>**	Associates the multicast VLAN with one or more VLANs. The no command removes the association between the specified VLANs and the multicast VLAN.
multicast-vlan association interface (ethernet | port-channel|) IFNAME **no multicast-vlan association interface (ethernet | port-channel|) IFNAME**	Associates the specified port with the multicast VLAN so that the port can receive the multicast traffic. The no command removes the association between the port and the multicast VLAN.
multicast-vlan mode {dynamic| compatible} **no multicast-vlan mode {dynamic| compatible}**	Configures the multicast VLAN mode (dynamic or compatible). The no command removes the mode configuration.

2. Configure IGMP Snooping

Command	Explanation

Global Mode	
ip igmp snooping vlan <*vlan-id*> **no ip igmp snooping vlan <*vlan-id*>**	Enables IGMP Snooping on the multicast VLAN. The no command disables IGMP Snooping on the multicast VLAN.
ip igmp snooping **no ip igmp snooping**	Enables IGMP Snooping. The no command disables IGMP Snooping.

3. Configure MLD Snooping

ipv6 mld snooping vlan <*vlan-id*> **no ipv6 mld snooping vlan <*vlan-id*>**	Enables MLD Snooping on the multicast VLAN. The no command disables MLD Snooping on the multicast VLAN.
ipv6 mld snooping **no ipv6 mld snooping**	Enables MLD Snooping. The no command disables MLD Snooping.

6.12.3 Multicast VLAN Examples

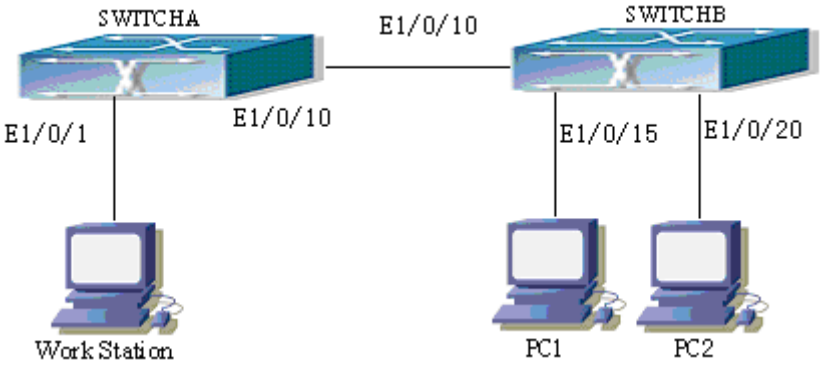


Figure 6-15 Multicast VLAN Configuration

As shown in the figure, the multicast server is connected to Layer 3 switch SwitchA through port 1/0/1, which belongs to VLAN 10 of the switch. SwitchA is connected to the Layer 2 switch through port 1/0/10, which is configured as a trunk port. On SwitchB, VLAN 100 is configured to contain port 1/0/15, and VLAN 101 to contain port 1/0/20. PC1 and PC2 are connected to ports 1/0/15 and 1/0/20, respectively. SwitchB is connected to SwitchA through port 1/0/10, which is configured as a trunk port. VLAN 20 is the multicast VLAN. With the multicast VLAN configured, PC1 and PC2 receive the multicast data from the multicast VLAN.

The following configuration assumes that the IP address of the switch has been configured and all devices are connected correctly.

Configuration procedure

```
SwitchA#config
SwitchA(config)#vlan 10
SwitchA(config-vlan10)#switchport access ethernet 1/0/1
SwitchA(config-vlan10)exit
SwitchA(config)#interface vlan 10
Switch(Config-if-Vlan10)#ip pim dense-mode
Switch(Config-if-Vlan10)#exit
SwitchA(config)#vlan 20
SwitchA(config-vlan20)#exit
SwitchA(config)#interface vlan 20
SwitchA(Config-if-Vlan20)#ip pim dense-mode
SwitchA(Config-if-Vlan20)#exit
SwitchA(config)#ip pim multicast
SwitchA(config)# interface ethernet1/0/10
SwitchA(Config-If-Ethernet1/0/10)switchport mode trunk
SwitchB#config
SwitchB(config)#vlan 100
```

```
SwitchB(config-vlan100)#Switchport access ethernet 1/0/15
SwitchB(config-vlan100)exit
SwitchB(config)#vlan 101
SwitchB(config-vlan101)#Switchport access ethernet 1/0/20
SwitchB(config-vlan101)exit
SwitchB(config)# interface ethernet 1/0/10
SwitchB(Config-If-Ethernet1/0/10)#switchport mode trunk
SwitchB(Config-If-Ethernet1/0/10)#exit
SwitchB(config)#vlan 20
SwitchB(config-vlan20)#multicast-vlan
SwitchB(config-vlan20)#multicast-vlan association 100,101
SwitchB(config-vlan20)#exit
SwitchB(config)#ip igmp snooping
SwitchB(config)#ip igmp snooping vlan 20
```

When the multicast VLAN is used for IPv6 multicast, the configuration is the same as for IPv4, except that MLD Snooping is used instead; therefore, no separate example is given.

Chapter 7 Security Function Configuration

7.1 ACL

7.1.1 Introduction to ACL

ACL (Access Control List) is an IP packet filtering mechanism used in switches. It controls network traffic by permitting or denying access through the switch, effectively safeguarding network security. A set of rules can be defined based on specific packet information; each rule specifies the action, "permit" or "deny", for packets that match it. These rules can be applied to the ingress direction of switch ports, so that the data streams in the ingress direction of the specified ports must comply with the assigned ACL rules.

7.1.1.1 Access List

An access list is a sequential collection of conditions that make up a rule set. Each rule consists of filter information and the action taken when the rule is matched. The information in a rule is an effective combination of conditions such as source IP address, destination IP address, IP protocol number, TCP port, and UDP port. Access lists can be categorized by the following criteria:

- By filter information: IP access list (Layer 3 or higher information), MAC access list (Layer 2 information), and MAC-IP access list (Layer 2 and Layer 3 or higher information).
- By configuration complexity: standard and extended. The extended type allows more specific filtering.
- By naming: numbered and named.

An ACL description should cover all three aspects above.

7.1.1.2 Access Group

After access lists are created, they can be applied to the ingress traffic of any port. An access group is the binding of an access list to the ingress direction of a specific port. After an access group is created, all packets received on the port are compared with the access list rules to decide whether to permit or deny them.

The current firmware supports only ingress ACL configuration.

7.1.1.3 Access List Action and Global Default Action

There are two access list actions and default actions: "permit" and "deny". The following rules apply:

- An access list can contain several rules. A packet is compared with the rules in order, from the first rule until the first matching rule; the remaining rules are not processed. The global default action applies only to IP packets in the ingress direction of the ports.
- The global default action applies only when packet filtering is enabled on a port and either no ACL is bound to the port or no bound ACL matches.

7.1.2 ACL Configuration Task List

ACL configuration task sequence:

1. Configure an access list

(1) Configure a numbered standard IP access list

(2) Configure a numbered extended IP access list

(3) Configure a named standard IP access list

a) Create a named standard IP access list
b) Specify multiple "permit" or "deny" rules
c) Exit ACL Configuration Mode

(4) Configure a named extended IP access list

a) Create a named extended IP access list
b) Specify multiple "permit" or "deny" rules
c) Exit ACL Configuration Mode

(5) Configure a numbered standard MAC access list

(6) Configure a numbered extended MAC access list

(7) Configure a named extended MAC access list

a) Create a named extended MAC access list
b) Specify multiple "permit" or "deny" rules
c) Exit ACL Configuration Mode

(8) Configure a numbered extended MAC-IP access list

(9) Configure a named extended MAC-IP access list

a) Create a named extended MAC-IP access list
b) Specify multiple "permit" or "deny" rules
c) Exit MAC-IP Configuration Mode

(10) Configure a numbered standard IPv6 access list

(11) Configure a numbered extended IPv6 access list

(12) Configure a named standard IPv6 access list

a) Create a named standard IPv6 access list
b) Specify multiple permit or deny rule entries
c) Exit ACL Configuration Mode

(13) Configure a named extended IPv6 access list

a) Create a named extended IPv6 access list
b) Specify multiple permit or deny rule entries
c) Exit ACL Configuration Mode

2. Configure the packet filtering function

(1) Enable the global packet filtering function

(2) Configure the default action

3. Configure the time range function

(1) Create a time range
(2) Configure a periodic time range
(3) Configure an absolute time range

4. Bind an access list to the ingress direction of the specified port

5. Clear the filtering information of the specified port

1. Configure an access list

(1) Configure a numbered standard IP access list

Command	Explanation
Global Mode	

access-list <*num*> {deny | permit} {{<*sIpAddr*> <*sMask*>} | any-source | {host-source <*sIpAddr*>}} **no access-list <*num*>**	Creates a numbered standard IP access list. If the access list already exists, a rule is added to it. The **no access-list <*num*>** command deletes a numbered standard IP access list.

(2) Configure a numbered extended IP access list

Command	Explanation
Global Mode	
access-list <*num*> {deny | permit} icmp {{<*sIpAddr*> <*sMask*>} | any-source | {host-source <*sIpAddr*>}} {{<*dIpAddr*> <*dMask*>} | any-destination | {host-destination <*dIpAddr*>}} [<*icmp-type*> [<*icmp-code*>]] [precedence <*prec*>] [tos <*tos*>][time-range<*time-range-name*>]	Creates a numbered extended IP access rule for ICMP. If the numbered extended access list does not exist, an access list is created with this number.
access-list <*num*> {deny | permit} igmp {{<*sIpAddr*> <*sMask*>} | any-source | {host-source <*sIpAddr*>}} {{<*dIpAddr*> <*dMask*>} | any-destination | {host-destination <*dIpAddr*>}} [<*igmp-type*>] [precedence <*prec*>] [tos <*tos*>][time-range<*time-range-name*>]	Creates a numbered extended IP access rule for IGMP. If the numbered extended access list does not exist, an access list is created with this number.
access-list <*num*> {deny | permit} tcp {{<*sIpAddr*> <*sMask*>} | any-source | {host-source <*sIpAddr*>}} [s-port {<*sPort*> | range <*sPortMin*> <*sPortMax*>}] {{<*dIpAddr*> <*dMask*>} | any-destination | {host-destination <*dIpAddr*>}} [d-port {<*dPort*> | range <*dPortMin*> <*dPortMax*>}] [ack+fin+psh+rst+urg+syn] [precedence <*prec*>] [tos <*tos*>][time-range<*time-range-name*>]	Creates a numbered extended IP access rule for TCP. If the numbered extended access list does not exist, an access list is created with this number.
access-list <*num*> {deny | permit} udp {{<*sIpAddr*> <*sMask*>} | any-source | {host-source <*sIpAddr*>}} [s-port {<*sPort*> | range <*sPortMin*> <*sPortMax*>}] {{<*dIpAddr*> <*dMask*>} | any-destination | {host-destination <*dIpAddr*>}} [d-port {<*dPort*> | range <*dPortMin*> <*dPortMax*>}] [precedence <*prec*>] [tos <*tos*>][time-range<*time-range-name*>]	Creates a numbered extended IP access rule for UDP. If the numbered extended access list does not exist, an access list is created with this number.
access-list <*num*> {deny | permit} {eigrp | gre | igrp | ipinip | ip | ospf | <*protocol-num*>} {{<*sIpAddr*> <*sMask*>} | any-source | {host-source <*sIpAddr*>}} {{<*dIpAddr*> <*dMask*>} | any-destination | {host-destination <*dIpAddr*>}} [precedence <*prec*>] [tos <*tos*>][time-range<*time-range-name*>]	Creates a numbered extended IP access rule for another specific IP protocol or for all IP protocols. If the numbered extended access list does not exist, an access list is created with this number.
no access-list <*num*>	Deletes a numbered extended IP access list.

(3) Configure a named standard IP access list

a. Create a named standard IP access list

Command	Explanation
Global Mode	
ip access-list standard <*name*> **no ip access-list standard <*name*>**	Creates a named standard IP access list. The **no ip access-list standard <*name*>** command deletes the named standard IP access list.

b. Specify multiple "permit" or "deny" rules

Command	Explanation

Standard IP ACL Mode	
[no] {deny | permit} {{<*sIpAddr*> <*sMask*>} | any-source | {host-source <*sIpAddr*>}}	Creates a named standard IP access rule. The no command deletes the named standard IP access rule.

c. Exit named standard IP ACL configuration mode

Command	Explanation
Standard IP ACL Mode	
exit	Exits named standard IP ACL configuration mode.

(4) Configure a named extended IP access list

a. Create a named extended IP access list

Command	Explanation
Global Mode	
ip access-list extended <*name*> **no ip access-list extended <*name*>**	Creates a named extended IP access list. The **no ip access-list extended <name>** command deletes the named extended IP access list.

b. Specify multiple "permit" or "deny" rules

Command	Explanation
Extended IP ACL Mode	
[no] {deny | permit} icmp {{<*sIpAddr*> <*sMask*>} | any-source | {host-source <*sIpAddr*>}} {{<*dIpAddr*> <*dMask*>} | any-destination | {host-destination <*dIpAddr*>}} [<*icmp-type*> [<*icmp-code*>]] [precedence <*prec*>] [tos <*tos*>][time-range<*time-range-name*>]	Creates a named extended IP access rule for ICMP. The no command deletes this named extended IP access rule.
[no] {deny | permit} igmp {{<*sIpAddr*> <*sMask*>} | any-source | {host-source <*sIpAddr*>}} {{<*dIpAddr*> <*dMask*>} | any-destination | {host-destination <*dIpAddr*>}} [<*igmp-type*>] [precedence <*prec*>] [tos <*tos*>][time-range<*time-range-name*>]	Creates a named extended IP access rule for IGMP. The no command deletes this named extended IP access rule.
[no] {deny | permit} tcp {{<*sIpAddr*> <*sMask*>} | any-source | {host-source <*sIpAddr*>}} [s-port {<*sPort*> | range <*sPortMin*> <*sPortMax*>}] {{<*dIpAddr*> <*dMask*>} | any-destination | {host-destination <*dIpAddr*>}} [d-port {<*dPort*> | range <*dPortMin*> <*dPortMax*>}] [ack+fin+psh+rst+urg+syn] [precedence <*prec*>] [tos <*tos*>][time-range<*time-range-name*>]	Creates a named extended IP access rule for TCP. The no command deletes this named extended IP access rule.
[no] {deny | permit} udp {{<*sIpAddr*> <*sMask*>} | any-source | {host-source <*sIpAddr*>}} [s-port {<*sPort*> | range <*sPortMin*> <*sPortMax*>}] {{<*dIpAddr*> <*dMask*>} | any-destination | {host-destination <*dIpAddr*>}} [d-port {<*dPort*> | range <*dPortMin*> <*dPortMax*>}] [precedence <*prec*>] [tos <*tos*>][time-range<*time-range-name*>]	Creates a named extended IP access rule for UDP. The no command deletes this named extended IP access rule.
[no] {deny | permit} {eigrp | gre | igrp | ipinip | ip | ospf | <protocol-num>} {{<*sIpAddr*> <*sMask*>} | any-source | {host-source <*sIpAddr*>}} {{<*dIpAddr*> <*dMask*>} | any-destination | {host-destination <*dIpAddr*>}} [precedence <*prec*>] [tos <*tos*>][time-range<*time-range-name*>]	Creates a named extended IP access rule for other IP protocols. The no command deletes this named extended IP access rule.

c. Exit extended IP ACL configuration mode

Command	Explanation
Extended IP ACL Mode	
exit	Exits named extended IP ACL configuration mode.

(5) Configure a numbered standard MAC access list

Command	Explanation
Global Mode	
access-list<*num*>{deny|permit}{any-source-mac|{host-source-mac<*host_smac*>}|{<*smac*><*smac-mask*>}} **no access-list <*num*>**	Creates a numbered standard MAC access list. If the access list already exists, a rule is added to it. The **no access-list <*num*>** command deletes a numbered standard MAC access list.

(6) Configure a numbered extended MAC access list

Command	Explanation
Global Mode	
access-list<*num*> {deny|permit} {any-source-mac|{host-source-mac<*host_smac*>}|{<*smac*><*smac-mask*>}}{any-destination-mac|{host-destination-mac<*host_dmac*>}|{<*dmac*><*dmac-mask*>}} [{untagged-eth2 | tagged-eth2 | untagged-802-3 | tagged-802-3} [<*offset1*> <*length1*> <value1> [<*offset2*> <*length2*> <value2> [<offset3> <length3> <value3> [<*offset4*> <*length4*> <*value4*>]]]]] **no access-list <*num*>**	Creates a numbered extended MAC access list. If the access list already exists, a rule is added to it. The **no access-list <*num*>** command deletes a numbered extended MAC access list.

(7) Configure a named extended MAC access list

a. Create a named extended MAC access list

Command	Explanation
Global Mode	
mac-access-list extended <*name*> **no mac-access-list extended <*name*>**	Creates a named extended MAC access list. The no command deletes this named extended MAC access list.

b. Specify multiple "permit" or "deny" rules

Command	Explanation
Named Extended MAC ACL Mode	

[no]{deny|permit}{any-source-mac|{host-source-mac<*host_smac*>}|{<*smac*><*smac-mask*>}} {any-destination-mac|{host-destination-mac <*host_dmac*>} |{<*dmac*> <*dmac-mask*>}} [cos <*cos-val*> [<*cos-bitmask*>] [vlanId <*vid-value*> [<*vid-mask*>][ethertype<*protocol*>[<*protocol-mask*>]]]] **[no]{deny|permit} {any-source-mac |{host-source-mac<*host_smac*>}|{<*smac*><*smac-mask*>}} {any-destination-mac|{host-destination-mac<*host_dmac*>}|{<*dmac*><*dmac-mask*>}} [ethertype <*protocol*> [<*protocol-mask*>]]** **[no]{deny|permit} {any-source-mac|{host-source-mac<*host_smac*>}|{<*smac*><*smac-mask*>}} {any-destination-mac |{host-destination-mac<*host_dmac*>}|{<*dmac*><*dmac-mask*>}} [vlanid <*vid-value*> [<*vid-mask*>][ethertype <*protocol*> [<*protocol-mask*>]]]**	Creates a named extended MAC access rule that matches MAC frames. The no command deletes this named extended MAC access rule.
[no]{deny|permit}{any-source-mac|{host-source-mac<*host_smac*>}|{<*smac*><*smac-mask*>}}{any-destination-mac|{host-destination-mac<*host_dmac*>}|{<*dmac*><*dmac-mask*>}} [untagged-eth2 [ethertype <*protocol*> [protocol-mask]]]	Creates a named extended MAC access rule that matches untagged Ethernet II frames. The no command deletes this named extended MAC access rule.
[no]{deny|permit}{any-source-mac|{host-source-mac<*host_smac*>}|{<*smac*><*smac-mask*>}} {any-destination-mac|{host-destination-mac <*host_dmac*>}|{<*dmac*><*dmac-mask*>}} [untagged-802-3]	Creates a named extended MAC access rule that matches untagged 802.3 frames. The no command deletes this named extended MAC access rule.
[no]{deny|permit}{any-source-mac|{host-source-mac<*host_smac*>}|{<*smac*><*smac-mask*>}}{any-destination-mac|{host-destination-mac<*host_dmac*>}|{<*dmac*><*dmac-mask*>}}[tagged-eth2 [cos <*cos-val*> [<*cos-bitmask*>]] [vlanId <*vid-value*> [<*vid-mask*>]] [ethertype<*protocol*> [<*protocol-mask*>]]]	Creates a named extended MAC access rule that matches tagged Ethernet II frames. The no command deletes this named extended MAC access rule.
[no]{deny|permit}{any-source-mac|{host-source-mac <*host_smac*>}|{<*smac*><*smac-mask*>}} {any-destination-mac|{host-destination-mac<*host_dmac*>}|{<*dmac*><*dmac-mask*>}} [tagged-802-3 [cos <*cos-val*> [<*cos-bitmask*>]] [vlanId <*vid-value*> [<*vid-mask*>]]]	Creates a named extended MAC access rule that matches tagged 802.3 frames. The no command deletes this named extended MAC access rule.

c. Exit ACL Configuration Mode

Command	Explanation
Named Extended MAC ACL Mode	
exit	Exits named extended MAC ACL configuration mode.

(8) Configure a numbered extended MAC-IP access list

Command	Explanation
Global Mode	
access-list<*num*>{deny|permit} {any-source-mac| {host-source-mac <*host_smac*>} | {<*smac*> <*smac-mask*>}} {any-destination-mac | {host-destination-mac <*host_dmac*>} | {<*dmac*><*dmac-mask*>}} icmp	Creates a numbered extended MAC-IP access rule for ICMP. If the numbered extended access list does not

{{<*source*> <*source-wildcard*>} |any-source| {host-source <*source-host-ip*>}} {{<*destination*> <*destination-wildcard*>} | any-destination | {host-destination <*destination-host-ip*>}} [<*icmp-type*> [<*icmp-code*>]] [precedence <*precedence*>] [tos <*tos*>] [time-range <*time-range-name*>]	exist, an access list is created with this number.
access-list<*num*>{deny|permit}{any-source-mac| {host-source-mac<*host_smac*>}|{<*smac*><*smac-mask*>}} {any-destination-mac|{host-destination-mac <*host_dmac*>}|{<*dmac*><*dmac-mask*>}}igmp {{<*source*><*source-wildcard*>}|any-source| {host-source<*source-host-ip*>}} {{<*destination*><*destination-wildcard*>}|any-destination| {host-destination<*destination-host-ip*>}} [<*igmp-type*>] [precedence <*precedence*>] [tos <*tos*>][time-range<*time-range-name*>]	Creates a numbered extended MAC-IP access rule for IGMP. If the numbered extended access list does not exist, an access list is created with this number.
access-list<num>{deny|permit}{any-source-mac| {host-source-mac<host_smac>}|{<smac><smac-mask>}}{any-destination-mac|{host-destination-mac <host_dmac>}|{<dmac><dmac-mask>}}tcp {{<source><source-wildcard>}|any-source| {host-source<source-host-ip>}} [s-port {<*port1*> | range <*sPortMin*> <*sPortMax*>}] {{<destination><destination-wildcard>}|any-destination| {host-destination <destination-host-ip>}} [d-port {<*port3*> | range <*dPortMin*> <*dPortMax*>}] [ack+fin+psh+rst+urg+syn] [precedence <precedence>] [tos <tos>][time-range<*time-range-name*>]	Creates a numbered extended MAC-IP access rule for TCP. If the numbered extended access list does not exist, an access list is created with this number.
access-list<*num*>{deny|permit}{any-source-mac| {host-source-mac<*host_smac*>}|{<*smac*><*smac-mask*>}}{any-destination-mac|{host-destination-mac <*host_dmac*>}|{<*dmac*><*dmac-mask*>}}udp {{<*source*><*source-wildcard*>}|any-source| {host-source<*source-host-ip*>}} [s-port {<*port1*> | range <*sPortMin*> <*sPortMax*>}] {{<*destination*><*destination-wildcard*>}|any-destination| {host-destination<*destination-host-ip*>}} [d-port {<*port3*> | range <*dPortMin*> <*dPortMax*>}] [precedence <*precedence*>] [tos <*tos*>][time-range<*time-range-name*>]	Creates a numbered extended MAC-IP access rule for UDP. If the numbered extended access list does not exist, an access list is created with this number.
access-list<*num*>{deny|permit}{any-source-mac| {host-source-mac<*host_smac*>}|{<*smac*><*smac-mask*>}} {any-destination-mac|{host-destination-mac <*host_dmac*>}|{<*dmac*><*dmac-mask*>}} {eigrp| gre|igrp|ip|ipinip|ospf|{<*protocol-num*>}} {{<*source*><*source-wildcard*>}|any-source| {host-source<*source-host-ip*>}} {{<*destination*><*destination-wildcard*>}|any-destination| {host-destination<*destination-host-ip*>}} [precedence <*precedence*>] [tos <*tos*>][time-range<*time-range-name*>]	Creates a numbered extended MAC-IP access rule for another specific protocol or for all protocols. If the numbered extended access list does not exist, an access list is created with this number.
no access-list <*num*>	Deletes this numbered extended MAC-IP access list.

(9) Configure a named extended MAC-IP access list

a. Create a named extended MAC-IP access list

Command	Explanation
Global Mode	

mac-ip-access-list extended <*name*> **no mac-ip-access-list extended <*name*>**	Creates a named extended MAC-IP access list. The no command deletes this named extended MAC-IP access list.

b. Specify multiple "permit" or "deny" rules

Command	Explanation
Named Extended MAC-IP ACL Mode	
[no]{deny|permit} {any-source-mac|{host-source-mac <*host_smac*>}|{<*smac*><*smac-mask*>}} {any-destination-mac|{host-destination-mac <*host_dmac*>}|{<*dmac*><*dmac-mask*>}}icmp {{<*source*><*source-wildcard*>}|any-source| {host-source<*source-host-ip*>}} {{<*destination*><*destination-wildcard*>}|any-destination| {host-destination <*destination-host-ip*>}} [<*icmp-type*> [<*icmp-code*>]] [precedence <*precedence*>][tos<*tos*>][time-range<*time-range-name*>]	Creates a named extended MAC-ICMP access rule. The no command deletes this named extended MAC-ICMP access rule.
[no]{deny|permit}{any-source-mac|{host-source-mac <*host_smac*>}|{<*smac*><*smac-mask*>}} {any-destination-mac|{host-destination-mac <*host_dmac*>}|{<*dmac*><*dmac-mask*>}}igmp {{<*source*><*source-wildcard*>}|any-source| {host-source<*source-host-ip*>}} {{<*destination*><*destination-wildcard*>}|any-destination| {host-destination <*destination-host-ip*>}} [<*igmp-type*>] [precedence <*precedence*>] [tos <*tos*>][time-range<*time-range-name*>]	Creates a named extended MAC-IGMP access rule. The no command deletes this named extended MAC-IGMP access rule.
[no]{deny|permit}{any-source-mac|{host-source-mac<host_smac>}|{<smac><smac-mask>}} {any-destination-mac|{host-destination-mac <host_dmac>}|{<dmac><dmac-mask>}}tcp {{<source><source-wildcard>}|any-source| {host-source<source-host-ip>}} [s-port {<*port1*> | range <*sPortMin*> <*sPortMax*>}] {{<destination><destination-wildcard>}|any-destination| {host-destination <destination-host-ip>}} [d-port {<*port3*> | range <*dPortMin*> <*dPortMax*>}] [ack+fin+psh+rst+urg+syn] [precedence<precedence>][tos<tos>][time-range<*time-range-name*>]	Creates a named extended MAC-TCP access rule. The no command deletes this named extended MAC-TCP access rule.
[no]{deny|permit}{any-source-mac|{host-source-mac<*host_smac*>}|{<*smac*><*smac-mask*>}} {any-destination-mac|{host-destination-mac <*host_dmac*>}|{<*dmac*><*dmac-mask*>}}udp {{<*source*><*source-wildcard*>}|any-source| {host-source<*source-host-ip*>}} [s-port {<*port1*> | range <*sPortMin*> <*sPortMax*>}] {{<*destination*><*destination-wildcard*>}|any-destination| {host-destination <*destination-host-ip*>}} [d-port {<*port3*> | range <*dPortMin*> <*dPortMax*>}] [precedence <*precedence*>] [tos <*tos*>][time-range<*time-range-name*>]	Creates a named extended MAC-UDP access rule. The no command deletes this named extended MAC-UDP access rule.
[no]{deny|permit}{any-source-mac|{host-source-mac<*host_smac*>}|{<*smac*><*smac-mask*>}} {any-destination-mac|{host-destination-mac <*host_dmac*>}|{<*dmac*><*dmac-mask*>}} {eigrp|gre| igrp|ip|ipinip|ospf|{<*protocol-num*>}} {{<*source*><*source-wildcard*>}|any-source| {host-	Creates a named extended access rule for other IP protocols. The no command deletes this named extended access rule.

source<*source-host-ip*>}} {{<*destination*><*destination-wildcard*>}|any-destination| {host-destination<*destination-host-ip*>}} [precedence<*precedence*>][tos<*tos*>][time-range<*time-range-name*>]	

c. Exit MAC-IP Configuration Mode

Command	Explanation
Named Extended MAC-IP ACL Mode	
exit	Exits named extended MAC-IP ACL configuration mode.

(10) Configure a numbered standard IPv6 access list

Command	Explanation
Global Mode	
ipv6 access-list <*num*> {deny | permit} {{<*sIPv6Addr*> <*sPrefixlen*>} | any-source | {host-source <*sIpv6Addr*>}} no ipv6 access-list <*num*>	Creates a numbered standard IPv6 access list. If the access list already exists, a rule is added to it. The **no access-list <*num*>** command deletes a numbered standard IPv6 access list.

(11) Configure a numbered extended IPv6 access list

Command	Explanation
Global Mode	
ipv6 access-list <num-ext> {deny | permit} icmp {{<sIPv6Prefix/sPrefixlen>} | any-source | {host-source <sIPv6Addr>}} {<dIPv6Prefix/dPrefixlen> | any-destination | {host-destination <dIPv6Addr>}} [<icmp-type> [<icmp-code>]] [dscp <dscp>] [flow-label <fl>][time-range<time-range-name>] **ipv6 access-list <num-ext> {deny | permit} tcp {{<sIPv6Prefix/<sPrefixlen>} | any-source | {host-source <sIPv6Addr>}} [s-port {<sPort> | range <sPortMin> <sPortMax>}] {{< dIPv6Prefix/<dPrefixlen>} | any-destination | {host-destination <dIPv6Addr>}} [dPort {<dPort> | range <dPortMin> <dPortMax>}] [syn | ack | urg | rst | fin | psh] [dscp <dscp>] [flow-label <flowlabel>][time-range<time-range-name>]** **ipv6 access-list <num-ext> {deny | permit} udp {{<sIPv6Prefix/<sPrefixlen>} | any-source | {host-source <sIPv6Addr>}} [s-port {<sPort> | range <sPortMin> <sPortMax>}] {{<dIPv6Prefix/<dPrefixlen>} | any-destination | {host-destination <dIPv6Addr>}} [dPort {<dPort> | range <dPortMin> <dPortMax>}] [dscp <dscp>] [flow-label <flowlabel>][time-range<time-range-name>]** **ipv6 access-list <num-ext> {deny | permit} <next-header> {<sIPv6Prefix/sPrefixlen> | any-source | {host-source <sIPv6Addr>}} {<dIPv6Prefix/dPrefixlen> | any-destination | {host-destination <dIPv6Addr>}} [dscp <dscp>] [flow-label <fl>][time-range<time-range-name>]** **no ipv6 access-list <num>**	Creates a numbered extended IPv6 access list. If the access list already exists, a rule is added to it. The no command deletes a numbered extended IPv6 access list.

(12) Configure a named standard IPv6 access list

a. Create a named standard IPv6 access list

Command	Explanation
Global Mode	
ipv6 access-list standard <name> **no ipv6 access-list standard <name>**	Creates a named standard IPv6 access list. The no command deletes the named standard IPv6 access list.

b. Specify multiple permit or deny rules

Command	Explanation
Standard IPv6 ACL Mode	
[no] {deny | permit} {{<sIPv6Prefix/sPrefixlen>} | any-source | {host-source <sIPv6Addr> }}	Creates a named standard IPv6 access rule. The no command deletes the named standard IPv6 access rule.

c. Exit named standard IPv6 ACL configuration mode

Command	Explanation
Standard IPv6 ACL Mode	
exit	Exits name-based standard IPv6 ACL configuration mode.

(13) Configure a named extended IPv6 access list

a. Create a named extended IPv6 access list

Command	Explanation
Global Mode	
ipv6 access-list extended <name> **no ipv6 access-list extended <name>**	Creates a named extended IPv6 access list. The no command deletes the named extended IPv6 access list.

b. Specify multiple permit or deny rules

Command	Explanation
Extended IPv6 ACL Mode	
[no] {deny | permit} icmp {{<sIPv6Prefix/sPrefixlen>} | any-source | {host-source <sIPv6Addr>}} {<dIPv6Prefix/dPrefixlen> | any-destination | {host-destination <dIPv6Addr>}} [<icmp-type> [<icmp-code>]] [dscp <dscp>] [flow-label <flowlabel>] [time-range <time-range-name>]	Creates a named extended IPv6 access rule for ICMP. The no command deletes this named extended IPv6 access rule.
[no] {deny | permit} tcp {<sIPv6Prefix/sPrefixlen> | any-source | {host-source <sIPv6Addr>}} [s-port {<sPort> | range <sPortMin> <sPortMax>}] {<dIPv6Prefix/dPrefixlen> | any-destination | {host-destination <dIPv6Addr>}} [d-port {<dPort> | range <dPortMin> <dPortMax>}] [syn | ack | urg | rst | fin | psh] [dscp <dscp>] [flow-label <fl>] [time-range<time-range-name>]	Creates a named extended IPv6 access rule for TCP. The no command deletes this named extended IPv6 access rule.
[no] {deny | permit} udp {<sIPv6Prefix/sPrefixlen> | any-	Creates a named extended IPv6 access rule for UDP. The no command deletes this named

source | {host-source <sIPv6Addr>}} [s-port {<sPort> | range <sPortMin> <sPortMax>}] {<dIPv6Prefix/dPrefixlen> | any-destination | {host-destination <dIPv6Addr>}} [d-port {<dPort> | range <dPortMin> <dPortMax>}] [dscp <dscp>] [flow-label <fl>] [time-range<time-range-name>]	extended IPv6 access rule.
[no] {deny | permit} <proto> {<sIPv6Prefix/sPrefixlen> | any-source | {host-source <sIPv6Addr>}} {<dIPv6Prefix/dPrefixlen> | any-destination | {host-destination <dIPv6Addr>}} [dscp <dscp>] [flow-label <flowlabel>] [time-range <time-range-name>]	Creates a named extended IPv6 access rule for other IPv6 protocols. The no command deletes this named extended IPv6 access rule.
[no] {deny | permit} {<sIPv6Prefix/sPrefixlen> | any-source | {host-source <sIPv6Addr>}} {<dIPv6Prefix/dPrefixlen> | any-destination | {host-destination <dIPv6Addr>}} [dscp <dscp>] [flow-label <flowlabel>] [time-range <time-range-name>]	Creates a named extended IPv6 access rule. The no command deletes this named extended IPv6 access rule.

c. Exit extended IPv6 ACL configuration mode

Command	Explanation
Extended IPv6 ACL Mode	
exit	Exits named extended IPv6 ACL configuration mode.

2. Configure the packet filtering function

(1) Enable the global packet filtering function

Command	Explanation
Global Mode	
firewall enable	Enables the global packet filtering function.
firewall disable	Disables the global packet filtering function.

3. Configure the time range function

(1) Create a time range

Command	Explanation
Global Mode	
time-range <*time_range_name*>	Creates a time range named time_range_name.
no time-range <*time_range_name*>	Stops the time range named time_range_name.

(2) Configure a periodic time range

Command	Explanation
Time Range Mode	

absolute-periodic {Monday | Tuesday | Wednesday | Thursday | Friday | Saturday | Sunday} <*start_time*> to {Monday | Tuesday | Wednesday | Thursday | Friday | Saturday | Sunday} <*end_time*>	Configures a time range for the specified days of the week; the time range repeats every week.
periodic {{Monday+Tuesday+Wednesday+Thursday+ Friday+Saturday+Sunday} | daily | weekdays | weekend} <*start_time*> to <*end_time*>	
[no] absolute-periodic {Monday | Tuesday | Wednesday | Thursday | Friday | Saturday | Sunday} <*start_time*> to {Monday | Tuesday | Wednesday | Thursday | Friday | Saturday | Sunday} <*end_time*>	Stops the specified weekly time range.
[no] periodic {{Monday+Tuesday+Wednesday+Thursday+ Friday+Saturday+Sunday} | daily | weekdays | weekend} <*start_time*> to <*end_time*>	

(3) Configure an absolute time range

Command	Explanation
Global Mode	
absolute start <*start_time*> <*start_data*> [end <*end_time*> <*end_data*>]	Configures an absolute time range.
[no] absolute start <*start_time*> <*start_data*> [end <*end_time*> <*end_data*>]	Stops the absolute time range.

4. Bind an access list to a specific direction of the specified port

Command	Explanation
Port Mode/VLAN Interface Mode	
{ip|ipv6|mac|mac-ip} access-group <*acl-name*> {in|out} [traffic-statistic] **no {ip|ipv6|mac|mac-ip} access-group <*acl-name*> {in|out}**	Applies an access list to the ingress or egress direction of the port. The no command removes the access list bound to the port.

5. Clear the filtering information of the specified port

Command	Explanation
Admin Mode	
clear access-group (in | out) statistic interface { <*interface-name*> | ethernet <*interface-name*> }	Clears the ingress or egress filtering information of the specified port.

7.1.3 ACL Example

Scenario 1:

Port 10 of the switch connects to the 10.0.0.0/24 network segment, and FTP access must be blocked for the users on this segment.

Configuration description:

1．Create an appropriate ACL.

2．Configure the packet filtering function.

3．Bind the ACL to the port.

The configuration steps are listed below:

```
Switch(config)#access-list 110 deny tcp 10.0.0.0 0.0.0.255 any-destination d-
port 21
Switch(config)#firewall enable
Switch(config)#interface ethernet 1/0/10
Switch(Config-If-Ethernet1/0/10)#ip access-group 110 in
Switch(Config-If-Ethernet1/0/10)#exit
Switch(config)#exit
```

Configuration result:

```
Switch#show firewall
Firewall status: enable.
Switch#show access-lists
access-list 110(used 1 time(s)) 1 rule(s)
access-list 110 deny tcp 10.0.0.0 0.0.0.255 any-destination d-port 21
Switch#show access-group interface ethernet 1/0/10
interface name:Ethernet1/0/10
the ingress acl use in firewall is 110, traffic-statistics Disable.
```

Scenario 2:

Requirement: the switch must drop all 802.3 frames with a source MAC address of 00-12-11-23-xx-xx received on port 10.

Configuration description:

1．Create the corresponding MAC ACL.

2．Configure packet filtering.

3．Bind the ACL to the related interface.

The configuration steps are listed below.

```
Switch(config)#access-list 1100 deny 00-12-11-23-00-00 00-00-00-00-ff-ff any-
destination-mac untagged-802-3
Switch(config)#access-list 1100 deny 00-12-11-23-00-00 00-00-00-00-ff-ff any
tagged-802
Switch(config)#firewall enable
Switch(config)#interface ethernet1/0/10
Switch(Config-If-Ethernet1/0/10)#mac access-group 1100 in
Switch(Config-If-Ethernet1/0/10)#exit
Switch(config)#exit
```

Configuration result:

```
Switch#show firewall
Firewall Status: Enable.
Switch #show access-lists
access-list 1100(used 1 time(s))
access-list 1100 deny 00-12-11-23-00-00 00-00-00-00-ff-ff
any-destination-mac
untagged-802-3
access-list 1100 deny 00-12-11-23-00-00 00-00-00-00-ff-ff
any-destination-mac
Switch #show access-group interface ethernet 1/0/10
interface name:Ethernet1/0/10
MAC Ingress access-list used is 1100,traffic-statistics Disable.
```

Scenario 3:

Requirement: the MAC address range of the network connected to port 10 of the switch is 00-12-11-23-xx-xx, and the IP network is 10.0.0.0/24. FTP must be blocked, and ping requests from the outside network must be blocked.

Configuration description:

1．Create the corresponding access list.

2．Configure packet filtering.

3．Bind the ACL to the related interface.

The configuration steps are listed below.

```
Switch(config)#access-list 3110 deny 00-12-11-23-00-00 00-00-00-00-ff-ff any-
destination-mac tcp 10.0.0.0 0.0.0.255 any-destination d-port 21
Switch(config)#access-list 3110 deny any-source-mac 00-12-11-23-00-00 00-00-
00-00-ff-ff icmp any-source 10.0.0.0 0.0.0.255
Switch(config)#firewall enable
Switch(config)#interface ethernet 1/0/10
Switch(Config-If-Ethernet1/0/10)#mac-ip access-group 3110 in
Switch(Config-Ethernet1/0/10)#exit
Switch(config)#exit
```

Configuration result:

```
Switch#show firewall
Firewall Status: Enable.
Switch#show access-lists
access-list 3110(used 1 time(s))
access-list 3110 deny 00-12-11-23-00-00 00-00-00-00-ff-ff
any-destination-mac
tcp 10.0.0.0 0.0.0.255 any-destination d-port 21
access-list 3110 deny any-source-mac 00-12-11-23-00-00 00-00-00-00-ff-ff icmp
any-source 10.0.0.0 0.0.0.255
Switch #show access-group interface ethernet 1/0/10
interface name:Ethernet1/0/10
MAC-IP Ingress access-list used is 3110, traffic-statistics Disable.
```

Scenario 4:

Requirement: IPv6 runs on port 10 of the switch, and the IPv6 network address is 2003:1:1:1::0/64. Users in the 2003:1:1:1:66::0/80 subnet must be blocked from accessing the outside network.

Configuration description:

1．Create the corresponding access list.

2．Configure packet filtering.

3．Bind the ACL to the related interface.

The configuration steps are listed below.

```
Switch(config)#ipv6 access-list 600 permit 2003:1:1:1:66::0/80 any-
destination
Switch(config)#ipv6 access-list 600 deny 2003:1:1:1::0/64 any-destination
Switch(config)#firewall enable
Switch(config)#interface ethernet 1/0/10
Switch(Config-If-Ethernet1/0/10)#ipv6 access-group 600 in
Switch(Config-If-Ethernet1/0/10)#exit
Switch(config)#exit
```

Configuration result:

```
Switch#show firewall
Firewall Status: Enable.
Switch#show ipv6 access-lists
Ipv6 access-list 600(used 1 time(s))
ipv6 access-list 600 deny 2003:1:1:1::0/64 any-source
ipv6 access-list 600 permit 2003:1:1:1:66::0/80 any-source
Switch #show access-group interface ethernet 1/0/10
interface name:Ethernet1/0/10
IPv6 Ingress access-list used is 600, traffic-statistics Disable.
```

Scenario 5:

Requirement: ports 1, 2, 5, and 7 belong to VLAN 100, and the host with IP address 192.168.0.1 must be blocked from accessing these ports.

Configuration description:

1．Create the corresponding access list.

2．Configure packet filtering.

3．Bind the ACL to the related interface.

The configuration steps are listed below.

```
Switch (config)#firewall enable
Switch (config)#vlan 100
Switch (Config-Vlan100)#switchport interface ethernet 1/0/1;2;5;7
Switch (Config-Vlan100)#exit
Switch (config)#access-list 1 deny host-source 192.168.0.1
Switch (config)#interface ethernet1/0/1;2;5;7
Switch (config-if-port-range)#ip access-group 1 in
Switch (Config-if-Vlan100)#exit
```

Configuration result:

```
Switch (config)#show access-group interface vlan 100
Interface VLAN 100:
Ethernet1/0/1:   IP Ingress access-list used is 1, traffic-statistics
Disable.
Ethernet1/0/2:   IP Ingress access-list used is 1, traffic-statistics
Disable.
Ethernet1/0/5:   IP Ingress access-list used is 1, traffic-statistics
Disable.
Ethernet1/0/7:   IP Ingress access-list used is 1, traffic-statistics
Disable.
```

7.1.4 ACL Troubleshooting

- Entries in an ACL are checked from top to bottom, and checking stops at the first matching entry.
- Each ingress port can be bound to one MAC-IP ACL, one IP ACL, one MAC ACL, and one IPv6 ACL (in Port Mode or VLAN Interface Mode).
- When four ACLs are bound and a packet matches several of them at the same time, the priorities from highest to lowest are as follows. If the priorities are equal, the ACL configured first has the higher priority.
    - Ingress IPv6 ACL
    - Ingress MAC-IP ACL
    - Ingress IP ACL
    - Ingress MAC ACL
- The number of ACLs that can be bound depends on the content of the ACLs and on hardware resource limits. A message is displayed if an ACL cannot be bound because of hardware resource limits.
- If an access list contains rules with the same filtering information but conflicting actions, binding it to a port fails with an error message. For example, "permit tcp any any-destination" and "deny tcp any any-destination" cannot be configured at the same time.
- Viruses such as "worm.blaster" can be blocked by configuring ACLs that block specific ICMP packets or packets with specific TCP or UDP ports.

7.2 Self-Defined ACL

7.2.1 Introduction to Self-Defined ACL

7.2.1.1 Named Self-Defined ACL

A named self-defined ACL can contain multiple ACL lists, and each list can contain multiple rules. The number of lists that can be issued depends on the card type. A rule can specify a value and a mask for up to 16 windows, and each window is 2 bytes long. The name string is 1 to 64 characters long and cannot consist of digits only.

7.2.2 Self-Defined ACL Configuration

Self-defined ACL configuration task list:

1. Configure the offset template
2. Configure a numbered standard user-defined ACL
3. Configure a named standard user-defined ACL
4. Bind the user-defined ACL to a port

1. Configure the offset template

Command	Explanation
Global Mode	
userdefined-access-list standard offset [window1 <*offset*>] [window2 <*offset*>] [window3 <*offset*>] [window4 <*offset*>] [window5 <*offset*>] [window6 <*offset*>] [window7 <*offset*>] [window8 <*offset*>] [window9 <*offset*>] [window10 <*offset*>] [window11 <*offset*>] [window12 <*offset*>] [window13 <*offset*>] [window14 <*offset*>] [window15 <*offset*>] [window16 <*offset*>] **no userdefined-access-list standard offset [window1] [window2] [window3] [window4] [window5] [window6] [window7] [window8] [window9] [window10] [window11] [window12] [window13] [window14] [window15] [window16]**	Creates a standard self-defined ACL template. If the template exists, the corresponding windows of the template can be modified. The no command deletes the specified windows of the template; if no window is specified, the whole template is deleted.

2. Configure the digital standard userdefined acl

Command	Explanation
Global Mode	
userdefined-access-list standard <*num*> {deny | permit} [packet-type ipv4 | ipv6 | l2-eth2 | l2-llc | l2-snap | mpls] [window1 <*value*> <*mask*>] [window2 <*value*> <*mask*>] [window3 <*value*> <*mask*>] [window4 <*value*> <*mask*>] [window5 <*value*> <*mask*>] [window6 <*value*> <*mask*>] [window7 <*value*> <*mask*>] [window8 <*value*> <*mask*>] [window9 <*value*> <*mask*>] [window10 <*value*> <*mask*>] [window11 <*value*> <*mask*>] [window12 <*value*> <*mask*>] [window13 <*value*> <*mask*>] [window14 <*value*> <*mask*>] [window15 <*value*> <*mask*>]	Creates a numbered standard self-defined ACL. If the ACL already exists, a rule is added to it. The no command deletes the numbered standard self-defined ACL.

[window16 <*value*> <*mask*>] **no userdefined-access-list <*num*>**	

3. Configure the named standard userdefined acl

Command	Explanation
Global Mode	
[no] udf-access-list standard <*name*>	Creates a named user-defined access list. The no command deletes it.
Named User-Defined ACL Mode	
[no] {deny | permit} [packet-type ipv4 | ipv6 | l2-eth2 | l2-llc | l2-snap | mpls] [window1 <*value*> <*mask*>] [window2 <*value*> <*mask*>] [window3 <*value*> <*mask*>] [window4 <*value*> <*mask*>] [window5 <*value*> <*mask*>] [window6 <*value*> <*mask*>] [window7 <*value*> <*mask*>] [window8 <*value*> <*mask*>] [window9 <*value*> <*mask*>] [window10 <*value*> <*mask*>] [window11 <*value*> <*mask*>] [window12 <*value*> <*mask*>] [window13 <*value*> <*mask*>] [window14 <*value*> <*mask*>] [window15 <*value*> <*mask*>] [window16 <*value*> <*mask*>]	Adds a rule to the named user-defined access list. The no command deletes the rule.

4. Bind the userdefined acl rule to the port

Command	Explanation
Port Mode	
[no] userdefined access-group {<*name*>| <*num*>*}* {in} [traffic-statistic]	Applies a user-defined access list to the specified direction of the port; the traffic-statistic option adds a statistics counter to the ACL. The no command removes the binding from the port.

7.2.3 Self-Defined ACL Example

Scenario 1:

Requirement: packets whose first and second bytes are 0x0003 must not be forwarded on port 1 of the switch.

Configuration description:

1. Create a self-defined ACL template based on the condition.
2. Create the corresponding self-defined ACL.
3. Bind the self-defined ACL to the port.

The configuration steps are listed below:

```
Switch(config)#userdefined-access-list standard offset window1 0
Switch(config)#userdefined-access-list standard 1200 deny window1 0003 FFFF
Switch(config)#
Switch(config)#firewall enable
Switch(config)#interface ethernet 1/0/1
Switch(config-if-ethernet1/0/1)#userdefined access-group 1200 in
Switch(config)#exit
```

Configuration result:

```
Switch #show access-lists
userdefined-access-list standard 1200(used 1 time(s)) 1 rule(s)
   rule ID 1: deny window1 0003 ffff
Switch#show access-group interface ethernet 1/0/1
```

```
interface name:Ethernet1/0/1
Userdefined Ingress access-list used is 1200, traffic-statistics Disable.
```

7.2.4 Self-Defined ACL Troubleshooting

Troubleshoot self-defined ACLs in the same way as ordinary ACLs. If an operation is incorrect, the system displays detailed error information.

7.3 802.1x

7.3.1 Introduction to 802.1x

The 802.1x protocol originated from 802.11, the IEEE wireless LAN protocol, and was designed to provide authentication when users access a wireless LAN. LANs defined by the IEEE 802 LAN protocols do not provide access authentication: any user who can connect to a LAN access device (such as a LAN switch) can reach all devices and resources in the LAN. This posed no real risk in early enterprise networks.

However, with the growth of applications such as mobile office and carrier service networks, service providers need to control and configure user access. In particular, the wide use of WLAN and LAN access in telecommunication networks makes port control necessary for user-level access control. As a result, the IEEE LAN/WAN committee defined the 802.1x standard for Port-Based Network Access Control. This standard is widely used in wireless LANs and Ethernet.

"Port-Based Network Access Control" means authenticating and controlling user devices at the port level of LAN access devices. User devices connected to a port can access LAN resources only after they pass authentication; otherwise, LAN resources are unavailable to them.

7.3.1.1 The Authentication Structure of 802.1x

A system using 802.1x has a typical client/server structure with three entities, as shown in the figure below: the supplicant system, the authenticator system, and the authentication server system.

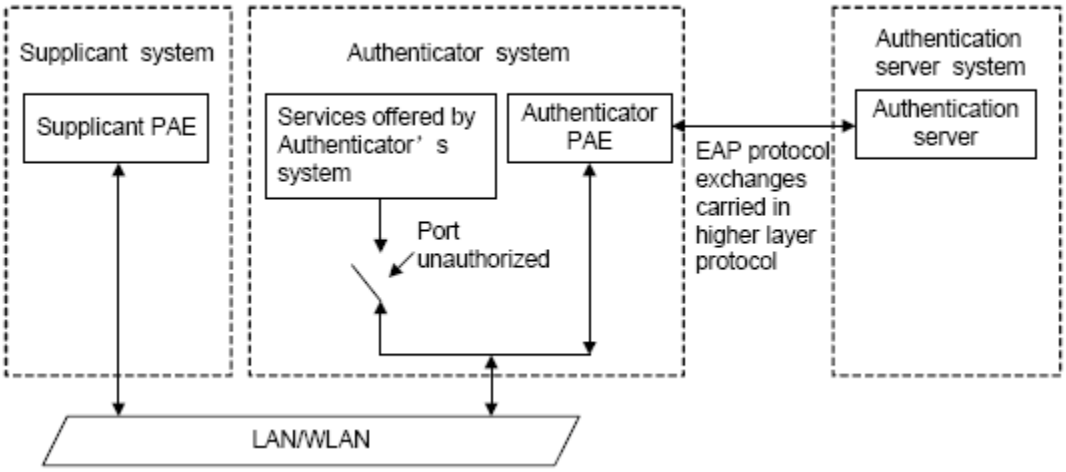


Figure 7-14 The Authentication Structure of 802.1x

- The supplicant system is an entity at one end of a LAN segment that is authenticated by the access control device at the other end of the link. A supplicant system is usually a user terminal. Users start 802.1x authentication by running the supplicant software. A supplicant system must support EAPOL (Extensible Authentication Protocol over LAN).
- The authenticator system is the entity at the other end of the LAN segment that authenticates the connected supplicant systems. An authenticator system is usually a network device that supports 802.1x and provides ports through which supplicant systems access the LAN. These ports can be physical or logical.

- The authentication server system provides authentication services for authenticator systems. It authenticates and authorizes users and performs accounting. It is usually a RADIUS (Remote Authentication Dial-In User Service) server, which stores user information such as the username, password, and other parameters, including the VLAN and ports to which the user belongs.

These three entities involve the following basic concepts: the PAE of a port, controlled ports, and the controlled direction.

1. PAE

The PAE (Port Access Entity) is the entity that runs the algorithms and protocol operations.

☞ The PAE of the supplicant system responds to authentication requests from the authenticator system and submits the user's authentication information to it. It can also send authentication requests and logoff requests to the authenticator.

☞ The PAE of the authenticator system authenticates the supplicant systems that need to access the LAN through the authentication server system, and sets the controlled port to the authenticated or unauthenticated state based on the authentication result. In the authenticated state, the user can access network resources; in the unauthenticated state, only EAPOL messages can be received and sent, and the user cannot access network resources.

2. Controlled/Uncontrolled Ports

The authenticator system provides ports through which supplicant systems access the LAN. Each of these ports is divided into two logical ports: a controlled port and an uncontrolled port.

☞ The uncontrolled port is always connected in both directions and is mainly used to transmit EAPOL protocol frames, ensuring that supplicant systems can always send and receive authentication messages.

☞ The controlled port is connected and transmits service messages only in the authenticated state. In the unauthenticated state, it does not receive any messages from supplicant systems.

☞ The controlled and uncontrolled ports are two parts of the same port: every frame that arrives at the port is visible on both of them.

3. Controlled Direction

In the unauthenticated state, a controlled port can be set to unidirectional or bidirectional control.

☞ With bidirectional control, sending and receiving of all frames are prohibited.

☞ With unidirectional control, frames from supplicant systems cannot be received, but frames can be sent to supplicant systems.

Note: Currently, the switch supports only unidirectional control.

7.3.1.2 802.1x Working Mechanism

IEEE 802.1x authentication uses EAP (Extensible Authentication Protocol) to exchange authentication information among the supplicant system, the authenticator system, and the authentication server system.

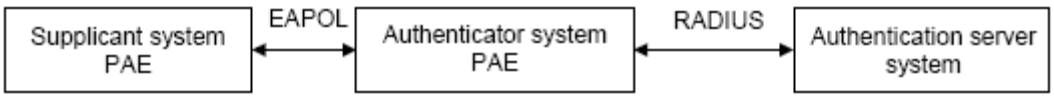


Figure 7-2 802.1x Working Mechanism

☞ In a LAN environment, EAP messages between the PAE of the supplicant system and the PAE of the authenticator system are encapsulated in EAPOL format.

☞ Between the PAE of the authenticator system and the RADIUS server, information can be exchanged in two ways: either EAP messages are encapsulated in the RADIUS protocol in EAPoR (EAP over RADIUS) format, or EAP messages are terminated at the PAE of the

authenticator system, which then uses messages containing PAP (Password Authentication Protocol) or CHAP (Challenge Handshake Authentication Protocol) attributes to authenticate with the RADIUS server.

☞ After the user passes authentication, the authentication server system sends the user's information to the authenticator system. The PAE of the authenticator system then sets the controlled port to the authenticated or unauthenticated state based on the authentication result from the RADIUS server.

7.3.1.3 The Encapsulation of EAPOL Messages

1. EAPOL Packet Format

EAPOL is a message encapsulation format defined in the 802.1x protocol. It is mainly used to transmit EAP messages between the supplicant system and the authenticator system across the LAN. In an IEEE 802/Ethernet LAN, the EAPOL packet format is shown in the figure below. The EAPOL packet starts at the Type/Length field of the MAC frame.

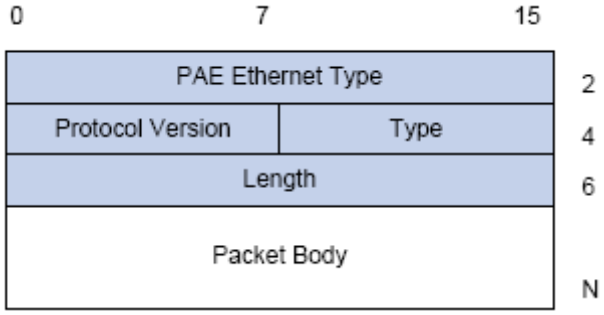


Figure 7-3 EAPOL Packet Format

PAE Ethernet Type: the protocol type; the value is 0x888E.

Protocol Version: the protocol version supported by the sender of the EAPOL packet.

Type: the type of the EAPOL packet, which can be one of the following:

☞ EAP-Packet (0x00): an authentication information frame that carries EAP messages. This frame can pass through the authenticator system to carry EAP messages between the supplicant system and the authentication server system.

☞ EAPOL-Start (0x01): a frame that starts authentication.

☞ EAPOL-Logoff (0x02): a frame that requests logoff.

☞ EAPOL-Key (0x03): a key information frame.

☞ EAPOL-Encapsulated-ASF-Alert (0x04): supports ASF (Alert Standard Forum) alerting messages. This frame encapsulates network management information, such as various alerts, and is terminated by terminal devices.

Length: the length of the data, that is, the length of the Packet Body, in bytes. If the value is 0, no data field follows.

Packet Body: the data content, whose format depends on the Type.

2. EAP Packet Format

When the Type field of an EAPOL packet is EAP-Packet, the Packet Body is in EAP format, as shown in the figure below.

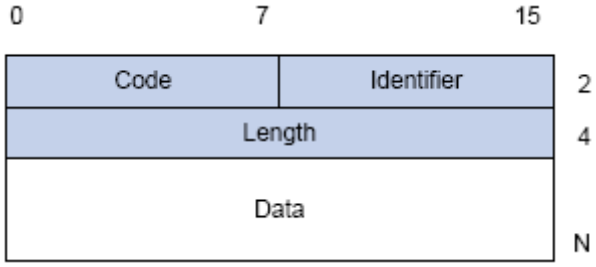


Figure 7-4 EAP Packet Format

Code: the type of the EAP packet. There are four types: Request (1), Response (2), Success (3), and Failure (4).

- ☞ Success and Failure packets have no Data field, and the value of their Length field is 4.
- ☞ The format of the Data field in Request and Response packets is shown in the figure below. Type is the EAP authentication type, and the content of Type Data depends on the Type. For example, Type 1 (Identity) is used to query the identity of the peer, and Type 4 (MD5-Challenge) contains challenge information, similar to PPP CHAP.

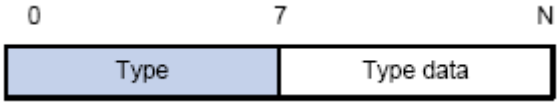


Figure 7-5 Format of the Data Field in Request and Response Packets

Identifier: helps match Request and Response messages.

Length: the length of the EAP packet in bytes, including the Code, Identifier, Length, and Data fields.

Data: the content of the EAP packet, which depends on the Code type.

7.3.1.4 The Encapsulation of EAP Attributes

RADIUS adds two attributes to support EAP authentication: EAP-Message and Message-Authenticator. For the RADIUS message format, see the RADIUS introduction later in this chapter.

1. EAP-Message

As shown in the figure below, this attribute encapsulates EAP packets. Its type code is 79, and the String field can be no longer than 253 bytes. If the data in an EAP packet is longer than 253 bytes, the packet can be fragmented and encapsulated in several EAP-Message attributes in their original order.

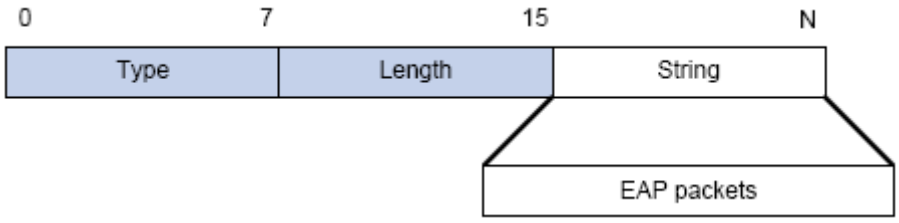


Figure 7-6 EAP-Message Attribute Encapsulation

2. Message-Authenticator

As shown in the figure below, this attribute is used with authentication methods such as EAP and CHAP to prevent access request packets from being eavesdropped. Packets that contain the EAP-Message attribute must also contain the Message-Authenticator attribute; otherwise, they are dropped as invalid.

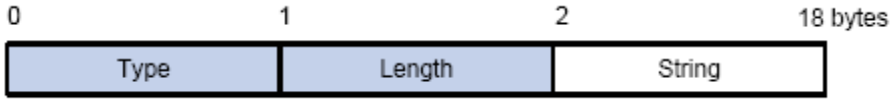


Figure 7-15 Message-Authenticator Attribute

7.3.1.5 The Authentication Methods of 802.1x

Authentication can be initiated either by the supplicant system or by the device. When the device detects an unauthenticated user accessing the network, it sends an EAP-Request/Identity message to the supplicant system to start authentication. Alternatively, the supplicant system can send an EAPOL-Start message to the device through the supplicant software.

802.1x systems support the EAP relay mode and the EAP termination mode for authentication with a remote RADIUS server. The following describes both authentication processes, each initiated by the supplicant system.

7.3.1.5.1 EAP Relay Mode

EAP relay is specified in the IEEE 802.1x standard to carry EAP in other higher-layer protocols, such as EAP over RADIUS, so that EAP messages can reach the authentication server through complex networks. In general, EAP relay requires the RADIUS server to support the EAP attributes EAP-Message and Message-Authenticator.

EAP is a widely used authentication framework that carries the actual authentication protocol, rather than a specific authentication mechanism. EAP provides common functions and allows the authentication mechanism, called the EAP method, to be negotiated. The advantage of EAP is that the underlying EAP mechanism needs no change when a new authentication protocol appears. The following figure shows the protocol stack of the EAP authentication methods.

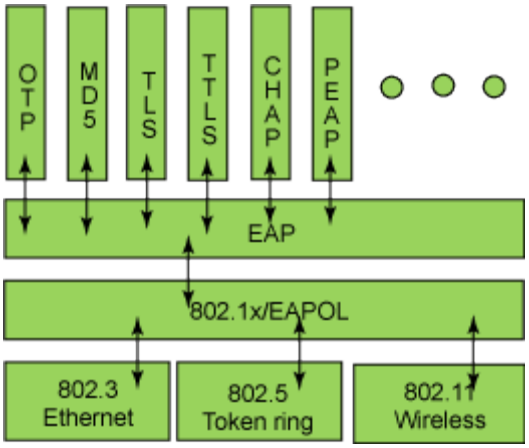


Figure 7-8 Protocol Stack of EAP Authentication Methods

More than 50 EAP authentication methods have been developed so far; they differ in authentication mechanism and key management. The four most common EAP authentication methods are:

- **EAP-MD5**
- **EAP-TLS** (Transport Layer Security)
- **EAP-TTLS** (Tunneled Transport Layer Security)
- **PEAP** (Protected Extensible Authentication Protocol)

They are described in detail below.

Note:

- As a pass-through access control device, the switch does not check the content of specific EAP methods. It therefore supports all the EAP methods above and any EAP methods that may be added in the future.

☞ In EAP relay mode, when EAP-MD5, EAP-TLS, EAP-TTLS, or PEAP is used, the supplicant system and the RADIUS server must use the same authentication method.

1. EAP-MD5 Authentication Method

EAP-MD5 is an IETF open standard that provides the lowest level of security, because the MD5 hash function is vulnerable to dictionary attacks.

The following figure shows the basic operation flow of the EAP-MD5 authentication method.

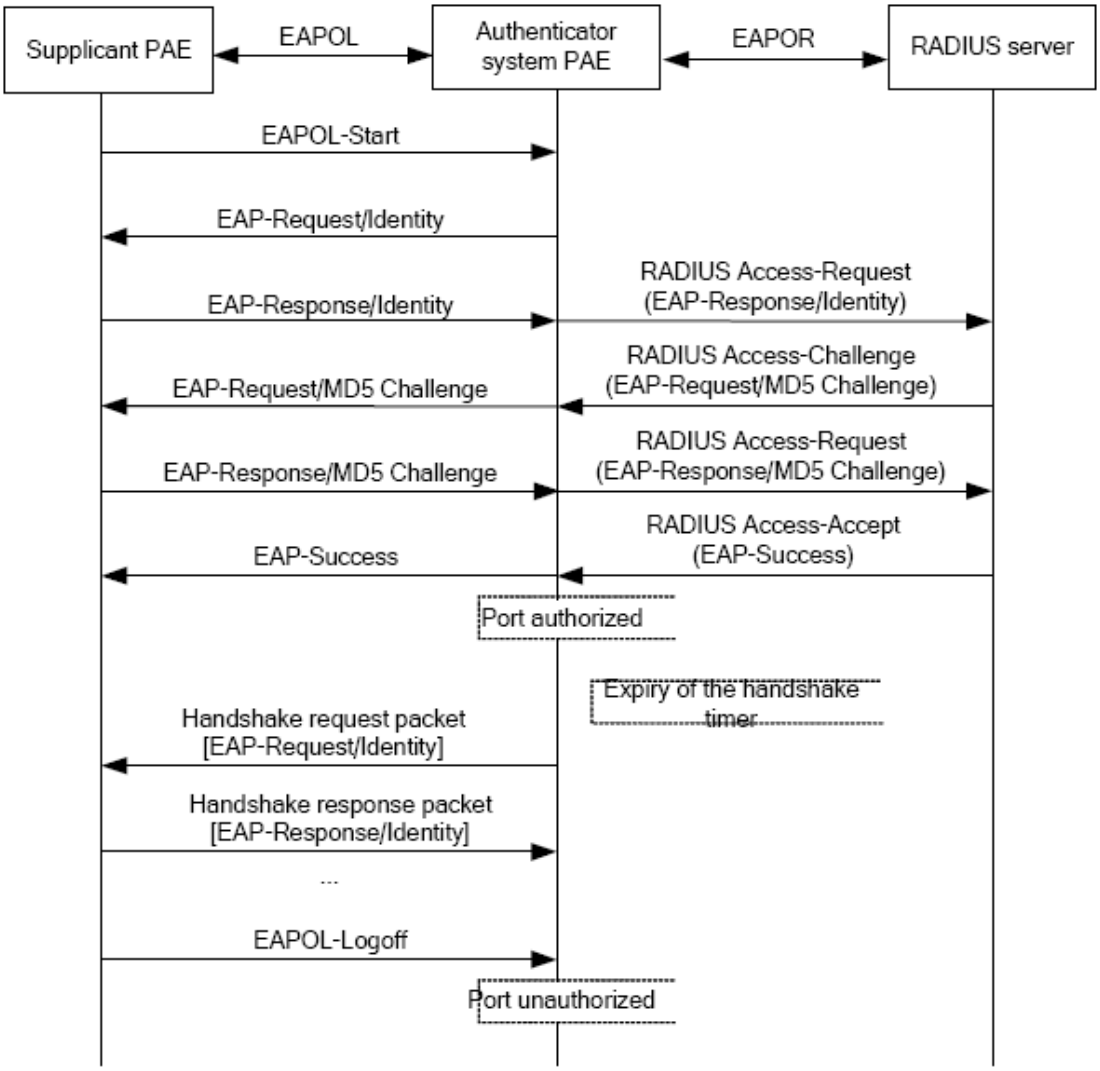


Figure 7-9 802.1x EAP-MD5 Authentication Flow

2. EAP-TLS Authentication Method

EAP-TLS was proposed by Microsoft based on the EAP and TLS protocols. It uses PKI to protect the identity authentication between the supplicant system and the RADIUS server and the dynamically generated session keys, and it requires both the supplicant system and the RADIUS server to have digital certificates for mutual authentication. It was the earliest EAP authentication method used in wireless LANs. Because every user needs a digital certificate, which is difficult to maintain, this method is rarely used in practice. However, it is still one of the most secure EAP standards and is widely supported by wireless LAN hardware and software vendors.

The following figure illustrates the basic operation flow of the EAP-TLS authentication method.

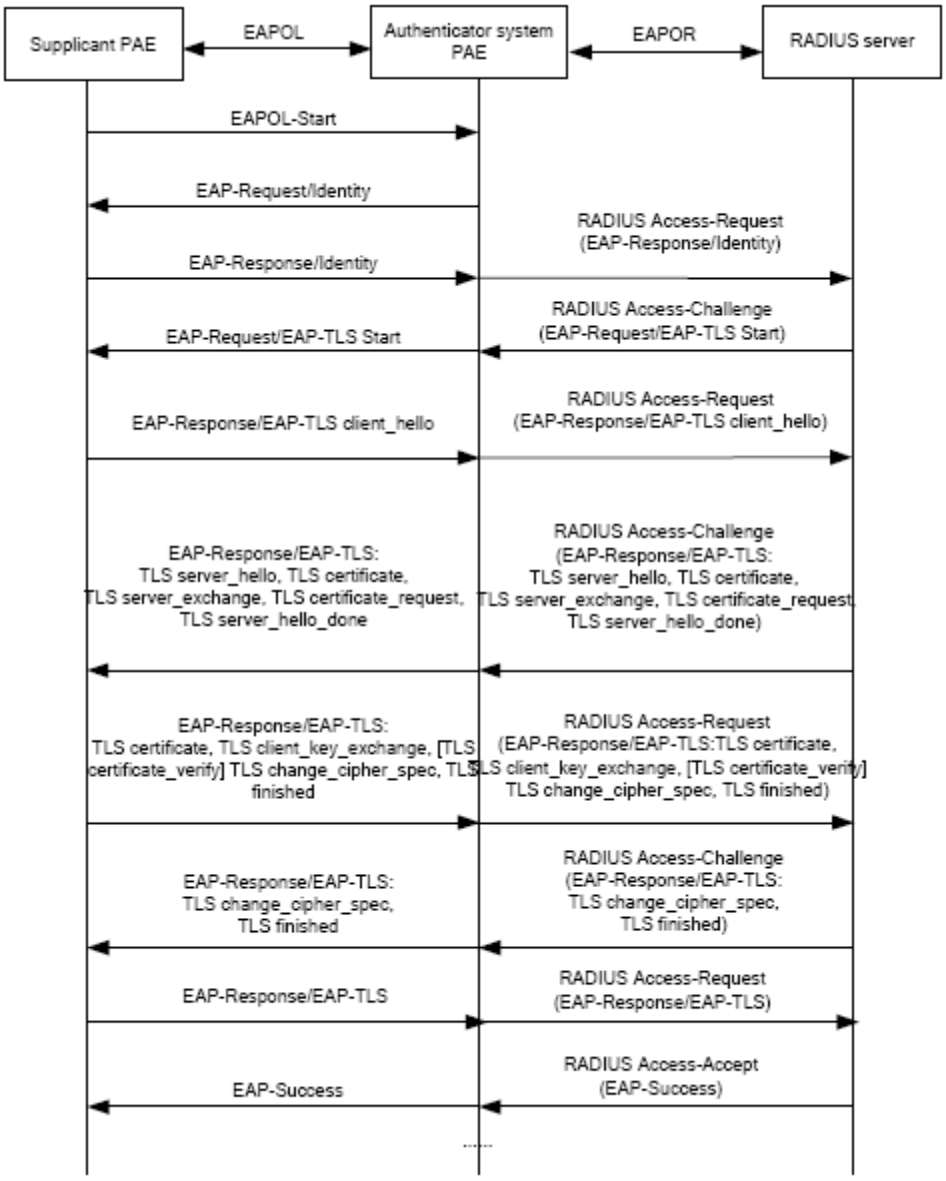


Figure 7-10 802.1x EAP-TLS Authentication Flow

3. EAP-TTLS Authentication Method

EAP-TTLS was developed jointly by Funk Software and Certicom. It provides authentication as strong as EAP-TLS but does not require users to have their own digital certificates; only the RADIUS server needs a digital certificate. User identity is authenticated with passwords transmitted in a securely encrypted tunnel established with the certificate of the authentication server. Any type of authentication request, including EAP, PAP, and MS-CHAPv2, can be carried in TTLS tunnels.

4. PEAP Authentication Method

PEAP was proposed by Cisco, Microsoft, and RSA Security as a recommended open standard. It has long been used in products and provides very good security. Its protocol and security design are similar to those of EAP-TTLS: it uses the server's PKI certificate to establish a secure TLS tunnel that protects user authentication.

The following figure illustrates the basic operation flow of PEAP authentication method.

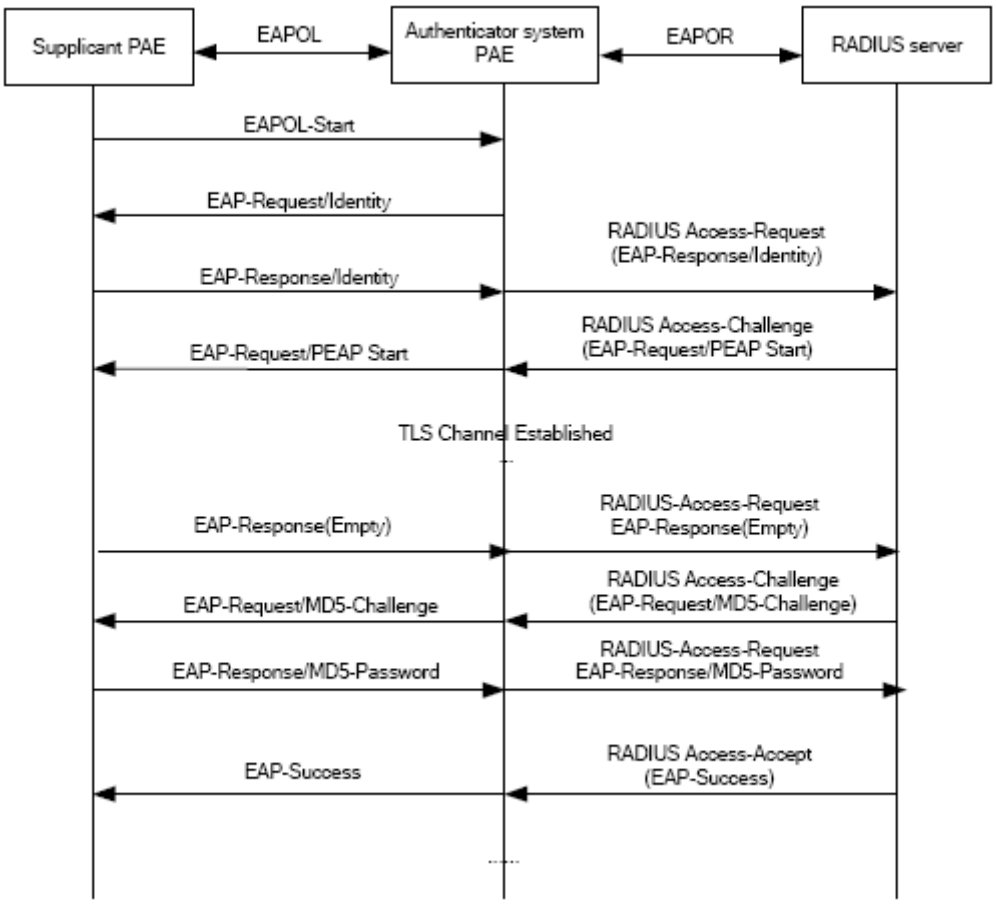


Figure 7-11 802.1x PEAP Authentication Flow

7.3.1.5.2 EAP Termination Mode

In this mode, EAP messages are terminated at the access control device and mapped to RADIUS messages for authentication, authorization, and accounting. The basic operation flow is shown in the figure below.

In EAP termination mode, the access control device and the RADIUS server can use PAP or CHAP authentication. The following figure shows the basic operation flow with CHAP.

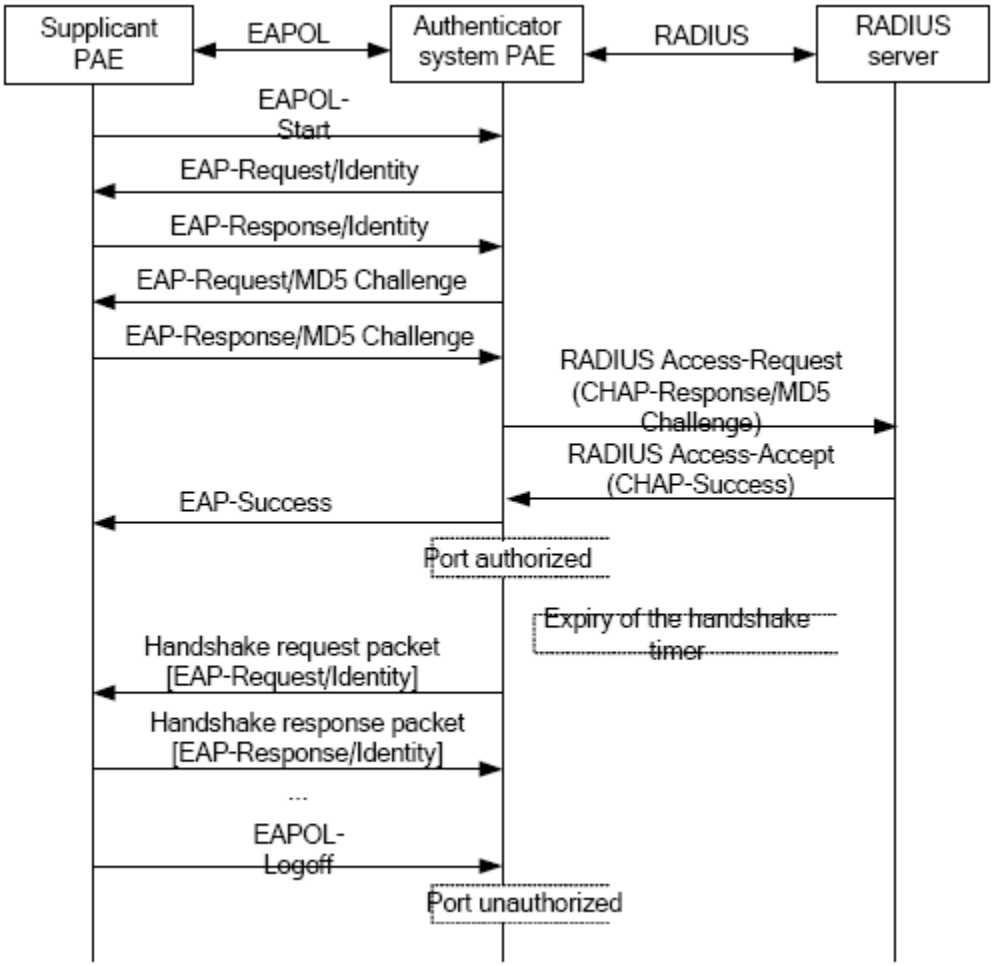


Figure 7-12 802.1x Authentication Flow in EAP Termination Mode

7.3.1.6 The Extension and Optimization of 802.1x

In addition to the port-based access authentication specified by the protocol, the device extends and optimizes 802.1x in its implementation of the EAP relay mode and the EAP termination mode:

- ☞ Supports applications in which one physical port has multiple users.
- ☞ Three access control methods (methods of authenticating users) are available: port-based, MAC-based, and user-based (IP address + MAC address + port).
  - With the port-based method, once the first user on the port passes authentication, all other users can access network resources without authentication. However, once the first user goes offline, the network becomes unavailable to all other users.
  - With the MAC-based method, each user on the port must be authenticated separately, and only authenticated users can access the network. When one user goes offline, the other users are not affected.
  - With the user-based (IP address + MAC address + port) method, users can access limited resources before authentication. This method provides two types of control: standard control and advanced control. User-based standard control does not restrict access to the limited resources, so all users of the port can access them before authentication. User-based advanced control restricts access to the limited resources, so that only specific users of the port can access them before authentication. After passing authentication, users can access all resources.

Note: When private supplicant systems are used, user-based advanced control is recommended to effectively prevent ARP spoofing.

Maximum number of authenticated users: the user-based method supports up to 700 IPv4 users and 1400 IPv6 users. The MAC-based method depends on the rate-limit value of the switch; it supports 4000 authenticated users, but no more than 2000 authenticated users are recommended.

7.3.1.7 VLAN Allocation Features

1. Auto VLAN

The Auto VLAN feature allows the RADIUS server to change the VLAN of the access port based on user information and access device information. When an 802.1x user passes authentication on the server, the RADIUS server sends authorization information to the device. If VLAN assignment is enabled on the RADIUS server, the Access-Accept message must include the following attributes:

- Tunnel-Type = VLAN (13)
- Tunnel-Medium-Type = 802 (6)
- Tunnel-Private-Group-ID = VLANID

VLANID is the VLAN ID, ranging from 1 to 4094. For example, Tunnel-Private-Group-ID = 30 means VLAN 30.

When the switch receives the assigned Auto VLAN information, the current Access port leaves the user-configured VLAN and joins the Auto VLAN.

Auto VLAN does not change or affect the port configuration. However, Auto VLAN has a higher priority than the user-configured VLAN: Auto VLAN takes effect after authentication, and the user-configured VLAN takes effect again only after the user goes offline.

Note: Currently, Auto VLAN can be used only in port-based access control mode and on ports whose link type is Access.

2. Guest VLAN

The Guest VLAN feature allows unauthenticated users to access specified resources.

Before passing 802.1x authentication, the authentication port belongs to a default VLAN (the Guest VLAN), and the user can access the resources in this VLAN without authentication but cannot access resources in other networks. After authentication, the port leaves the Guest VLAN, and the user can access the resources of other networks.

In the Guest VLAN, users can obtain 802.1x supplicant software, upgrade the supplicant, or update other applications (such as antivirus software and operating system patches). The access device adds the port to the Guest VLAN if no supplicant passes authentication within a certain period, for example because the user does not have the dedicated supplicant software or the supplicant version is too old.

When the 802.1x feature is enabled and the Guest VLAN is properly configured, the port is added to the Guest VLAN (in the same way as to an Auto VLAN) if the device has sent more authentication-triggering messages (EAP-Request/Identity) from the port than the upper limit without receiving any response from the supplicant system.

- If the authentication server assigns an Auto VLAN, the port leaves the Guest VLAN and joins the assigned Auto VLAN. When the user goes offline, the port returns to the specified Guest VLAN.
- If the authentication server assigns an Auto VLAN, the port leaves the Guest VLAN and joins the specified VLAN. When the user goes offline, the port returns to the specified Guest VLAN.

7.3.2 802.1x Configuration Task List

802.1x configuration task list:

1. Enable the IEEE 802.1x function

2. Configure access control unit properties

1) Configure the port authentication status

2) Configure the port access control method: MAC-based or port-based

3) Configure extended 802.1x functions

3. Configure supplicant-related properties (optional)

1. Enable the 802.1x function

Command	Explanation
Global Mode	
dot1x enable **no dot1x enable**	Enables the 802.1x function on the switch and ports. The no command disables the 802.1x function.
dot1x privateclient enable **no dot1x privateclient enable**	Forces client software to use the private 802.1x authentication packet format. The no command disables this function.
dot1x user free-resource <*prefix*> <*mask*> **no dot1x user free-resource**	Sets the network resources that unauthenticated 802.1x users can access freely. The no command removes the free resources.
dot1x unicast enable **no dot1x unicast enable**	Enables the 802.1x unicast pass-through function on the switch. The no command disables this function.

2. Configure access control unit properties

1) Configure the port authentication status

Command	Explanation
Port Mode	
dot1x port-control {auto| force-authorized|force-unauthorized } **no dot1x port-control**	Sets the 802.1x authentication mode; the no command restores the default setting.

2) Configure the port access control method

dot1x guest-vlan <*vlanID*> **no dot1x guest-vlan**	Sets the Guest VLAN of the specified port. The no command deletes the Guest VLAN.
dot1x portbased mode single-mode **no dot1x portbased mode single-mode**	Enables single-mode for port-based authentication. The no command disables this function.

3) Configure extended 802.1x functions

Command	Explanation
Global Mode	
dot1x macfilter enable **no dot1x macfilter enable**	Enables the 802.1x address filter function on the switch. The no command disables the 802.1x address filter function.
dot1x macbased port-down-flush **no dot1x macbased port-down-flush**	When MAC-based 802.1x authentication is used and the port goes down, deletes the users that passed authentication on the port. The no command disables this operation.

dot1x accept-mac <*mac-address*> [interface <*interface-name*>] **no dot1x accept-mac <*mac-address*> [interface <*interface-name*>]**	Adds an 802.1x address filter table entry. The no command deletes 802.1x address filter table entries.
dot1x eapor enable **no dot1x eapor enable**	Enables EAP relay authentication on the switch. The no command uses EAP local termination authentication instead.

3. Configure supplicant-related properties

Command	Explanation
Global Mode	
dot1x max-req <count> no dot1x max-req	Sets the number of EAP-Request/MD5 frames sent before the switch restarts authentication when the supplicant does not respond. The no command restores the default setting.
dot1x re-authentication no dot1x re-authentication	Enables periodical supplicant authentication; the no command disables this function.
dot1x timeout quiet-period <seconds> no dot1x timeout quiet-period	Sets the quiet period after a port authentication failure. The no command restores the default value.
dot1x timeout re-authperiod <seconds> no dot1x timeout re-authperiod	Sets the supplicant re-authentication interval; the no command restores the default setting.
dot1x timeout tx-period <seconds> no dot1x timeout tx-period	Sets the interval at which EAP-Request/Identity frames are retransmitted to the supplicant. The no command restores the default setting.
dot1x re-authenticate [interface <interface-name>]	Forces IEEE 802.1x re-authentication immediately (without waiting for the re-authentication timer) on all ports or on the specified port.

7.3.3 802.1x Application Example

7.3.3.1 Guest VLAN Application Example

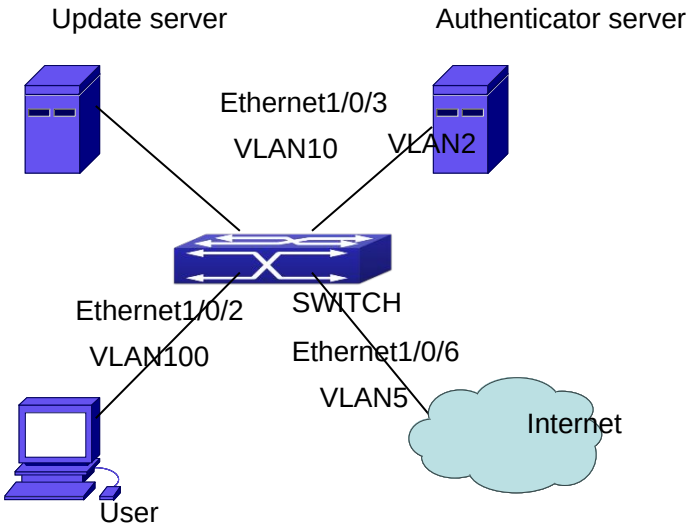


Figure 7-16 The Network Topology of Guest VLAN

Note: In the figures in this section, E2 means Ethernet 1/0/2, E3 means Ethernet 1/0/3, and E6 means Ethernet 1/0/6.

As shown in Figure 7-13, the switch uses 802.1x authentication for network access, with a RADIUS server as the authentication server. Ethernet1/0/2, the port through which the user accesses the switch, belongs to VLAN 100; the authentication server is in VLAN 2; the Update Server, in VLAN 10, is used by the user to download and update supplicant software; and Ethernet1/0/6, the port through which the switch accesses the Internet, is in VLAN 5.

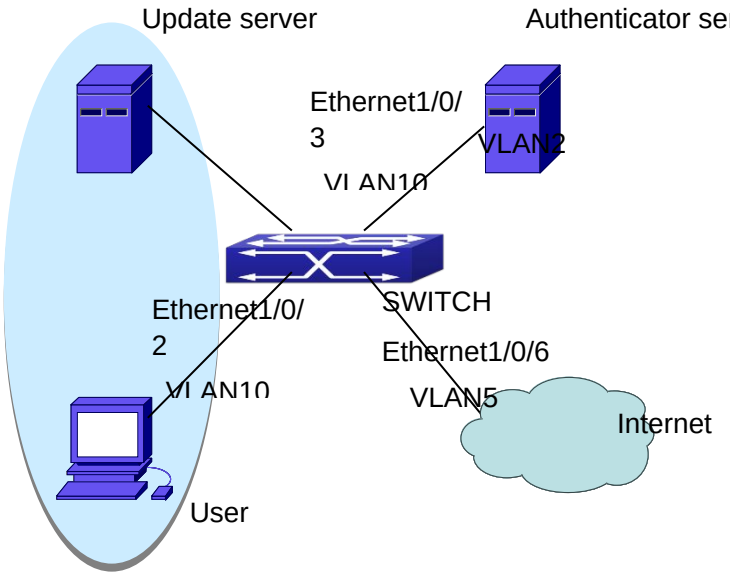


Figure 7-14 User Joining the Guest VLAN

As shown in the figure above, the 802.1x feature is enabled on port Ethernet1/0/2, and VLAN 10 is set as the Guest VLAN of the port. Before the user is authenticated, or if authentication fails, port Ethernet1/0/2 is added to VLAN 10, which allows the user to access the Update Server.

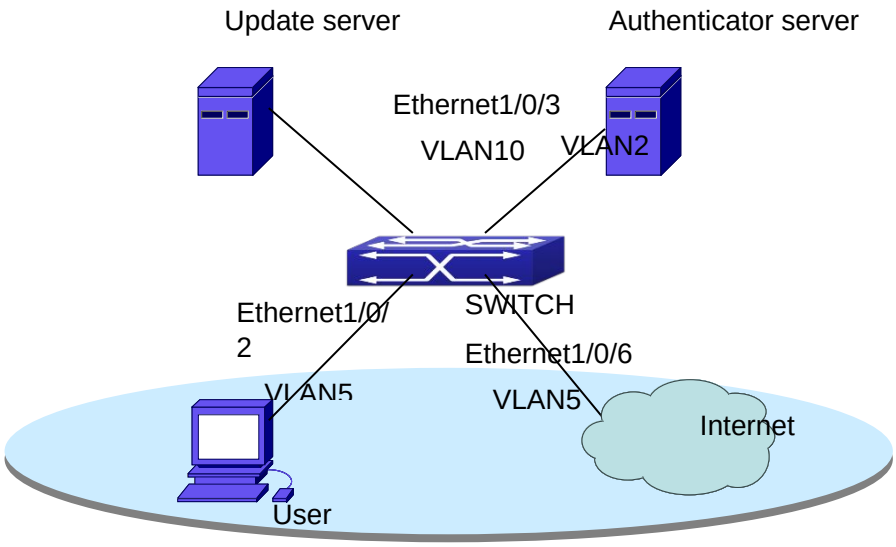


Figure 7-15 User Online, VLAN Offline

As shown in the figure above, after the user passes authentication and goes online, the authentication server assigns VLAN 5, which places the user and Ethernet1/0/6 in VLAN 5 and allows the user to access the Internet.

The configuration steps are as follows:

Configure the RADIUS server.

```
Switch(config)#radius-server authentication host 10.1.1.3
Switch(config)#radius-server accounting host 10.1.1.3
Switch(config)#radius-server key test
Switch(config)#aaa enable
Switch(config)#aaa-accounting enable
```

Create VLAN 100.

```
Switch(config)#vlan 100
```

Enable 802.1x globally.

```
Switch(config)#dot1x enable
```

Enable 802.1x on port Ethernet1/0/2.

```
Switch(config)#interface ethernet1/0/2
Switch(Config-If-Ethernet1/0/2)#dot1x enable
```

Set the link type of the port to access.

```
Switch(Config-If-Ethernet1/0/2)#switch-port mode access
```

Set the access control method of the port to port-based.

```
Switch(Config-If-Ethernet1/0/2)#dot1x port-method portbased
```

Set the port control mode of the port to auto.

```
Switch(Config-If-Ethernet1/0/2)#dot1x port-control auto
```

Set the Guest VLAN of the port to VLAN 100.

```
Switch(Config-If-Ethernet1/0/2)#dot1x guest-vlan 100
Switch(Config-If-Ethernet1/0/2)#exit
```

Use the **show running-config** or **show interface ethernet1/0/2** command to check the Guest VLAN configuration. When there is no online user, no failed authentication, and no successful logoff, and more authentication-triggering messages (EAP-Request/Identity) have been sent than the upper limit, use the **show vlan id 100** command to check whether the Guest VLAN configured on the port has taken effect.

7.3.3.2 IPv4 RADIUS Application Example

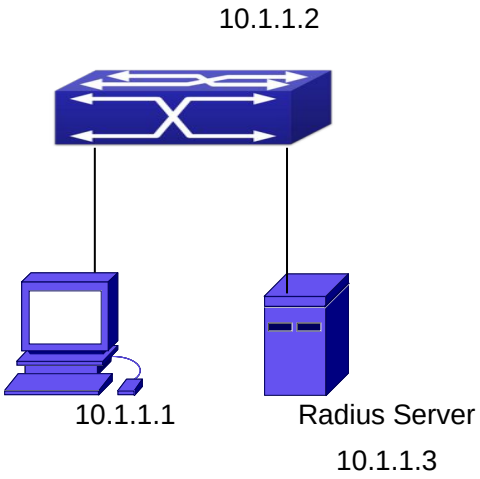


Figure 7-176 IEEE 802.1x Configuration Example Topology

The PC is connected to port 1/0/2 of the switch, on which IEEE 802.1x authentication is enabled with the default MAC-based access method. The IP address of the switch is 10.1.1.2. A port other than 1/0/2 connects to the RADIUS authentication server, whose IP address is 10.1.1.3 and which uses the default port 1812 for authentication and port 1813 for accounting. IEEE 802.1x client software is installed on the PC for IEEE 802.1x authentication.

The configuration steps are as follows:

```
Switch(config)#interface vlan 1
Switch(Config-if-vlan1)#ip address 10.1.1.2 255.255.255.0
Switch(Config-if-vlan1)#exit
Switch(config)#radius-server authentication host 10.1.1.3
Switch(config)#radius-server accounting host 10.1.1.3
Switch(config)#radius-server key test
Switch(config)#aaa enable
Switch(config)#aaa-accounting enable
Switch(config)#dot1x enable
Switch(config)#interface ethernet 1/0/2
Switch(Config-Ethernet1/0/2)#dot1x enable
Switch(Config-Ethernet1/0/2)#dot1x port-control auto
Switch(Config-Ethernet1/0/2)#exit
```

7.3.3.3 IPv6 RADIUS Application Example

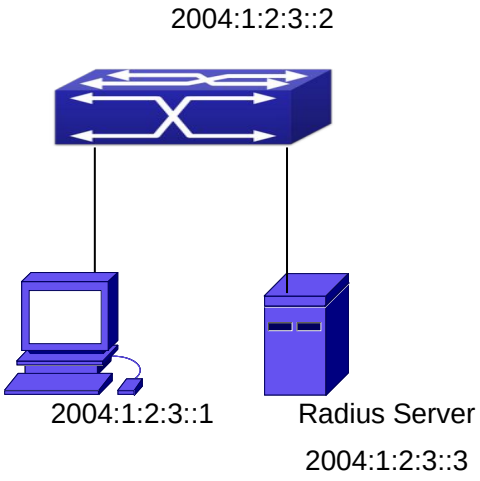


Figure 7-17 IPv6 RADIUS

The computer is connected to port 1/0/2 of the switch, on which IEEE 802.1x is enabled with MAC-based authentication. The IP address of the switch is 2004:1:2:3::2, and a port other than 1/0/2 connects the switch to the RADIUS authentication server, whose IP address is 2004:1:2:3::3. The default ports 1812 and 1813 are used for authentication and accounting, respectively. IEEE 802.1x client software is installed on the computer for IEEE 802.1x authentication.

The configuration steps are as follows:

```
Switch(config)#interface vlan 1
Switch(Config-if-vlan1)#ipv6 address 2004:1:2:3::2/64
Switch(Config-if-vlan1)#exit
Switch(config)#radius-server authentication host 2004:1:2:3::3
Switch(config)#radius-server accounting host 2004:1:2:3::3
Switch(config)#radius-server key test
Switch(config)#aaa enable
Switch(config)#aaa-accounting enable
Switch(config)#dot1x enable
Switch(config)#interface ethernet 1/0/2
Switch(Config-If-Ethernet1/0/2)#dot1x enable
Switch(Config-If-Ethernet1/0/2)#dot1x port-control auto
Switch(Config-If-Ethernet1/0/2)#exit
```

7.3.4 802.1x Troubleshooting

If 802.1x is configured on a port with authentication set to auto, but the switch cannot enter the authenticated state after the user runs the 802.1x supplicant software, the possible causes and solutions are as follows:

- ☞ If 802.1x cannot be enabled on a port, make sure that the port is not configured for MAC binding or port aggregation. These functions must be disabled before 802.1x authentication can be enabled.
- ☞ If the switch is configured correctly but authentication still fails, verify the connectivity between the switch and the RADIUS server and between the switch and the 802.1x client, and check the port and VLAN configuration of the switch.
- ☞ Check the event log on the RADIUS server for possible causes. The event log records failed logins and the reasons for them. If the log indicates a wrong authenticator password, modify the radius-server key parameter; if it indicates no such authenticator, add the authenticator to the RADIUS server; if it indicates no such login user, verify and re-enter the user login ID and password.

7.4 MAC and IP Number Limitation on Ports and VLANs

7.4.1 Introduction to MAC and IP Number Limitation on Ports and VLANs

The MAC address table maps destination MAC addresses to switch ports. It contains two kinds of MAC addresses: static and dynamic. Static MAC addresses are configured by users, have the highest priority (they are not overwritten by dynamic MAC addresses), and are always valid. Dynamic MAC addresses are learned by the switch while forwarding data frames and are valid only for a limited time. When the switch receives a data frame to be forwarded, it learns the source MAC address of the frame and maps it to the receiving port, and then looks up the destination MAC address in the MAC address table. If a matching entry is found, the switch forwards the frame through the corresponding port; otherwise, it broadcasts the frame in the VLAN to which the frame belongs. If a dynamically learned MAC address receives no traffic for a long time, the switch deletes it from the MAC address table.

The switch supports both static configuration and dynamic learning of MAC addresses, so each port can have multiple static and dynamically learned MAC addresses and can forward traffic to known MAC addresses. When a MAC address ages out, traffic to it is broadcast. Without a limit, each port can configure or learn MAC addresses until the hardware table entries are exhausted. To prevent a port from having too many MAC addresses, the number of MAC addresses on a port should be limited.

Without a limit, the number of IP addresses on a VLAN interface is bounded only by the maximum number of users on the interface, which is also the maximum number of ARP and ND entries, and no command controls the number of these entries. To improve security and controllability, the number of MAC addresses on each port and the number of ARP and ND entries on each VLAN interface need to be controlled: the number of static or dynamic MAC addresses on a port, and the number of users in each VLAN, must not exceed the configured values.

Limiting the number of MAC and ARP entries can prevent DoS attacks to a certain extent. Malicious users who frequently spoof MAC or ARP can easily fill the MAC and ARP tables of the switch and launch successful DoS attacks.

This function therefore limits the number of MAC addresses on ports and the number of ARP and ND entries on ports and VLANs through configuration commands.

Limiting the number of dynamic MAC and IP addresses on a port:

1. Dynamic MAC limit: if the number of MAC addresses dynamically learned on the port is greater than or equal to the configured maximum, MAC learning is disabled on the port; otherwise, the port continues learning.

2. Dynamic IP limit: if the number of dynamically learned ARP and ND entries on the port is greater than or equal to the configured maximum, ARP and ND learning is disabled on the port; otherwise, the port continues learning.

Limiting the number of MAC, ARP, and ND entries in a VLAN:

1. Dynamic MAC limit: if the number of MAC addresses dynamically learned in the VLAN is greater than or equal to the configured maximum, MAC learning is disabled on all ports in the VLAN; otherwise, all ports in the VLAN (except special ports) continue learning.

2. Dynamic IP limit: if the number of dynamically learned ARP and ND entries is greater than or equal to the configured maximum, the VLAN does not learn any new ARP or ND entries; otherwise, learning continues.

7.4.2 MAC and IP Number Limitation Configuration Task Sequence

1. Enable MAC and IP number limitation on ports
2. Enable MAC and IP number limitation in VLANs
3. Configure the dynamic MAC query timeout
4. Configure the port violation mode
5. Display and debug MAC and IP number limitation information

1. Enable MAC and IP number limitation on ports

Command	Explanation
Port Mode	
switchport mac-address dynamic maximum <*value*> **no switchport mac-address dynamic maximum**	Enables MAC number limitation on the port. The no command disables it.
switchport arp dynamic maximum <*value*> **no switchport arp dynamic maximum**	Enables ARP number limitation on the port. The no command disables it.
switchport nd dynamic maximum <*value*> **no switchport nd dynamic maximum**	Enables ND number limitation on the port. The no command disables it.

2. Enable MAC and IP number limitation in VLANs

Command	Explanation
VLAN Mode	
vlan mac-address dynamic maximum <*value*> **no vlan mac-address dynamic maximum**	Enables MAC number limitation in the VLAN. The no command disables it.
VLAN Interface Mode	
ip arp dynamic maximum <*value*> **no ip arp dynamic maximum**	Enables ARP number limitation in the VLAN. The no command disables it.
ipv6 nd dynamic maximum <*value*> **no ipv6 nd dynamic maximum**	Enables ND (neighbor) number limitation in the VLAN. The no command disables it.

3. Configure the dynamic MAC query timeout

Command	Explanation
Global Mode	

mac-address query timeout <seconds>	Configures the timeout for querying dynamic MAC addresses.

4. Configure the port violation mode

Command	Explanation
Port Mode	
switchport mac-address violation {protect | shutdown} [recovery <5-3600>] **no switchport mac-address violation**	Sets the violation mode of the port. The no command restores the violation mode to **protect**.

5. Display and debug MAC and IP number limitation information

Command	Explanation
Admin Mode	
show mac-address dynamic count {vlan <*vlan-id*> | interface ethernet <portName> }	Displays the number of dynamic MAC addresses on the specified port or in the specified VLAN.
show arp-dynamic count {vlan <vlan-id> | interface ethernet <*portName*> }	Displays the number of dynamic ARP entries on the specified port or in the specified VLAN.
show nd-dynamic count {vlan <*vlan-id*> | interface ethernet <*portName*> }	Displays the number of dynamic ND (neighbor) entries on the specified port or in the specified VLAN.
debug switchport mac count **no debug switchport mac count**	Displays debug information about MAC number limitation on ports.
debug switchport arp count **no debug switchport arp count**	Displays debug information about ARP number limitation on ports.
debug switchport nd count **no debug switchport nd count**	Displays debug information about ND (neighbor) number limitation on ports.
debug vlan mac count **no debug vlan mac count**	Displays debug information about MAC number limitation in VLANs.
debug ip arp count **no debug ip arp count**	Displays debug information about ARP number limitation in VLANs.
debug ipv6 nd count **no debug ipv6 nd count**	Displays debug information about ND number limitation in VLANs.

7.4.3 MAC and IP Number Limitation Typical Example

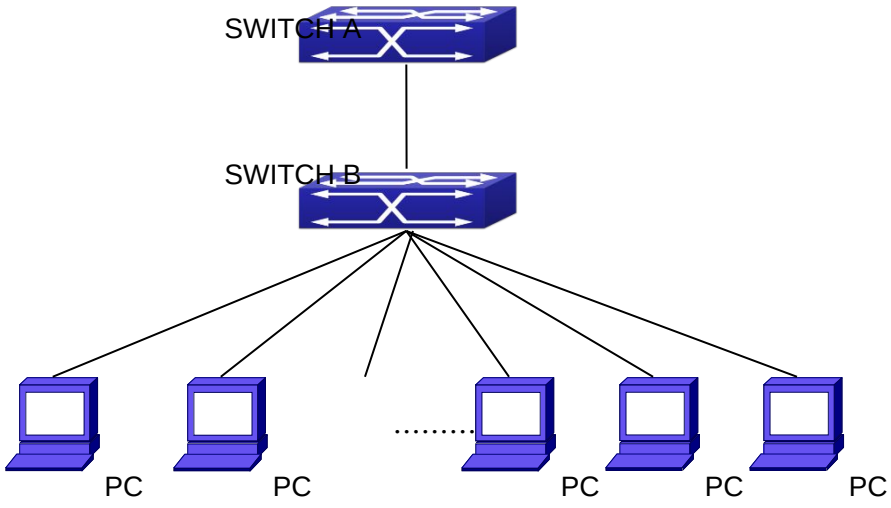


Figure 7-18 Typical Configuration Example of MAC and IP Number Limitation on Ports and VLANs

In the network topology above, Switch B connects to many PCs. Before MAC and IP number limitation is enabled, if the system hardware has no other limit, Switch A and Switch B can learn the MAC, ARP, and ND entries of all the PCs. Malicious users who frequently spoof MAC or ARP can easily fill the MAC and ARP tables of the switch and launch successful DoS attacks. Limiting the MAC, ARP, and ND entries can prevent such DoS attacks to a certain extent.

On port 1/0/1 of Switch A, set the maximum number of dynamically learned MAC addresses to 20, dynamic ARP entries to 20, and ND entries to 10. In VLAN 1, set the maximum number of dynamic MAC addresses to 30, dynamic ARP entries to 30, and ND entries to 20.

Switch A configuration task sequence:

```
Switch (config)#interface ethernet 1/0/1
Switch (Config-If-Ethernet1/0/1)#switchport mac-address dynamic maximum 20
Switch (Config-If-Ethernet1/0/1)#switchport arp dynamic maximum 20
Switch (Config-If-Ethernet1/0/1)#switchport nd dynamic maximum 10
Switch (Config-if-Vlan1)#vlan mac-address dynamic maximum 30
```

7.4.4 MAC and IP Number Limitation Troubleshooting

MAC and IP number limitation on ports and VLANs is disabled by default; enable it to limit the number of users accessing the network. If MAC number limitation cannot be configured, check whether spanning tree, dot1x, or TRUNK is running on the switch and whether the port is configured as a MAC-binding port. MAC number limitation is mutually exclusive with these configurations, so make sure that these functions are disabled on the port before enabling MAC number limitation.

If all configurations are correct, after enabling the function, use the debug commands to check the details of each limitation and verify that the function works correctly. If the problem persists, contact EnGenius technical support.

7.5 AM

7.5.1 Introduction to AM Function

AM (Access Management) works as follows: when the switch receives an IP or ARP packet, it compares information extracted from the packet (such as the source IP address or source MAC-IP address) with the configured hardware address pool. If an entry in the pool matches, the packet is forwarded; otherwise, it is dropped. Source-IP-based AM is supplemented by source-MAC-IP-based AM because the IP address of a host can be changed: with IP binding alone, a user can change the host IP address to a permitted IP address and have the packets of the host forwarded by the switch. Because a MAC-IP pair uniquely identifies a host, binding MAC-IP to a host prevents users from maliciously modifying the host IP address to have their packets forwarded by the switch.

With the interface binding attribute of AM, network administrators can bind the IP (or MAC-IP) addresses of legal users to a specified interface. Then only packets sent by users with the specified IP (or MAC-IP) addresses can be forwarded through that interface, which strengthens network security monitoring.

7.5.2 AM Function Configuration Task List

1. Enable the AM function
2. Enable AM on an interface
3. Configure forwarding IP addresses
4. Configure forwarding MAC-IP addresses
5. Delete all configured IP addresses, MAC-IP addresses, or both
6. Display AM configuration information

1. Enable the AM function

Command	Explanation
Global Mode	
am enable **no am enable**	Enables or disables the AM function globally.

2. Enable AM on an interface

Command	Explanation
Port Mode	
am port **no am port**	Enables or disables AM on the port. When AM is enabled on a port, no IP or ARP packets are forwarded by default.

3. Configure forwarding IP addresses

Command	Explanation
Port Mode	
am ip-pool <*ip-address*> <*num*> **no am ip-pool <*ip-address*> <*num*>**	Configures the forwarding IP addresses of the port.

4. Configure forwarding MAC-IP addresses

Command	Explanation
Port Mode	
am mac-ip-pool <*mac-address*> <*ip-address*> **no am mac-ip-pool <*mac-address*> <*ip-address*>**	Configures the forwarding MAC-IP addresses of the port.

5. Delete all configured IP addresses, MAC-IP addresses, or both

Command	Explanation
Global Mode	
no am all [ip-pool|mac-ip-pool]	Deletes the MAC-IP address pools, the IP address pools, or both, configured by all users.

6. Display AM configuration information

Command	Explanation
Global Mode	
show am [interface <*interface-name*>]	Displays the AM configuration of one port or all ports.

7.5.3 AM Function Example

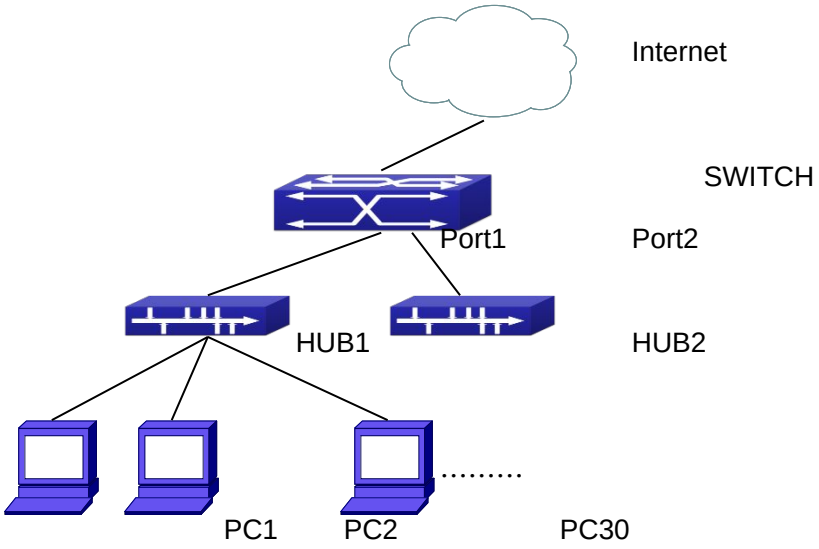


Figure 7-19 Typical AM Configuration Example

In the topology above, 30 PCs are aggregated by HUB1 and connected to port 1 of the switch. The IP addresses of these 30 PCs range from 100.10.10.1 to 100.10.10.30. For security, the system administrator treats only users with IP addresses in this range as legal users. The switch forwards packets from legal users only and drops packets from other users.

To meet these requirements, configure the switch as follows:

```
Switch(config)#am enable
Switch(config)#interface ethernet1/0/1
Switch(Config-If-Ethernet 1/0/1)#am port
Switch(Config-If-Ethernet 1/0/1)#am ip-pool 10.10.10.1 10
```

7.5.4 AM Function Troubleshooting

The AM function is disabled by default. AM settings can be configured after the function is enabled.

Use the **show am** command to view the current AM configuration, such as whether AM is enabled and the AM information on each interface. Use the **show am [interface *<interface-name>*]** command to check the AM configuration of a specific interface.

If an operation error occurs, the system displays a detailed prompt.

7.6 Security Feature

7.6.1 Introduction to Security Feature

DoS (Denial of Service) is a simple but effective destructive attack on the Internet. A server under DoS attack keeps processing the attacker's packets and drops normal user packets, which results in denial of service and, worse, may lead to leakage of sensitive server data.

Security features refer to applications such as protocol checks that protect servers from attacks such as DoS. A protocol check drops packets that match specified conditions. The security features provide several simple and effective protections against DoS attacks without affecting the wire-speed forwarding performance of the switch.

7.6.2 Security Feature Configuration

7.6.2.1 IP Spoofing Prevention Configuration Task Sequence

1. Enable the IP spoofing prevention function.

Command	Explanation
Global Mode	
[no] dosattack-check srcip-equal-dstip enable	Enables or disables checking whether the source IP address of a packet is the same as its destination IP address.

7.6.2.2 TCP Unauthorized Flag Attack Prevention Configuration Task Sequence

1. Enable the TCP unauthorized flag attack prevention function
2. Enable the IPv4 fragment check function

Command	Explanation
Global Mode	
[no] dosattack-check tcp-flags enable	Enables or disables the TCP flag check.
[no] dosattack-check ipv4-first-fragment enable	Enables or disables the IPv4 fragment check. This command has no effect on its own; however, if this function is not enabled, the switch does not drop IPv4 fragments that contain unauthorized TCP flags.

7.6.2.3 Anti-Port-Spoofing Configuration Task Sequence

1 . Enable the anti-port-spoofing function

Command	Explanation
Global Mode	
[no] dosattack-check srcport-equal-dstport enable	Enables or disables the anti-port-spoofing function.
[no] dosattack-check ipv4-first-fragment enable	Enables or disables the IPv4 fragment check. This command has no effect on its own; however, if this function is not enabled, the switch does not drop IPv4 fragments whose source port is the same as the destination port.

7.6.2.4 TCP Fragment Attack Prevention Configuration Task Sequence

1. Enable the TCP fragment attack prevention function
2. Configure the minimum permitted TCP header length of packets

Command	Explanation
Global Mode	
[no] dosattack-check tcp-fragment enable	Enables or disables the TCP fragment attack prevention function.
dosattack-check tcp-header <*size*>	Configures the minimum permitted TCP header length of packets. This command takes effect only when **dosattack-check tcp-fragment enable** is configured.

7.6.2.5 ICMP Fragment Attack Prevention Configuration Task Sequence

1. Enable the ICMP fragment attack prevention function
2. Configure the maximum permitted ICMPv4 payload length

3. Configure the maximum permitted ICMPv6 payload length

Command	Explanation
Global Mode	
[no] dosattack-check icmp-attacking enable	Enables or disables the ICMP fragment attack prevention function.
dosattack-check icmpv4-size <*size*>	Configures the maximum permitted ICMPv4 payload length. This command takes effect only when **dosattack-check icmp-attacking enable** is configured.
dosattack-check icmpv6-size <*size*>	Configures the maximum permitted ICMPv6 payload length. This command takes effect only when **dosattack-check icmp-attacking enable** is configured.

7.6.3 Security Feature Example

Scenario:

Requirements: the switch must not forward packets whose source IP address is the same as the destination IP address, or packets whose source port is the same as the destination port. Only ping with default options is allowed in the IPv4 network; that is, ICMP request packets cannot be fragmented, and their payload length is normally less than 100.

Configuration procedure:

```
Switch(config)# dosattack-check srcip-equal-dstip enable
Switch(config)# dosattack-check srcport-equal-dstport enable
Switch(config)# dosattack-check ipv4-first-fragment enable
Switch(config)# dosattack-check icmp-attacking enable
Switch(config)# dosattack-check icmpV4-size 100
```

7.7 TACACS+

7.7.1 Introduction to TACACS+

TACACS+ (Terminal Access Controller Access-Control System Plus) is a protocol similar to RADIUS for controlling terminal access to the network. It also provides the three independent functions of authentication, authorization, and accounting. Unlike RADIUS, TACACS+ uses TCP as its transport protocol and encrypts the packet (everything except the standard header), which makes its transmission and encryption more reliable and better suited to security control.

Based on TACACS+ (version 1.78), the switch provides TACACS+ authentication: when a user logs in, for example through Telnet, the username and password can be authenticated through TACACS+.

7.7.2 TACACS+ Configuration Task List

1. Configure the TACACS+ authentication key
2. Configure the TACACS+ server
3. Configure the TACACS+ authentication timeout
4. Configure the IP address of the TACACS+ NAS

1. Configure the TACACS+ authentication key

Command	Explanation
Global Mode	

tacacs-server key {0 | 7}<*string*> **no tacacs-server key**	Configures the TACACS+ server key. The **no tacacs-server key** command deletes the key.

2. Configure the TACACS+ server

Command	Explanation
Global Mode	
tacacs-server authentication host <*ip-address*> [port <*port-number*>] [timeout <*seconds*>] [key {0 | 7} <*string*>] [primary] **no tacacs-server authentication host <*ip-address*>**	Configures the IP address, listening port number, timeout value, and key string of the TACACS+ server. The no command deletes the TACACS+ authentication server.

3. Configure the TACACS+ authentication timeout

Command	Explanation
Global Mode	
tacacs-server timeout <*seconds*> **no tacacs-server timeout**	Configures the authentication timeout for the TACACS+ server. The **no tacacs-server timeout** command restores the default configuration.

4. Configure the IP address of the TACACS+ NAS

Command	Explanation
Global Mode	
tacacs-server nas-ipv4 <*ip-address*> **no tacacs-server nas-ipv4**	Configures the source IP address of TACACS+ packets sent by the switch.

7.7.3 TACACS+ Typical Example

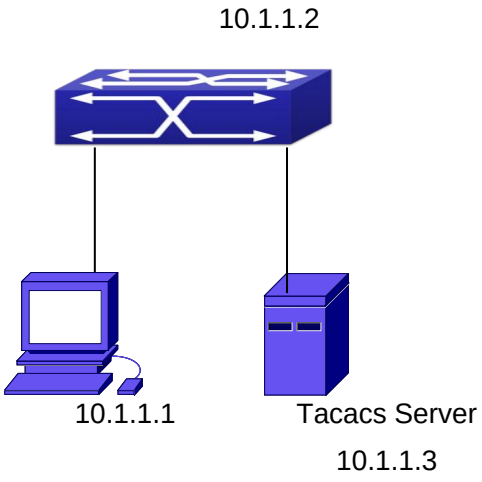


Figure 7-200 TACACS+ Configuration

A computer is connected to the switch, whose IP address is 10.1.1.2. The switch is connected to a TACACS+ authentication server with IP address 10.1.1.3 and the default authentication port 49. Telnet login authentication on the switch is set to tacacs local, so that Telnet users are authenticated through the TACACS+ authentication server.

```
Switch(config)#interface vlan 1
Switch(Config-if-vlan1)#ip address 10.1.1.2 255.255.255.0
Switch(Config-if-vlan1)#exit
Switch(config)#tacacs-server authentication host 10.1.1.3
Switch(config)#tacacs-server key test
Switch(config)#authentication line vty login tacacs
```

7.7.4 TACACS+ Troubleshooting

When TACACS+ is configured and used, authentication may fail because of physical connection failures or configuration errors. Ensure the following:

- ☞ First, the physical connection to the TACACS+ server is in good condition.
- ☞ Second, all interfaces and link protocols are up (use the **show interface** command).
- ☞ Third, the TACACS+ key configured on the switch matches the key configured on the TACACS+ server.
- ☞ Finally, the switch connects to the correct TACACS+ server.

7.8 RADIUS

7.8.1 Introduction to RADIUS

7.8.1.1 AAA and RADIUS Introduction

AAA stands for Authentication, Authorization, and Accounting. It provides a consistent framework for secure network management. Through its three functions, the framework controls access to the network: who can access the network device, which access level a user has, and accounting for network resource usage.

RADIUS (Remote Authentication Dial-In User Service) is a distributed client/server protocol for information exchange. The RADIUS client usually runs on a network device and implements AAA together with the 802.1x protocol. The RADIUS server maintains the AAA database and communicates with the RADIUS client through the RADIUS protocol. RADIUS is the most commonly used protocol in the AAA framework.

7.8.1.2 RADIUS Message Structure

RADIUS uses UDP to carry protocol packets. The packet format is shown below.

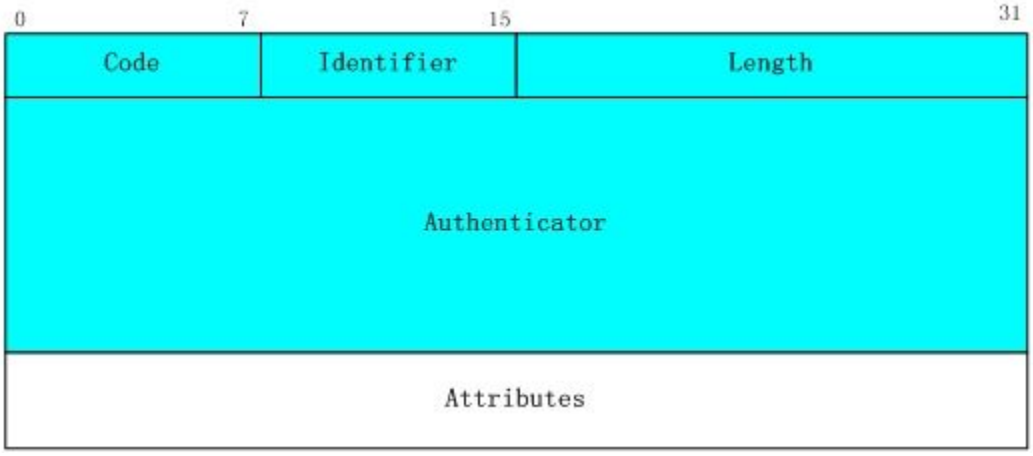


Figure 7-211 RADIUS Message Structure

Code field (1 octet): the type of the RADIUS packet. The values of the Code field are as follows:

1 Access-Request

2 Access-Accept

3 Access-Reject

4 Accounting-Request

5 Accounting-Response

11 Access-Challenge

Identifier field (1 octet): Identifier for the request and answer packets.

Length field (2 octets): the length of the entire RADIUS packet, including the Code, Identifier, Length, Authenticator, and Attributes fields.

Authenticator field (16 octets): used to validate packets received from the RADIUS server; it can also carry encrypted passwords. There are two kinds: the Request Authenticator and the Response Authenticator.

Attribute field: carries detailed AAA information. An attribute consists of the Type, Length, and Value fields.

☞ Type field (1 octet): the type of the attribute, as listed below:

Type	Attribute	Type	Attribute
1	User-Name	23	Framed-IPX-Network
2	User-Password	24	State
3	CHAP-Password	25	Class
4	NAS-IP-Address	26	Vendor-Specific
5	NAS-Port	27	Session-Timeout
6	Service-Type	28	Idle-Timeout
7	Framed-Protocol	29	Termination-Action
8	Framed-IP-Address	30	Called-Station-Id
9	Framed-IP-Netmask	31	Calling-Station-Id
10	Framed-Routing	32	NAS-Identifier
11	Filter-Id	33	Proxy-State
12	Framed-MTU	34	Login-LAT-Service
13	Framed-Compression	35	Login-LAT-Node
14	Login-IP-Host	36	Login-LAT-Group
15	Login-Service	37	Framed-AppleTalk-Link
16	Login-TCP-Port	38	Framed-AppleTalk-Network
17	(unassigned)	39	Framed-AppleTalk-Zone
18	Reply-Message	40-59	(reserved for accounting)
19	Callback-Number	60	CHAP-Challenge
20	Callback-Id	61	NAS-Port-Type
21	(unassigned)	62	Port-Limit
22	Framed-Route	63	Login-LAT-Port

☞ Length field (1 octet): the length of the attribute in octets, including the Type, Length, and Value fields.

☞ Value field: the value of the attribute; its content and format are determined by the type and length of the attribute.

7.8.2 RADIUS Configuration Task List

1. Enable the authentication and accounting functions
2. Configure the RADIUS authentication key
3. Configure the RADIUS server
4. Configure RADIUS service parameters
5. Configure the IP address of the RADIUS NAS

1. Enable the authentication and accounting functions

Command	Explanation
Global Mode	

aaa enable **no aaa enable**	Enables AAA authentication. The no command disables AAA authentication.
aaa-accounting enable **no aaa-accounting enable**	Enables AAA accounting. The no command disables AAA accounting.
aaa-accounting update {enable|disable}	Enables or disables the accounting update function.

2. Configure the RADIUS authentication key

Command	Explanation
Global Mode	
radius-server key {0 | 7} <*string*> **no radius-server key**	Configures the encryption key for the RADIUS server. The no command removes the configured key.

3. Configure the RADIUS server

Command	Explanation
Global Mode	
radius-server authentication host {<*ipv4-address*> | <*ipv6-address*>} [port <*port-number*>] [key *{0* | 7} <string>] [primary] [access-mode {dot1x | telnet}] **no radius-server authentication host {<*ipv4-address*> | <*ipv6-address*>**	Specifies the IPv4/IPv6 address and port number of the RADIUS authentication server and whether it is the primary server. The no command deletes the RADIUS authentication server.
radius-server accounting host {<ipv4-address> | <ipv6-address>} [port <port-number>] [key {*0* | 7} <string>] [primary] **no radius-server accounting host {<ipv4-address> | <ipv6-address>}**	Specifies the IPv4/IPv6 address and port number of the RADIUS accounting server and whether it is the primary server. The no command deletes the RADIUS accounting server.

4. Configure RADIUS service parameters

Command	Explanation
Global Mode	
radius-server dead-time <*minutes*> **no radius-server dead-time**	Configures the interval after which a RADIUS server that went down is considered available again. The no command restores the default configuration.
radius-server retransmit <*retries*> **no radius-server retransmit**	Configures the number of retransmissions of RADIUS packets. The no command restores the default configuration.
radius-server timeout <*seconds*> **no radius-server timeout**	Configures the timeout for the RADIUS server. The no command restores the default configuration.
radius-server accounting-interim-update timeout <*seconds*> **no radius-server accounting-interim-update timeout**	Configures the accounting update interval. The no command restores the default configuration.

5. Configure the IP address of the RADIUS NAS

Command	Explanation
Global Mode	
radius nas-ipv4 <*ip-address*> **no radius nas-ipv4**	Configures the source IP address of RADIUS packets sent by the switch.
radius nas-ipv6 <*ipv6-address*> **no radius nas-ipv6**	Configures the source IPv6 address of RADIUS packets sent by the switch.

7.8.3 RADIUS Typical Examples

7.8.3.1 IPv4 RADIUS Example

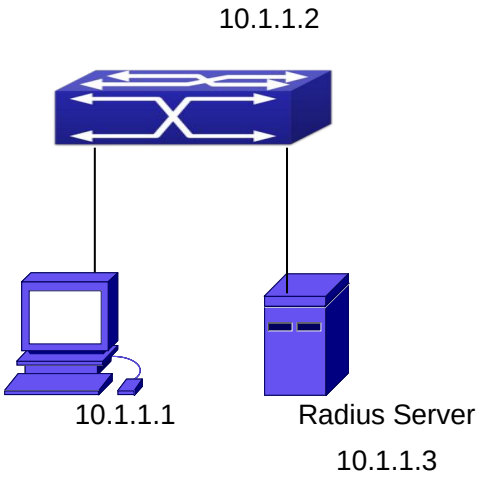


Figure 7-182 The Topology of IEEE802.1x configuration

A computer is connected to the switch, whose IP address is 10.1.1.2. The switch is connected to a RADIUS authentication server through a port other than Ethernet1/0/2. The IP address of the server is 10.1.1.3; the default authentication port is 1812, and the default accounting port is 1813.

The configuration steps are as follows:

```
Switch(config)#interface vlan 1
Switch(Config-if-vlan1)#ip address 10.1.1.2 255.255.255.0
Switch(Config-if-vlan1)#exit
Switch(config)#radius-server authentication host 10.1.1.3
Switch(config)#radius-server accounting host 10.1.1.3
Switch(config)#radius-server key test
Switch(config)#aaa enable
Switch(config)#aaa-accounting enable
```

7.8.3.2 IPv6 RADIUS Example

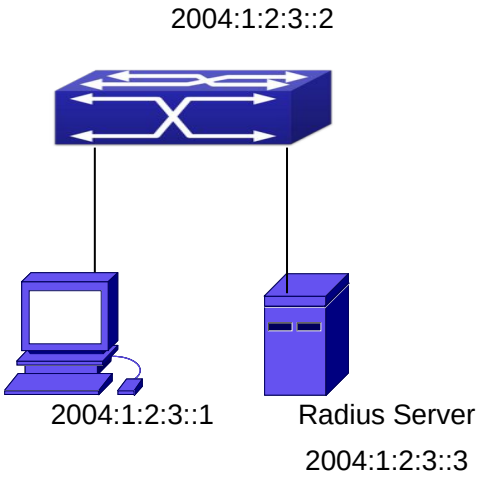


Figure 7-233 IPv6 RADIUS Configuration Topology

A computer is connected to the switch, whose IP address is 2004:1:2:3::2. The switch is connected to a RADIUS authentication server through a port other than Ethernet1/0/2. The IP address of the server is 2004:1:2:3::3; the default authentication port is 1812, and the default accounting port is 1813.

The configuration steps are as follows:

```
Switch(config)#interface vlan 1
Switch(Config-if-vlan1)#ipv6 address 2004:1:2:3::2/64
Switch(Config-if-vlan1)#exit
Switch(config)#radius-server authentication host 2004:1:2:3::3
Switch(config)#radius-server accounting host 2004:1:2:3::3
Switch(config)#radius-server key test
Switch(config)#aaa enable
Switch(config)#aaa-accounting enable
```

7.8.4 RADIUS Troubleshooting

When RADIUS is configured and used, authentication may fail because of physical connection failures or configuration errors. Ensure the following:

- First, the physical connection to the RADIUS server is in good condition.
- Second, all interfaces and link protocols are up (use the **show interface** command).
- Third, the RADIUS key configured on the switch matches the key configured on the RADIUS server.
- Finally, the switch connects to the correct RADIUS server.

If the RADIUS authentication problem persists, use **debug aaa** and other debug commands, record the debug messages for 3 minutes, and contact EnGenius technical support with the recorded messages.

7.9 SSL

7.9.1 Introduction to SSL

As networking technology spreads, network security has an increasingly important impact on the availability and usability of network applications, and it has become one of the greatest barriers to modern network applications.

To protect sensitive data transmitted over the Web, Netscape introduced the Secure Sockets Layer (SSL) protocol for its web browser. SSL 2.0 and 3.0 have been released. SSL 2.0 is obsolete because of security problems and is not supported on the switch. SSL uses public-key encryption and has become the industry standard for secure web browsing on the Internet. Web browsers combine HTTP and SSL to achieve secure communication.

SSL is a security protocol that protects private data transmitted over the Internet. It provides secure transmission between a client and a server, with server authentication and optional client authentication. SSL must run over a reliable transport layer (such as TCP) and is independent of the application layer: protocols such as HTTP, FTP, and Telnet can run transparently over SSL. Before data is transmitted, SSL negotiates the encryption algorithm, the encryption key, and server authentication. After the negotiation, all transmitted data is encrypted.

As described above, the secure channel provided by SSL has three characteristics:

- Privacy: the cipher suite is negotiated first, and then all messages are encrypted.
- Authentication: client authentication is optional, but the server is always authenticated.
- Reliability: each message includes a message integrity check (using a MAC).

7.9.1.1 Basic Elements of SSL

SSL provides a secure channel for arbitrary application data between two communicating programs. In theory, an SSL connection is similar to an encrypted TCP connection. SSL sits below the application layer and above TCP. If the underlying data transport is reliable, data written to the

network is delivered to the other program in sequence, without packet loss or retransmission. In theory, many transport protocols can provide this service, but in practice SSL almost always runs over TCP rather than directly over UDP or IP.

When the web function is running on the switch and clients access the switch web interface through a browser, SSL can be used. Communication between the client and the switch over an SSL connection improves security.

First, enable SSL on the switch. When a client accesses the switch through HTTPS, an SSL session is established between the switch and the client. After the SSL session is established, all application-layer data is encrypted.

The SSL handshake takes place while the SSL session is being established, and the switch must provide a certificate key. Currently, the keys provided by the switch are not formal certificate keys issued by an official authority but private certificate keys generated by SSL software under Linux, which may not be recognized by the web browser. For switch applications, a formal SSL certificate key is not necessary; a private certificate key is sufficient to secure communication between users and the switch. Currently, the client is not required to verify the validity of the certificate key. The encryption key and encryption method are negotiated during the handshake and then used for data encryption.

SSL session handshake process:

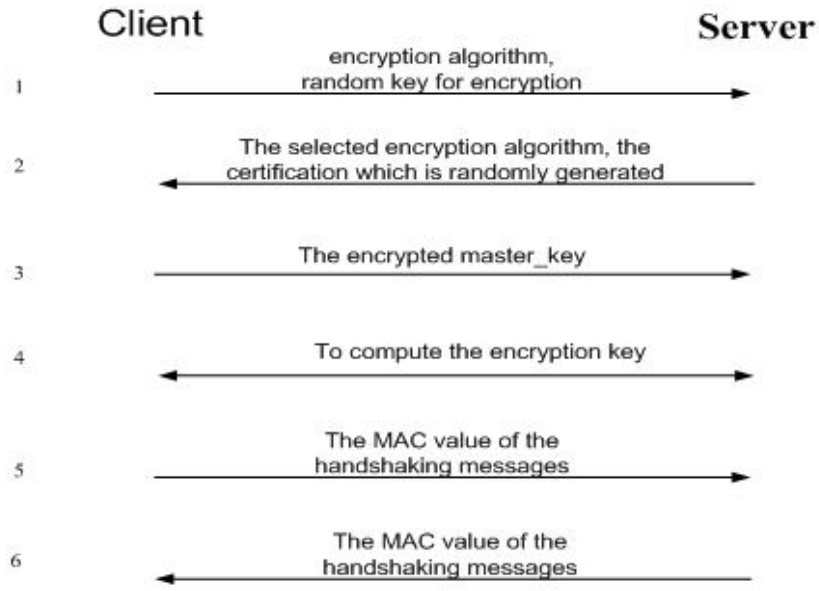


Figure 7-244 SSL Session Handshake Process

7.9.2 SSL Configuration Task List

1. Enable or disable the SSL function
2. Configure or delete the SSL port number
3. Configure or delete the SSL cipher suite
4. Maintain and diagnose the SSL function

1. Enable or disable the SSL function

Command	Explanation
Global Mode	
ip http secure-server **no ip http secure-server**	Enables or disables the SSL function.

2. Configure or delete the SSL port number

Command	Explanation
Global Mode	
ip http secure-port <*port-number*> **no ip http secure-port**	Configures the port number used by SSL. The **no ip http secure-port** command deletes the port number.

3. Configure or delete the SSL cipher suite

Command	Explanation
Global Mode	
ip http secure-ciphersuite {des-cbc3-sha|rc4-128-sha| des-cbc-sha} **no ip http secure-ciphersuite**	Configures or deletes the cipher suite used by SSL.

4. Maintain and diagnose the SSL function

Command	Explanation
Admin Mode and Configuration Mode	
show ip http secure-server status	Displays the configured SSL information.
debug ssl **no debug ssl**	Enables or disables SSL debugging.

7.9.3 SSL Typical Example

When the web function is enabled on the switch, SSL can be configured so that users access the switch web interface securely. With SSL configured, communication between the client and the switch is encrypted through SSL.

First, enable SSL on the switch. When a client accesses the switch through HTTPS, an SSL session is established between the switch and the client. After the SSL session is established, all application-layer data is encrypted.

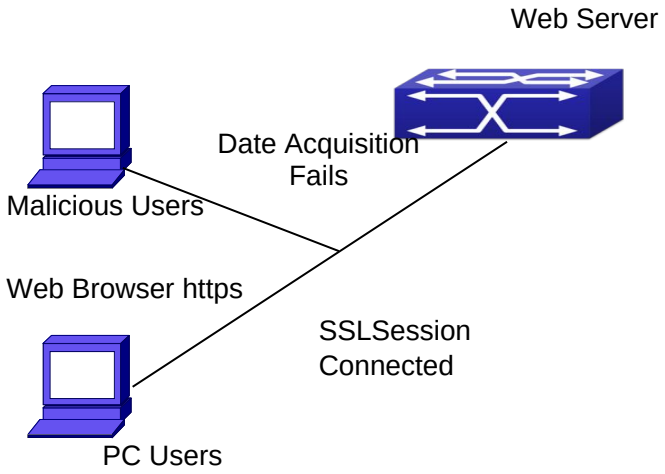


Figure 7-25

Configuration on the switch:

```
Switch(config)# ip http secure-server
Switch(config)# ip http secure-port 1025
Switch(config)# ip http secure-ciphersuite rc4-128-sha
```

7.9.4 SSL Troubleshooting

When SSL is configured and used, the SSL function may fail because of physical connection failures or configuration errors. Ensure the following:

- ☞ First, the physical connection is in good condition.
- ☞ Second, all interfaces and link protocols are up (use the show interface command).
- ☞ Then, make sure that the SSL function is enabled (use the ip http secure-server command).
- ☞ If a non-default port number is configured, use that port number when entering the web address.
- ☞ If SSL is enabled, restart SSL after changing the port or encryption configuration.
- ☞ IE 7.0 or later is required to use des-cbc-sha.
- ☞ If the SSL problem persists, use debug ssl and other debug commands, record the debug messages for 3 minutes, and contact EnGenius technical support with the recorded messages.

7.10 VLAN-ACL

7.10.1 Introduction to VLAN-ACL

ACL policies can be applied to a VLAN to control access on all ports in the VLAN, which simplifies network management. An ACL policy configured on a VLAN takes effect on all member ports of the VLAN, so it does not need to be configured on each member port.

When VLAN ACLs and port ACLs are configured at the same time, deny takes precedence: if a packet matches both a VLAN ACL and a port ACL and either rule is deny, the packet is dropped.

VLAN ACLs can filter packets in the egress and ingress directions; packets that match specific rules are permitted or denied. IP ACL, MAC ACL, MAC-IP ACL, and IPv6 ACL are supported. In the ingress direction, a VLAN can be bound to all four ACL types at the same time. In the egress direction, a VLAN has four resources: an IP ACL or a MAC ACL uses one resource each, and a MAC-IP ACL or an IPv6 ACL uses two resources each, so the egress direction of a VLAN cannot be bound to all four ACL types at the same time. When three ACL types are bound at the same time, they must be IP, MAC, and MAC-IP, or IP, MAC, and IPv6. When two ACL types are bound at the same time, any combination is valid. Only one ACL of each type can be applied to a VLAN.

7.10.2 VLAN-ACL Configuration Task List

1. Configure an IP VLAN ACL
2. Configure a MAC VLAN ACL
3. Configure a MAC-IP VLAN ACL
4. Configure an IPv6 VLAN ACL
5. Display VLAN ACL configuration and statistics
6. Clear VLAN ACL statistics

1. Configure an IP VLAN ACL

Command	Explanation
Global Mode	
vacl ip access-group {<1-299> | WORD} {in | out} [traffic-statistic] vlan WORD **no vacl ip access-group {<1-299> | WORD} {in | out} vlan WORD**	Configures or deletes an IP VLAN ACL.

2. Configure a MAC VLAN ACL

Command	Explanation
Global Mode	

vacl mac access-group {<700-1199> | WORD} {in | out} [traffic-statistic] vlan WORD **no vacl mac access-group {<700-1199> | WORD} {in | out} vlan WORD**	Configures or deletes a MAC VLAN ACL.

3. Configure a MAC-IP VLAN ACL

Command	Explanation
Global Mode	
vacl mac-ip access-group {<3100-3299> | WORD} {in | out} [traffic-statistic] vlan WORD **no vacl mac-ip access-group {<3100-3299> | WORD} {in | out} vlan WORD**	Configures or deletes a MAC-IP VLAN ACL.

4. Configure an IPv6 VLAN ACL

Command	Explanation
Global Mode	
vacl ipv6 access-group (<500-699> | WORD) {in | out} (traffic-statistic|) vlan WORD **no ipv6 access-group {<500-699> | WORD} {in | out} vlan WORD**	Configures or deletes an IPv6 VLAN ACL.

5. Display VLAN ACL configuration and statistics

Command	Explanation
Admin Mode	
show vacl [in | out] vlan [<*vlan-id*>]	Displays the VACL configuration and statistics.

6. Clear VLAN ACL statistics

Command	Explanation
Admin Mode	
clear vacl [in | out] statistic vlan [<*vlan-id*>]	Clears the VACL statistics.

7.10.3 VLAN-ACL Configuration Example

A company's departments are divided into VLANs: the technical department is VLAN 1, and the finance department is VLAN 2. The technical department may access the outside network only outside working hours, and for security, the finance department is never allowed to access the outside network. The following policies are configured:

- ☞ Policy VACL_A for the technical department: outside working hours, access to the outside network is permitted; at other times, it is denied. The policy is applied to VLAN 1.
- ☞ Policy VACL_B for the finance department: access to the outside network is denied at all times, but access to the internal network is unrestricted. The policy is applied to VLAN 2.

The network environment is shown below:

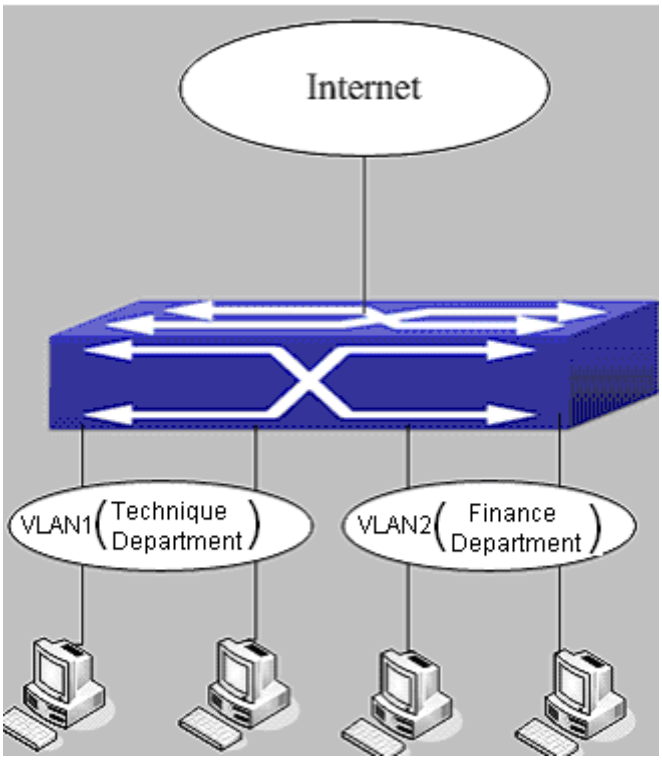


Figure 7-256 VLAN-ACL Configuration Example

Configuration example:

1) First, configure a time range whose valid period is the working hours of weekdays:

```
Switch(config)#time-range t1
Switch(config-time-range-t1)#periodic weekdays 9:00:00 to 12:00:00
Switch(config-time-range-t1)#periodic weekdays 13:00:00 to 18:00:00
```

2) Configure the extended IP ACL vacl_a, which during working hours allows access only to resources within the internal network (such as 192.168.0.255).

```
Switch(config)# ip access-list extended vacl_a
Switch(config-ip-ext-nacl-vacl_a)# permit ip any-source 192.168.0.0 0.0.0.255
time-range t1
Switch(config-ip-ext-nacl-vacl_a)# deny ip any-source any-destination time-
range t1
```

3) Configure the extended IP ACL vacl_b, which at all times allows access only to resources within the internal network (such as 192.168.1.255).

```
Switch(config)#ip access-list extended vacl_b
Switch(config-ip-ext-nacl-vacl_a)# permit ip any-source 192.168.1.0 0.0.0.255
Switch(config-ip-ext-nacl-vacl_a)# deny ip any-source any-destination
```

4) Apply the ACLs to the VLANs.

```
Switch(config)#vacl ip access-group vacl_a in vlan 1
Switch(config)#vacl ip access-group vacl_b in vlan 2
```

7.10.4 VLAN-ACL Troubleshooting

- ☞ When VLAN ACLs and port ACLs are configured at the same time, deny takes precedence: if a packet matches both a VLAN ACL and a port ACL and either rule is deny, the packet is dropped.
- ☞ Only one ACL of each type can be applied to a VLAN; for example, each VLAN can have only one basic IP ACL.

7.11 Captive Portal Authentication

7.11.1 Captive Portal Authentication Configuration

7.11.1.1 Introduction to Captive Portal Authentication

Authentication is a way to manage and control user access to network resources. The authentication server stores client authentication information according to certain principles. When a user needs to use network resources, the captive portal function redirects the user's network request to the authentication server, and the user must provide a valid username, password, and other information. The authentication server checks this information and decides whether the user is allowed to use the network resources. In the authentication function, the switch relays communication between the user and the authentication server: through the switch configuration, the user can connect to and communicate with the authentication server, and the server analyzes the data and returns the corresponding response to the user. The authentication function is based on the redirection function.

Redirection forwards an original request to a predetermined site and continues the operation there. When the AP receives a client request, it forwards the request to a predetermined address, where the client interacts with the redirected address to complete certain functions and operations. This allows users to be managed and monitored. The client is redirected to the portal authentication page and must enter a username and password; only after the username and password are authenticated can the client use network resources. Portal authentication can apply different control policies to different types of users.

The portal server function allows a different external portal server to be configured for each CP configuration. When the network is bound to CP configurations with different portal servers, each redirect page is launched through the portal server bound to its configuration. Up to 10 external portal servers can be configured, and each CP configuration can be bound to one portal server.

7.11.1.2 Captive Portal Authentication Configuration

The captive portal authentication configuration task list is as follows:

1. Enable or disable the captive portal authentication function

2. Configure the captive portal redirect function

1) Configure or delete the portal server name

2) Configure the user instance

3) Configure the redirect address

4) Configure the redirect URL head

5) Configure or delete the RADIUS server name

6) Bind or unbind the portal server

7) Configure the URL to carry the ac-name parameter

8) Configure the URL to carry the SSID parameter

9) Configure the URL to carry the nas-ip parameter

3. Configure the AAA function

1) Enable or disable the AAA function

2) Configure the RADIUS authentication server group name

4. Configure the RADIUS authentication server

1) Configure the RADIUS server key

2) Configure the RADIUS authentication server address

5. Bind the portal rule to the port

1. Enable or disable the captive portal authentication function

Command	Explanation
Captive Portal Mode	
enable disable	Enables or disables the captive portal function globally.

2. Configure the captive portal redirect function

Command	Explanation
Captive Portal Mode	
external portal-server server-name <*name*> {ipv4 | ipv6} <*ipaddr*> [port <*1-65535*>] **no external portal-server {ipv4 | ipv6}server-name <*name*>**	Configures or deletes an external portal server.
nas-ip4 <A.B.C.D>	Configures the NAS IP address.
configuration <*cp-id*> no configuration <*cp-id*>	Configures or deletes a portal instance for a type of user. Up to 10 instances can be configured.
Captive Portal Instance Mode	
redirect url-head <word> **no redirect url-head**	Configures the redirect URL head, including the transfer protocol, host name, port, and path. The no command deletes the configuration.
radius-auth-server <*server-name*> no radius-auth-server	Configures or deletes the authentication server name.
portal-server {ipv4 | ipv6} <*name*> **no portal-server {ipv4 | ipv6}**	Binds or unbinds the portal server.
enable disable	Enables or disables a portal instance.
ac-name <word> **no ac-name**	Configures the URL to carry the ac-name parameter. The no command deletes it.
redirect attribute ssid name <word> **no redirect attribute ssid name**	Configures the URL to carry the SSID parameter. The no command restores the default SSID value.
redirect attribute nas-ip enable **no redirect attribute nas-ip enable**	Configures the URL to carry the nas-ip parameter. The no command disables this function.
redirect attribute nas-ip name <word> **no redirect attribute nas-ip name**	Configures the name of the nas-ip parameter carried in the URL. The no command restores the default name.

3. Configure the AAA function

Command	Explanation
Global Mode	
aaa enable no aaa enable	Enables or disables the AAA function of a captive portal instance.
aaa group server radius <*word*> no aaa group server radius <*word*>	Configures or deletes the RADIUS server group name of the AAA function.

4. Configure the RADIUS authentication server

Command	Explanation
Global Mode	

radius-server key <*word*> no radius-server key	Configures or deletes the RADIUS server key.
radius-server authentication host <*A.B.C.D*> no radius-server authentication host <*A.B.C.D*>	Configures or deletes the RADIUS authentication server address.

5. Bind the portal rule to the port

Command	Explanation
Global Mode	
vlan-pool <1-255> <WORD> **no vlan-pool <1-255>**	Configures or deletes a VLAN pool (optional).
Port Mode	
portal enable configuration <id> [vlan-pool WORD] **no portal enable [vlan-pool WORD]**	Enables portal authentication on the port and specifies the instance number bound to the port. You can also specify the VLANs on which portal authentication is enabled; if no VLAN is specified, portal authentication is enabled on all VLANs by default. The no command disables portal authentication on the port (required).

7.11.1.3 Captive Portal Authentication Examples

Figure 7-27 Authentication Function Configuration

As shown above, PC1 is the terminal client. It has an HTTP browser but no 802.1X authentication client, and it accesses the network through portal authentication.

Switch1 is the access device. Its accounting server address is configured as the IP address and port of the RADIUS server, and the accounting function is enabled. Ethernet1/0/2 connects to PC1 and has portal authentication enabled, and the redirect address is configured as the IP address and port of the portal server. Ethernet1/0/2 therefore blocks all traffic except DHCP, DNS, and ARP packets.

Switch2 is the aggregation switch. Its Ethernet1/0/2 connects to the RADIUS server, and Ethernet1/0/3 connects to the portal server. The address of the RADIUS server is 192.168.40.100, and the address of the portal server is 192.168.40.99. Ethernet1/0/4 connects to the DHCP server, and Ethernet1/0/5 connects to the DNS server. Ethernet1/0/6 is a trunk port connected to trunk port Ethernet1/0/4 of Switch1.

Configure the RADIUS server:

```
switch (config)#radius-server key 0 test
switch (config)#aaa group server radius radius_aaa_1
switch (config-sg-radius)# server 192.168.40.100
```

Configure global authentication:

```
Switch(config)#interface vlan 1
Switch(config-if-vlan1)#ip address 192.168.40.50 255.255.255.0
Switch(config)#free-resource 1 destination ipv4 192.168.40.99/32
Switch(config)# vlan-pool 1 2-10
```

Configure the portal function and portal server in the portal instance:

```
Switch (config)#captive-portal
Switch (config-cp)#enable
Switch(config-cp)# nas-ipv4 192.168.40.50
Switch(config-cp)# external portal-server server-name abc ipv4 192.168.40.99
Switch (config-cp)# configuration 1
Switch (config-cp-instance)#name helix4
```

```
Switch (config-cp-instance)#radius-auth-server abc99
Switch (config-cp-instance)# redirect attribute nas-ip enable
Switch (config-cp-instance)#redirect attribute nas-ip name kk
Switch (config-cp-instance)#ac-name helix4
Switch (config-cp-instance)#redirect url-head http://192.168.40.99/a70.htm
Switch (config-cp-instance) # portal-server ipv4 abc
```

Enable the portal function on the port:

```
Switch (config)# interface ethernet1/0/2
Switch (config-if-ethernet1/0/2)#portal enable configuration 1 vlan-pool 1
```

7.11.1.4 Captive Portal Authentication Troubleshooting

If problems occur when using the redirection function, check for the following causes:

- Whether the captive portal function and the portal configuration are both enabled. If either is disabled, the captive portal function does not work and the client cannot be redirected to the specified page.
- Whether the authentication server name in the AAA module matches the authentication server name configured for the captive portal.
- Whether portal authentication is enabled on the port that connects the PC to the switch.

7.11.2 Accounting Function Configuration

7.11.2.1 Introduction to Accounting Function

The accounting function monitors and accounts for users of network resources. A client cannot access network resources until it passes captive portal authentication. The session duration of a user can be defined to control the user's network usage time and traffic.

7.11.2.2 Accounting Function Configuration

The accounting function configuration task list is as follows:

1. Configure the RADIUS accounting server

1) Configure or delete the accounting server address

2. Configure the AAA accounting function

1) Enable or disable the accounting service

3. Configure the captive portal accounting function

1) Block or unblock the portal function

2) Configure or delete the captive portal configuration name

3) Enable or disable captive portal accounting

4) Configure or delete the captive portal accounting server name

5) Configure or delete the captive portal session timeout

1. Configure the RADIUS accounting server

Command	Explanation
Global Mode	
radius-server accounting host <*A.B.C.D*> no radius-server accounting host <*A.B.C.D*>	Configures or deletes the accounting server address.

2. Configure the AAA accounting function

Command	Explanation
Global Mode	

aaa-accounting enable no aaa-accounting	Enables or disables the accounting service.

3. Configure the captive portal accounting function

Command	Explanation
Captive Portal Instance Mode	
name <*word*> no name	Configures or deletes the captive portal configuration name.
radius accounting no aaa-accounting	Enables or disables captive portal accounting.
radius-acct-server <*word*> no radius-acct-server	Configures or deletes the captive portal accounting server name.
session-timeout <*0-86400*> session-timeout	Configures or deletes the captive portal session timeout.

7.11.2.3 Accounting Function Examples

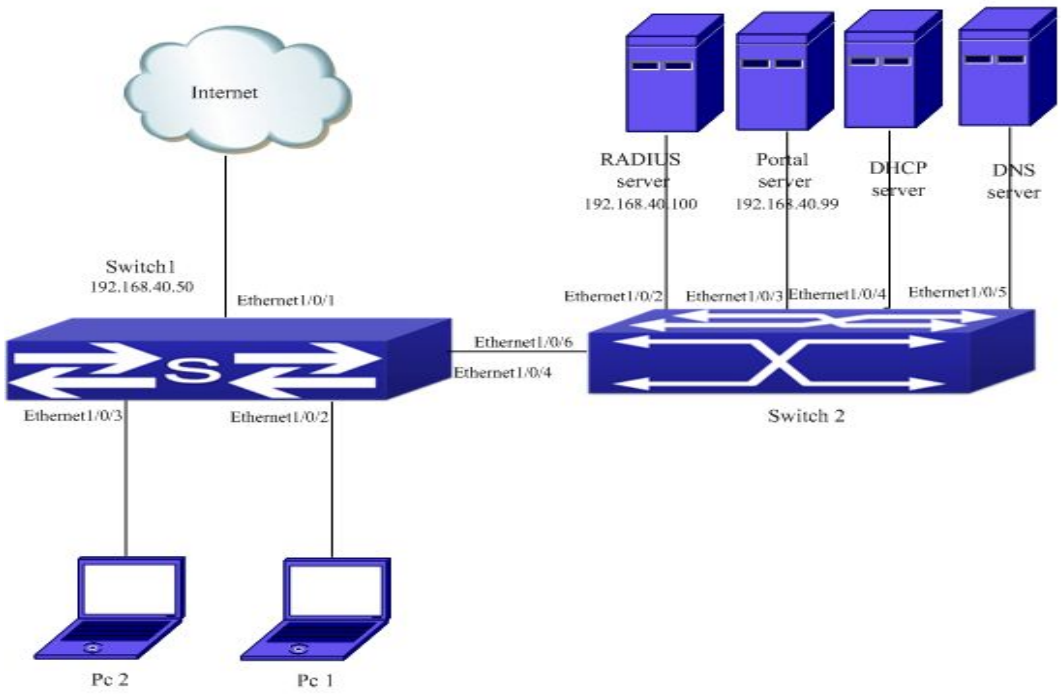


Figure 7-28 Accounting Function Configuration

1. Configure the AAA accounting function on Switch1.

AAA configuration on Switch1:

```
switch 1(config)# aaa enable
switch 1(config)# aaa-accounting enable
switch 1(config)# radius-server accounting host 192.168.40.100
switch1 (config)#radius-server key 0 test
switch1 (config)#aaa group server radius abc99
switch (config-sg-radius)# server 192.168.40.100
```

2. Configure the captive portal accounting function on Switch1.

```
Switch 1(config)#captive-portal
Switch 1(config-cp)#enable
Switch1 (config-cp)# configuration 1
Switch1 (config-cp-instance)#radius accounting
Switch1 (config-cp-instance)# radius-acct-server abc99
```

7.11.2.4 Accounting Function Troubleshooting

If problems occur when using the accounting function, check for the following causes:

- Whether the captive portal function and the portal configuration are both enabled. If either is disabled, the captive portal function does not work and the client cannot be redirected to the specified page.
- Whether the authentication server name in the AAA module matches the authentication server name configured for the captive portal.

7.11.3 Free-Resource Configuration

7.11.3.1 Introduction to Free-Resource

Free resources allow the captive portal module to grant access to specified resources without authentication. By configuring free-resource rules, clients can access specific network resources directly without portal authentication.

7.11.3.2 Free-Resource Configuration

1. Configure a free-resource rule

Command	Explanation
Global Mode	
free-resource destination {ipv4 A.B.C.D/M | ipv6 X:X:X:X/M} **[no] free-resource destination {ipv4 A.B.C.D/M |ipv6 X:X:X:X/M}**	Configures or deletes a free-resource rule.

7.11.3.3 Free-Resource Examples

Case:

Set up the environment shown below. IP is the address segment of Client1, and Destination IP is the address segment of the resources that clients want to access. RADIUS server 1 is specified as the authentication server. Client1 and Client2 can access the free resources in 3.1.1.0/24 without being redirected to the authentication server.

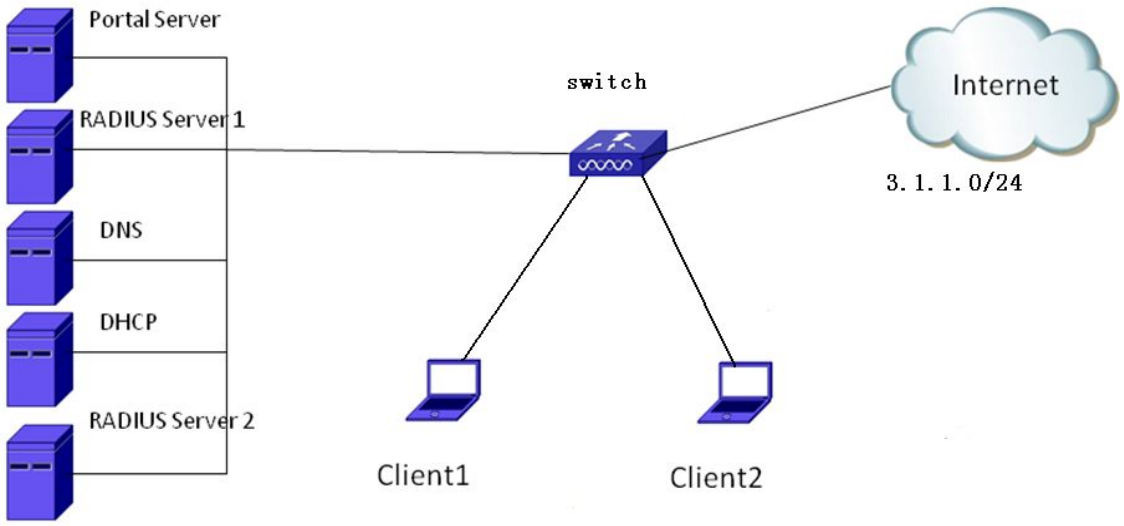


Figure 7-29 Multi-Portal Server Function Configuration

Configuration steps:

```
Switch1(config-)# free-resource destination ipv4 3.1.1.0/24
```

7.11.3.4 Free-Resource Troubleshooting

If problems occur when using the redirection function, check for the following causes:

- Whether the captive portal function and the portal configuration are both enabled. If either is disabled, the captive portal function does not work and the client cannot be redirected to the specified page.
- Whether a portal rule is enabled on the port that connects the client to the switch.

7.11.4 Authentication White-List Configuration

7.11.4.1 Introduction to Authentication White-List

The authentication white list is used for special users in the network. The administrator can allow these users to connect to the network and use network resources without authentication; to do so, the administrator needs the MAC address of each such user. Users with this permission are also not billed, so they are considered advanced users.

7.11.4.2 Authentication White-List Configuration

1. Configure the user MAC addresses allowed by the authentication white list

Command	Explanation
Global Mode	
free mac < MACADD><MACMASK> **no free mac < MACADD><MACMASK>**	Configures or deletes a MAC address that does not need authentication.

7.11.4.3 Authentication White-List Examples

Case:

As shown below, Client1 and Client2 are terminal clients, and portal authentication is enabled on the switch port they connect to. Because these two clients are advanced users, they can access the network without authentication.

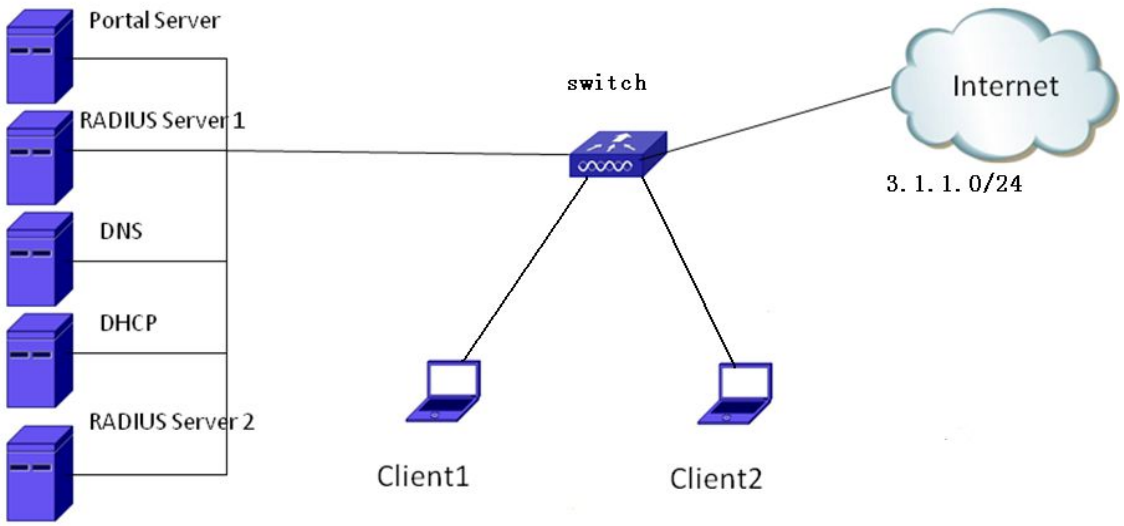


Figure 7-260 Authentication White-List Function Configuration

Configuration steps:

Configure the authentication white list for PC1 and PC2.

```
Switch1 (config)#free-mac 68-74-7f-29-76-04 ff-ff-ff-ff-ff-ff
Switch1 (config)#free-mac 00-03-0f-11-11-11 ff-ff-ff-ff-ff-ff
```

7.11.4.4 Authentication White-List Troubleshooting

If problems occur when using the authentication white-list function, check for the following causes:

- Whether the free MAC address matches the client MAC address.
- Whether a portal rule is enabled on the port that connects the client to the switch.

7.11.5 http-redirect-filter

7.11.5.1 Introduction to http-redirect-filter

http-redirect-filter specifies the IP addresses or domain names used for HTTP redirection in portal authentication. Only HTTP packets destined for a configured IP address or domain name are redirected.

This function can be used with Captive Portal to filter user access after captive portal is enabled.

7.11.5.2 http-redirect-filter Configuration

http-redirect-filter configuration task list:

1. Configure the http-redirect-filter rule
2. Bind the http-redirect-filter rule to a CP instance

1. Configure the http-redirect-filter rule

Command	Explanation
Captive Portal Mode	
http-redirect-filter <1-32> (ip A.B.C.D| domain WORD) **no http-redirect-filter (<1-32>|all)**	Configures an http-redirect-filter rule. The no command deletes it.

2. Bind the http-redirect-filter rule to a CP instance

Command	Explanation
Captive Portal Mode	
http-redirect-filter <1-32> **no http-redirect-filter (<1-32>|all)**	Binds a rule to a captive portal instance. The no command deletes the binding.

7.11.5.3 http-redirect-filter Examples

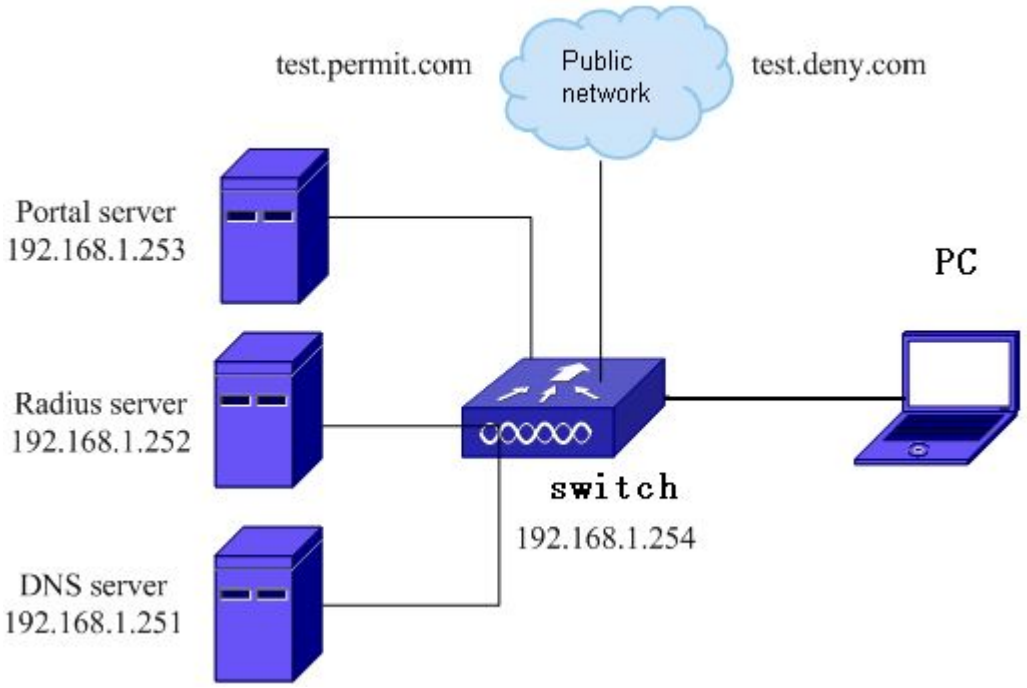


Figure 7-282 http-redirect-filter Function Case

As shown above, if the client needs to access the public network site "test.permit.com" before portal authentication, configure a URL white list. If the client must be prevented from accessing the public network site "test.deny.com" after portal authentication, configure a URL black list.

Configure as follows:

1. Configure the authentication key, authentication server, accounting server, and AAA mode of the RADIUS server in Global Mode:

```
switch (config)#radius-server key 0 test
switch (config)#radius-server authentication host 192.168.1.252
switch (config)#radius-server accounting host 192.168.1.252
switch (config)#aaa-accounting enable
switch (config)#aaa enable
switch (config)#aaa group server radius radius_aaa_1
switch config-sg-radius)# server 192.168.1.252
switch(config)#interface vlan 192
switch(config-if-vlan1)#ip address 192.16.1.254 255.255.255.0
switch(config)#free-resource 1 destination ipv4 192.168.1.252/32
```

2. Configure the portal function and portal server in the portal instance:

```
Switch (config)#captive-portal
Switch (config-cp)#enable
Switch(config-cp)# nas-ipv4 192.168.1.254
Switch(config-cp)# external portal-server server-name abc ipv4 192.168.1.253
Switch (config-cp)# configuration 1
Switch (config-cp-instance)#enable
Switch (config-cp-instance)#name helix4
Switch (config-cp-instance)#radius accounting
Switch (config-cp-instance)#radius-acct-server abc99
Switch (config-cp-instance)#radius-auth-server abc99
Switch (config-cp-instance)#redirect attribute nas-ip enable
Switch (config-cp-instance)#ac-name helix4
Switch (config-cp-instance)#redirect url-head http://192.168.1.253/a70.htm
Switch (config-cp-instance)#portal-server ipv4 abc
```

3. Configure the http-redirect-filter rule:

```
Switch (config)#captive-portal
Switch (config-cp)# http-redirect-filter 1 domain test.permit.com
Switch (config-cp)#configuration 1
Switch (config-cp-instance)# http-redirect-filter 1
```

4. Enable the portal function on the port:

```
Switch (config)# interface ethernet1/0/2
Switch (config-if-ethernet1/0/2)#portal enable configuration 1
```

Before authentication, the client is redirected for authentication only when it accesses "test.permit.com"; accessing any other address does not trigger redirection.

7.11.5.4 http-redirect-filter Troubleshooting

If problems occur when using the http-redirect-filter function, check the following:

- Whether the configured rule matches the domain name being accessed.
- Whether the DNS server is configured correctly and can resolve the configured domain name.
- Whether the http-redirect-filter command is configured under the captive portal configuration.

7.11.6 Portal Non-Perception

7.11.6.1 Introduction to Portal Non-Perception

MAC authentication provides an "authenticate once, use many times" user experience. When fast MAC authentication is enabled, after a user authenticates successfully on the Portal page the first time, the user can subsequently use any application.

To provide fast MAC authentication for a large number of users, an external server must store the MAC binding information, which is added dynamically rather than manually. This solution is called the fast MAC authentication scheme. Because users do not need to enter a username and password manually when they access the network again, it is also known as the Portal non-perception authentication scheme.

7.11.6.2 Portal Non-Perception Configuration

1. Enable or disable fast MAC authentication

Command	Explanation
Captive Portal Mode	
fast-mac-auth **no fast-mac-auth**	Enables or disables fast MAC authentication.

7.11.6.3 Portal Non-Perception Examples

The environment, shown in the following figure, includes the following parts:

1. PC: the user accesses the network through the switch.

2. Public network: this part can be empty or consist of other switch devices.

3. Servers, including:

MAC binding server: stores the MAC addresses of authenticated terminals;

RADIUS server: used for portal authentication and accounting;

Portal server: used for portal authentication.

The MAC binding server, RADIUS server, and portal server can be the same device. The MAC binding server is an extension of the RADIUS server.

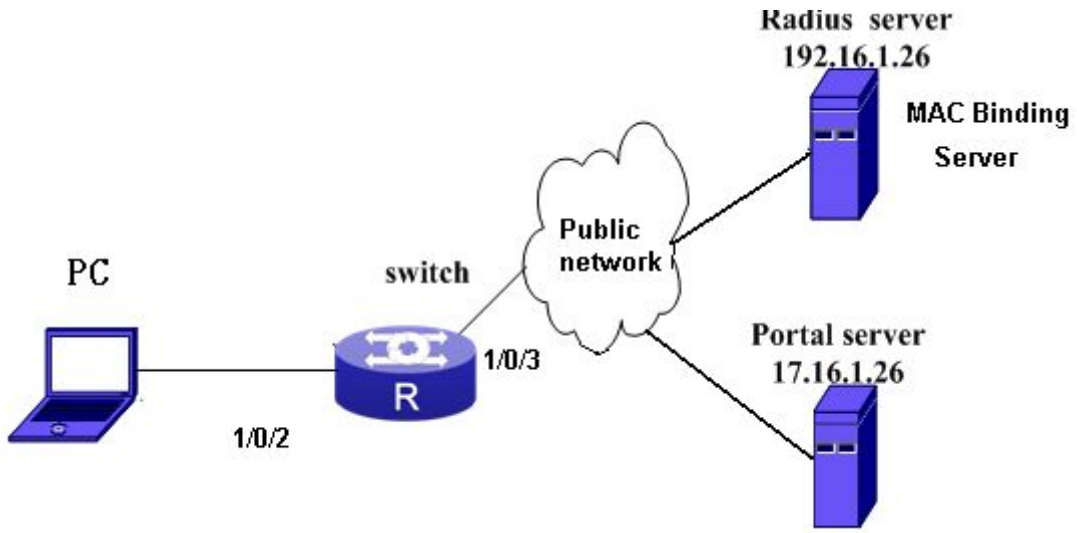


Figure 7-193

Configure as follows:

1. Configure the authentication key, authentication server, accounting server, and AAA mode of the RADIUS server in Global Mode:

```
switch (config)#radius-server key 0 test
switch (config)#radius-server authentication host 192.16.1.26
switch (config)#radius-server accounting host 192.16.1.26
switch (config)#aaa-accounting enable
switch (config)#aaa enable
switch (config)#aaa group server radius radius_aaa_1
switch config-sg-radius)# server 192.16.1.26
switch(config)#interface vlan 192
switch(config-if-vlan1)#ip address 192.16.1.50 255.255.255.0
switch(config)#free-resource 1 destination ipv4 192.16.1.26/32
```

2. Configure the portal function and portal server in the portal instance:

```
Switch (config)#captive-portal
Switch (config-cp)#enable
Switch(config-cp)# nas-ipv4 192.168.1.50
Switch(config-cp)# external portal-server server-name abc ipv4 172.16.1.26
Switch (config-cp)# configuration 1
Switch (config-cp-instance)#enable
Switch (config-cp-instance)#name helix4
Switch (config-cp-instance)#radius accounting
Switch (config-cp-instance)#radius-acct-server abc99
Switch (config-cp-instance)#radius-auth-server abc99
Switch (config-cp-instance)#redirect attribute nas-ip enable
Switch (config-cp-instance)#ac-name helix4
Switch (config-cp-instance)#redirect url-head http://172.16.1.26/a70.htm
Switch (config-cp-instance) # portal-server ipv4 abc
```

3. Configure portal non-perception:

```
Switch (config-cp)#enable
Switch (config-cp)#fast-mac-auth
```

4. Enable the portal function on the port:

```
Switch (config)# interface ethernet1/0/2
Switch (config-if-ethernet1/0/2)#portal enable configuration 1
```

Normal portal authentication is required for the first access. After that, the user can use portal non-perception authentication.

7.11.6.4 Portal Non-Perception Troubleshooting

If problems occur when using portal non-perception, check for the following causes:

- Whether the captive portal function is enabled.
- Whether fast MAC authentication is enabled.
- If fast MAC authentication does not take effect after it is configured, whether the app table entries have been delivered to the switch.

7.11.7 Portal Escaping

Current portal deployments carry a risk: when communication between the access device and the portal server is broken, new users cannot go online and online users cannot go offline, and the information on the access device and the portal server becomes inconsistent, which causes accounting errors. These problems inconvenience both operators and users.

The portal escaping function solves these problems. When the portal server or RADIUS server is not working normally, online users can continue to use the network normally and new users can still access the network. Portal escaping includes portal server escaping and RADIUS server escaping.

7.11.7.1 Portal Server Escaping

7.11.7.1.1 Introduction to Portal Server Escaping

Portal server escaping works as follows: the switch probes the portal server periodically. If a probe succeeds, the server status is set to UP. If probing fails N times (N is configurable), the server status changes to unreachable (DOWN, the escaping state); the switch then removes the network authentication restriction, allows portal users to access the network without authentication, and sends trap and log messages about the status change. When probing finds the server reachable again, the switch restores the server status to UP (the authentication state), reapplies the network authentication restriction, rejects unauthenticated users, and sends log and trap messages about the status recovery.

The switch probes the portal server status through TCP connections: it periodically initiates a TCP connection to the portal server port (2000 by default; configurable). If the connection succeeds, the portal server is running, the probe is considered successful, and the server is reachable (status UP); if the connection fails, the probe is considered failed.

Probing interval and maximum number of probing failures: the probing interval is configurable by command. The maximum number of probing failures is the number of failures allowed before the server is considered unreachable. A single probing failure does not mean that the server is unreachable; failures are counted until the count reaches the configured maximum, at which point the server is considered unreachable. After a successful probe, the count is reset to 0. Both the probing interval and the maximum number of probing failures can be configured by command.

When the server status changes from reachable to unreachable, the following three actions can be triggered, as selected by the administrator:

- Send trap: sends a trap to the network management server. The trap records the portal server name and the server status before and after the change.
- Send log: sends a log message to the log server. The log records the portal server name and the server status before and after the change.
- permit-all: also called portal escaping. It temporarily cancels portal authentication and allows all portal users to access the network while the portal server is unreachable (DOWN).

When the server status changes from unreachable to reachable, the following three actions are triggered. "Send trap" and "Send log" are configurable; "Disable portal escaping" is always performed:

- Send trap: sends a trap to the network management server. The trap records the portal server name and the server status before and after the change.
- Send log: sends a log message to the log server. The log records the portal server name and the server status before and after the change.
- Disable portal escaping: when the portal server status changes to reachable (UP), portal authentication on the VAP is restored, and new users must pass portal authentication to access the network.

Note: Portal escaping only ensures that new and existing users can continue to access the network. Other methods handle the case in which users cannot go offline normally. For example, when the portal server recovers to UP, the access device forces users offline to ensure that they go offline normally.

7.11.7.1.2 Portal Escaping Configuration

Portal escaping function configuration task list:

1. Enable the Portal escaping function and configure the probing interval and maximum number of failures.

2. Display the current connection status of the portal server.

1. Enable the Portal escaping function and configure the probing interval and maximum number of failures

Command	Explanation
Captive Portal Mode	
portal-server-detect server-name <name> [interval <interval>] [retry <retries>][action {log | permit-all | trap }] **no portal-server-detect server-name <name>**	Enables portal server escaping and configures the related optional parameters and the actions taken when the server status changes.

2．Show the current connection status of the Portal server

Command	Explanation
Admin Mode	
show captive-portal ext-portal-server server-name <name> status	Displays the portal server status, including the server address and whether portal escaping is enabled.

7.11.7.1.3 Portal Escaping Examples

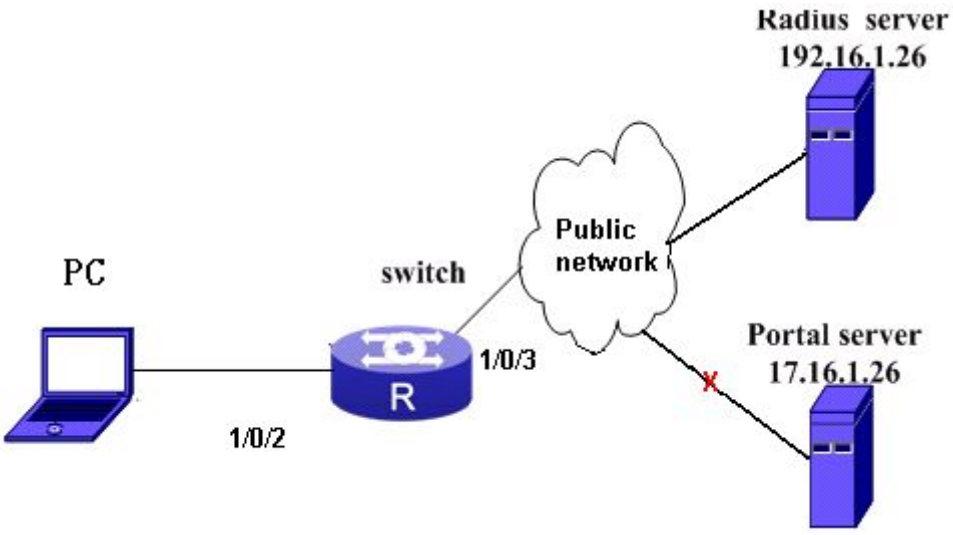


Figure 7-304 Portal Server Escaping Function Case

As shown above, when the portal server works normally, clients go online through normal portal authentication. When the portal server goes down or its connection to the switch breaks, clients cannot authenticate and go online if portal escaping is not enabled on the switch. If portal escaping is enabled, the switch detects that the portal server is unavailable and starts portal escaping, and clients can access the network without authentication. Clients that passed authentication before the portal server failed are not affected and can still access the network.

The configuration is as follows:

1. Configure the authentication key, authentication server, accounting server, and AAA mode of the RADIUS server in Global Mode.

```
switch (config)#radius-server key 0 test
switch (config)#radius-server authentication host 192.16.1.26
switch (config)#radius-server accounting host 192.16.1.26
switch (config)#aaa-accounting enable
switch (config)#aaa enable
switch (config)#aaa group server radius radius_aaa_1
switch config-sg-radius)# server 192.16.1.26
switch(config)#interface vlan 192
switch(config-if-vlan1)#ip address 192.16.1.50 255.255.255.0
switch(config)#free-resource 1 destination ipv4 192.16.1.26/32
```

2. Configure the portal function and portal server in the portal instance:

```
Switch (config)#captive-portal
Switch (config-cp)#enable
Switch(config-cp)# nas-ipv4 192.168.1.50
Switch(config-cp)# external portal-server server-name abc ipv4 172.16.1.26
Switch (config-cp)# configuration 1
Switch (config-cp-instance)#enable
Switch (config-cp-instance)#name helix4
Switch (config-cp-instance)#radius accounting
Switch (config-cp-instance)#radius-acct-server abc99
Switch (config-cp-instance)#radius-auth-server abc99
Switch (config-cp-instance)#redirect attribute nas-ip enable
Switch (config-cp-instance)#ac-name helix4
Switch (config-cp-instance)#redirect url-head http://172.16.1.26/a70.htm
Switch (config-cp-instance) # portal-server ipv4 abc
```

3. Configure portal escaping:

```
Switch (config-cp)#enable
Switch (config-cp)# portal-server-detect server-name abc interval 600 retry 2
action log permit-all trap
```

4. Enable the portal function on the port:

```
Switch (config)# interface ethernet1/0/2
Switch (config-if-ethernet1/0/2)#portal enable configuration 1
```

As shown above, portal server abc is bound to the CP instance and probing is configured with an interval of 600s. If probing fails twice, the switch sends trap and log messages reporting that the server is unreachable and enables portal escaping, allowing users to access the network without authentication.

7.11.7.1.4 Portal Escaping Troubleshooting

If the portal escaping function does not take effect, use the following methods:

- Use the **show captive-portal ext-portal-server server-name <name> status** command to check whether the detect mode of the portal server is "enable". If it is not, portal server escaping is not enabled; enable it.
- If portal escaping is enabled, check whether the Detect Operational Status is down. Portal escaping takes effect only when the server status is down.

If the portal server status is down but escaping does not take effect, the device may have a problem. If the problem persists, contact EnGenius technical support.

7.11.7.2 RADIUS Server Escaping

7.11.7.2.1 Introduction to RADIUS Server Escaping

When RADIUS server escaping is enabled, escaping takes effect when none of the RADIUS servers is reachable, and traffic from portal authentication clients is allowed. When the authentication server is detected as reachable again, the allow rule is deleted.

7.11.7.2.2 RADIUS Server Escaping Configuration

RADIUS server escaping configuration task list:

1. Enable the RADIUS server escaping function

2. Configure the RADIUS server detection interval

1. Enable the radius server escaping function

Command	Explanation
Global Mode	
radius-server escape {enable | disable }	Enables or disables RADIUS server escaping.

2. Configure the detection interval of radius server

Command	Explanation
Global Mode	
radius-server escape detection-interval {default | <second>}	Configures the RADIUS server detection interval. The default is 180s.

7.11.7.2.3 RADIUS Server Escaping Examples

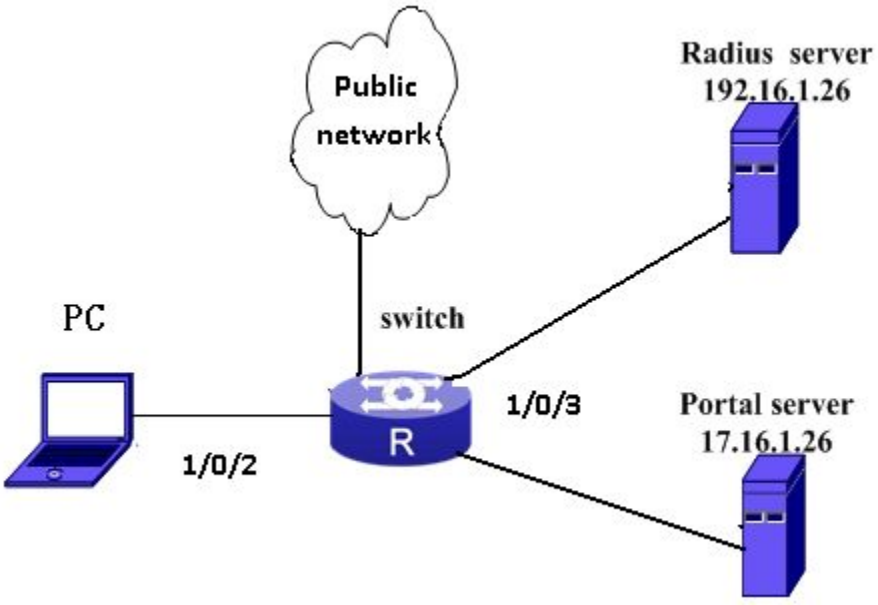


Figure 7-315 RADIUS Server Escaping Case

The configuration is as follows:

1 Configure the authentication key, authentication server, accounting server, and AAA mode of the RADIUS server in Global Mode.

```
switch (config)#radius-server key 0 test
switch (config)#radius-server authentication host 192.16.1.26
switch (config)#radius-server accounting host 192.16.1.26
```

```
switch (config)#aaa-accounting enable
switch (config)#aaa enable
switch (config)#aaa group server radius radius_aaa_1
switch config-sg-radius)# server 192.16.1.26
switch(config)#interface vlan 192
switch(config-if-vlan1)#ip address 192.16.1.50 255.255.255.0
switch(config)#free-resource 1 destination ipv4 192.16.1.26/32
```

2 Configure the portal function and portal server in the portal instance:

```
Switch (config)#captive-portal
Switch (config-cp)#enable
Switch(config-cp)# nas-ipv4 192.168.1.50
Switch(config-cp)# external portal-server server-name abc ipv4 172.16.1.26
Switch (config-cp)# configuration 1
Switch (config-cp-instance)#enable
Switch (config-cp-instance)#name helix4
Switch (config-cp-instance)#radius accounting
Switch (config-cp-instance)#radius-acct-server abc99
Switch (config-cp-instance)#radius-auth-server abc99
Switch (config-cp-instance)#redirect attribute nas-ip enable
Switch (config-cp-instance)#ac-name helix4
Switch (config-cp-instance)#redirect url-head http://172.16.1.26/a70.htm
Switch (config-cp-instance) # portal-server ipv4 abc
```

3 Configure RADIUS server escaping:

```
Switch (config)# radius-server escape enable
Switch (config)# radius-server escape detection-interval 120
```

4 Enable the portal function on the port:

```
Switch (config)# interface ethernet1/0/2
Switch (config-if-ethernet1/0/2)#portal enable configuration 1
```

The RADIUS server is probed every 120s.

7.11.7.2.4 RADIUS Server Escaping Troubleshooting

If problems occur when using RADIUS server escaping, check the following:

- Whether RADIUS server escaping is enabled.
- If it is enabled, obtain the Retransmit and Time Out values with the **show aaa config** command, and check whether the configured detection-interval is larger than (Retransmit+1)*Time Out. If the detection-interval is smaller, escaping fails.
- If the **show aaa config** command shows "Is Server live by radius escape function =0", escaping is running but not effective. If the problem persists, contact EnGenius technical support.

7.12 MAB

7.12.1 Introduction to MAB

Some devices in a network, such as printers and PDAs, cannot install an authentication client and therefore cannot perform 802.1X authentication. To access network resources, these devices can use MAB authentication instead of 802.1X authentication.

MAB authentication is a network access authentication method based on the access port and the MAC address of the MAB user. The user does not need to install any authentication client. After receiving ARP packets from a MAB user, the authentication device authenticates the MAC address of the user against the corresponding authentication information on the authentication server. When authentication succeeds, packets matching the port and source MAC address are allowed to pass. MAB users do not need to enter a username and password manually during authentication.

Currently, the MAB authentication device supports only the RADIUS authentication method. The authentication username and password can be either the MAC address of the MAB user or a fixed username and password (all users authenticate with the configured username and password).

7.12.2 MAB Configuration Task List

MAB configuration task list:

1. Enable the MAB function

1) Enable the global MAB function

2) Enable the port MAB function

2. Configure the MAB authentication username and password

3. Configure MAB parameters

1) Configure the guest VLAN

2) Configure the port binding limit

3) Configure the reauthentication period

4) Configure the offline detection period

5) Configure other parameters

1. Enable MAB function

Command	Explanation
Global Mode	
mac-authentication-bypass enable **no mac-authentication-bypass enable**	Enables or disables the global MAB authentication function.
Port Mode	
mac-authentication-bypass enable no mac-authentication-bypass enable	Enables or disables MAB authentication on the port.

2. Configure MAB authentication username and password

Command	Explanation
Global Mode	
mac-authentication-bypass username-format {mac-address | {fixed username WORD password WORD}}	Sets the username and password mode of MAB authentication.

3. Configure MAB parameters

Command	Explanation
Port Mode	
mac-authentication-bypass guest-vlan <1-4094> **no mac-authentication-bypass guest-vlan**	Sets the guest VLAN for MAB authentication. This command applies only to hybrid ports and does not take effect on access ports.
mac-authentication-bypass binding-limit <1-100> **no mac-authentication-bypass binding-limit**	Sets the maximum number of MAB bindings on the port.
Global Mode	
mac-authentication-bypass timeout reauth-period <1-3600> **no mac-authentication-bypass timeout reauth-period**	Sets the reauthentication interval after an authentication failure.

mac-authentication-bypass timeout offline-detect (0|<60-7200>) no mac-authentication-bypass timeout offline-detect	Sets the offline detection interval.
mac-authentication-bypass timeout quiet-period <1-60> no mac-authentication-bypass timeout quiet-period	Sets the quiet period of MAB authentication.
mac-authentication-bypass timeout stale-period <0-60> no mac-authentication-bypass timeout stale-period	Sets the time after which bindings are deleted when the port goes down.
mac-authentication-bypass timeout linkup-period <0-30> **no mac-authentication-bypass timeout linkup-period**	Sets the interval between port down and up when a MAB binding changes VLAN, so that the client obtains an IP address again.
	*
authentication mab {radius|local} (none|) **no authentication mab**	Configures the MAC address authentication mode and priority. The no command restores the default authentication mode.

7.12.3 MAB Example

Example:

The following is a typical MAB authentication example:

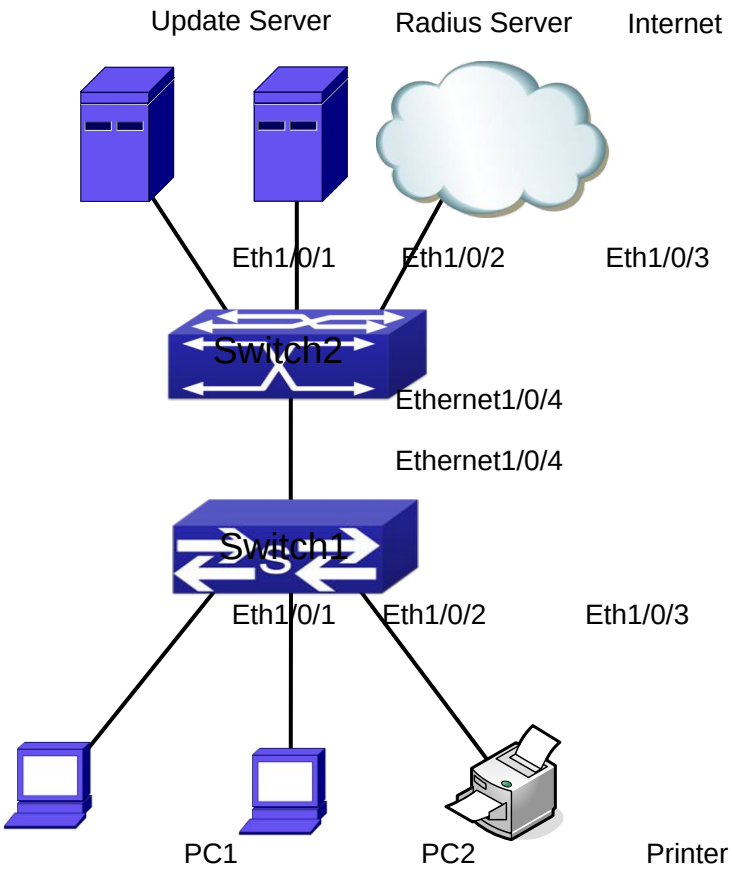


Figure 7-206 MAB application

Switch1 is a Layer 2 access switch, and Switch2 is a Layer 3 aggregation switch.

Ethernet 1/0/1 of Switch1 is an access port connected to PC1. 802.1X port-based authentication is enabled on it, and its guest VLAN is vlan8.

Ethernet 1/0/2 is a hybrid port connected to PC2. Its native VLAN is vlan1 and its guest VLAN is vlan8; it joins vlan1, vlan8, and vlan10 as untagged and has MAB enabled.

Ethernet 1/0/3 is an access port connected to the printer, with MAB enabled.

Ethernet 1/0/4 is a trunk port connected to Switch2.

Ethernet 1/0/4 of Switch2 is a trunk port connected to Switch1.

Ethernet 1/0/1 is an access port in vlan8, connected to the update server for downloading and upgrading client software.

Ethernet 1/0/2 is an access port in vlan9, connected to the RADIUS server, which is configured with auto VLAN vlan10.

Ethernet 1/0/3 is an access port in vlan10, connected to external Internet resources.

The configuration for this application is as follows:

Switch1 configuration:

(1) Enable 802.1X and MAB authentication globally, and configure the MAB authentication username and password and the RADIUS server address:

```
Switch(config)# dot1x enable
Switch(config)# mac-authentication-bypass enable
```

```
Switch(config)#mac-authentication-bypass username-format fixed username
mabuser password mabpwd
Switch(config)#vlan 8-10
Switch(config)#interface vlan 9
Switch(config-if-vlan9)ip address 192.168.61.9 255.255.255.0
Switch(config-if-vlan9)exit
Switch(config)#radius-server authentication host 192.168.61.10
Switch(config)#radius-server accounting host 192.168.61.10
Switch(config)#radius-server key test
Switch(config)#aaa enable
Switch(config)#aaa-accounting enable
```

(2) Enable authentication on each port:

```
Switch(config)#interface ethernet  1/0/1
Switch(config-if-ethernet1/0/1)#dot1x enable
Switch(config-if-ethernet1/0/1)#dot1x port-method portbased
Switch(config-if-ethernet1/0/1)#dot1x guest-vlan 8
Switch(config-if-ethernet1/0/1)#exit
Switch(config)#interface ethernet 1/0/2
Switch(config-if-ethernet1/0/2)#switchport mode hybrid
Switch(config-if-ethernet1/0/2)#switchport hybrid native vlan 1
Switch(config-if-ethernet1/0/2)#switchport hybrid allowed vlan 1;8;10 untag
Switch(config-if-ethernet1/0/2)#mac-authentication-bypass enable
Switch(config-if-ethernet1/0/2)#mac-authentication-bypass enable guest-vlan 8
Switch(config-if-ethernet1/0/2)#exit
Switch(config)#interface ethernet 1/0/3
Switch(config-if-ethernet1/0/3)#switchport mode access
Switch(config-if-ethernet1/0/3)#mac-authentication-bypass enable
Switch(config-if-ethernet1/0/3)#exit
Switch(config)#interface ethernet 1/0/4
Switch(config-if-ethernet1/0/4)# switchport mode trunk
```

7.12.4 MAB Troubleshooting

If a problem occurs when using the MAB function, check whether it is caused by the following:

- ☞ Make sure that the global and port MAB functions are enabled.
- ☞ Make sure that the correct MAB authentication username and password are used.
- ☞ Make sure that the RADIUS server configuration is correct. MAB offline detection works by checking whether the dynamic MAC address exists; if the MAC address exists in the MAC address table, the binding is not deleted. The actual offline time without traffic is 1-2 MAC aging periods plus 0-1 MAB offline-detect period.

7.13 PPPoE Intermediate Agent

7.13.1 Introduction to PPPoE Intermediate Agent

7.13.1.1 Brief Introduction to PPPoE

PPPoE (Point-to-Point Protocol over Ethernet) applies PPP to Ethernet. PPP is a link layer protocol that provides point-to-point communication and is typically used for host dial-up links, such as telephone-line dial-up. PPPoE allows many hosts on an Ethernet network to connect to a remote access concentrator through one or more bridging devices. If the remote access concentrator is a broadband access server (BAS), it can provide broadband access and accounting for these hosts, so PPPoE is commonly used for broadband access authentication on Ethernet.

7.13.1.2 Introduction to PPPoE IA

As broadband access technology develops rapidly, broadband access networks keep growing, but security has become a key concern: in current access networks, clients, access devices, and the network all face security threats, especially from clients. Traditional Ethernet users cannot be accurately identified, traced, or located. In an open yet controllable network, however, user identification and location are basic requirements. For example, for applications that require users to log in with accounts, the method provided by PPPoE Intermediate Agent can effectively prevent user accounts from being stolen.

PPPoE works in two stages: the discovery stage and the session stage. The discovery stage obtains the MAC address of the remote server to establish a point-to-point link and a session ID with the server; the session stage uses this session ID to communicate. PPPoE Intermediate Agent involves only the discovery stage, so only the discovery stage is described here.

The discovery stage has four steps:

1. Client sends PADI packet: In the first step, the client broadcasts a PADI (PPPoE Active Discovery Initiation) packet, using the broadcast address as the destination address, to discover access concentrators in the Layer 2 network. Note: this message may be sent to many access concentrators in the network.
2. Broadband access server responds with PADO packet: In the second step, the server responds with a PADO (PPPoE Active Discovery Offer) packet to the client based on the source MAC address of the received PADI packet. The packet carries the server name and service name.
3. Client sends PADR packet: In the third step, the client selects a server for the session based on the received PADO packets. The client may receive many PADO packets, because the PADI message in the first step may be sent to many servers (the server is selected according to whether the service information in the PADO packet matches the service information required by the client). After the server is selected, the MAC address of the peer for the session is known, and the client sends a PADR (PPPoE Active Discovery Request) packet to the server to request the session.
4. Server responds with PADS packet: In the fourth step, the server establishes a session ID based on the received PADR packet and sends it to the client in a PADS (PPPoE Active Discovery Session-confirmation) packet. The PPPoE discovery stage is then complete, and the session stage begins.

PADT (PPPoE Active Discovery Terminate) is a special PPPoE packet. Its Ethernet protocol number (0x8863) is the same as that of the four packets above, so it is considered a discovery stage packet. PADT can be sent at any time during a session to terminate the PPPoE session, by either the client or the server.

PPPoE Intermediate Agent provides user identification and location. During the PPPoE discovery stage, when PADI and PADR messages sent by the client pass through the network access device, the device adds its access link tag to them so that the server can accurately identify and locate the user.

If the directly connected access device is a LAN switch, the added information includes MAC, Slot ID, Port Index, VLAN ID, and so on. This function is implemented according to Migration to Ethernet-based DSL Aggregation.

7.13.1.2.1 PPPoE Intermediate Agent Exchange Process

The PPPoE Intermediate Agent exchange process is similar to the PPPoE exchange process, except that the access link tag is added to PADI and PADR packets in the first exchange. The exchange process is as follows:

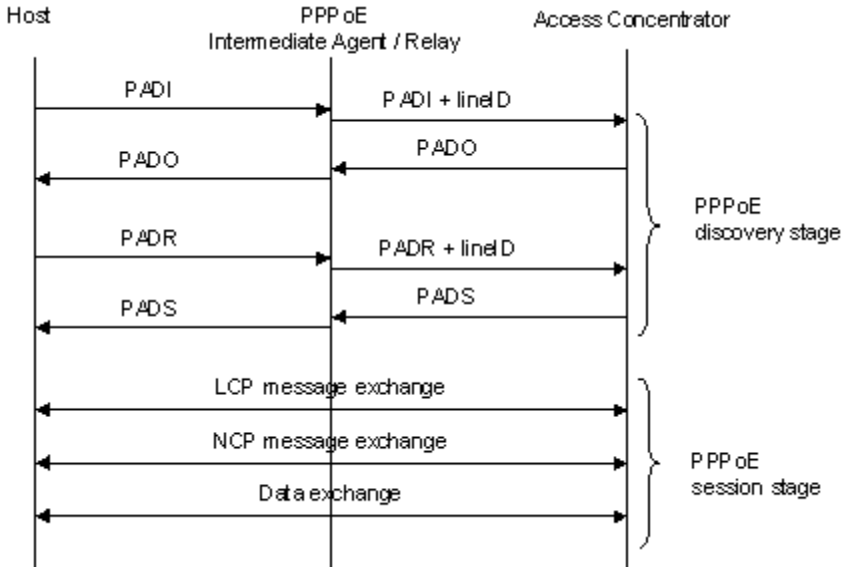


Figure 7-37 PPPoE IA Protocol Exchange Process

7.13.1.2.2 PPPoE Packet Format

The PPPoE packet format is as follows:

Ethernet II frame

Destination MAC	Source MAC	Type Field	PPPoE Data	CRC Check Sum

PPPoE data

Version	Type	Code	Session ID	Length Field	TLV1	……	TLV N

TLV frame

Type	Length	Data

The fields are described as follows:

Type field (2 bytes) of the Ethernet II frame: the protocol sets the type field value of PPPoE discovery stage packets (the 5 kinds of discovery stage packets only) to 0x8863, and of session stage packets to 0x8864.

PPPoE version field (4 bits): specifies the PPPoE protocol version. In the current version, it must be set to 0x1.

PPPoE type field (4 bits): specifies the protocol type. In the current version, it must be set to 0x1.

PPPoE code field (1 byte): specifies the packet type. 0x09 indicates a PADI packet, 0x07 a PADO packet, 0x19 a PADR packet, 0x65 a PADS packet, and 0xa7 a PADT packet.

PPPoE session ID field (2 bytes): specifies the session ID.

PPPoE length field (2 bytes): specifies the total length of all TLVs.

TLV type field (2 bytes): a TLV frame represents a tag, and the type field indicates the tag type, as listed in the following table.

TLV length field (2 bytes): specifies the length of the tag data field.

TLV data field (variable length): carries the tag data.

Tag Type	Tag Explanation
0x0000	End of a series of tags in the PPPoE data field. It is retained for version compatibility and is used by some packets.
0x0101	Service name. Indicates the services provided by the network.
0x0102	Server name. When the user receives the PADO response packet from the AC, it can obtain the server name from this tag and select the corresponding server.
0x0103	Host-unique tag. Similar to the tag field of PPPoE data packets, it is used to match the sending and receiving ends (because many PPPoE data packets

	may exist simultaneously on a broadcast network).
0x0104	AC-Cookie. Used to prevent malicious DoS attacks.
0x0105	Vendor identifier.
0x0110	Relay session ID. A PPPoE data packet can be relayed to another AC; this field is used to maintain the other connection.
0x0201	Service name error. When the requested service name is not accepted by the peer, the response packet carries this tag.
0x0202	Server name error.
0x0203	Generic error.

Table 7-1 PPPoE TAG Value Types

7.13.1.2.3 PPPoE Intermediate Agent Vendor Tag Frame

The following is the format of the tag added by PPPoE IA. Adding this tag is the primary function of PPPoE IA.

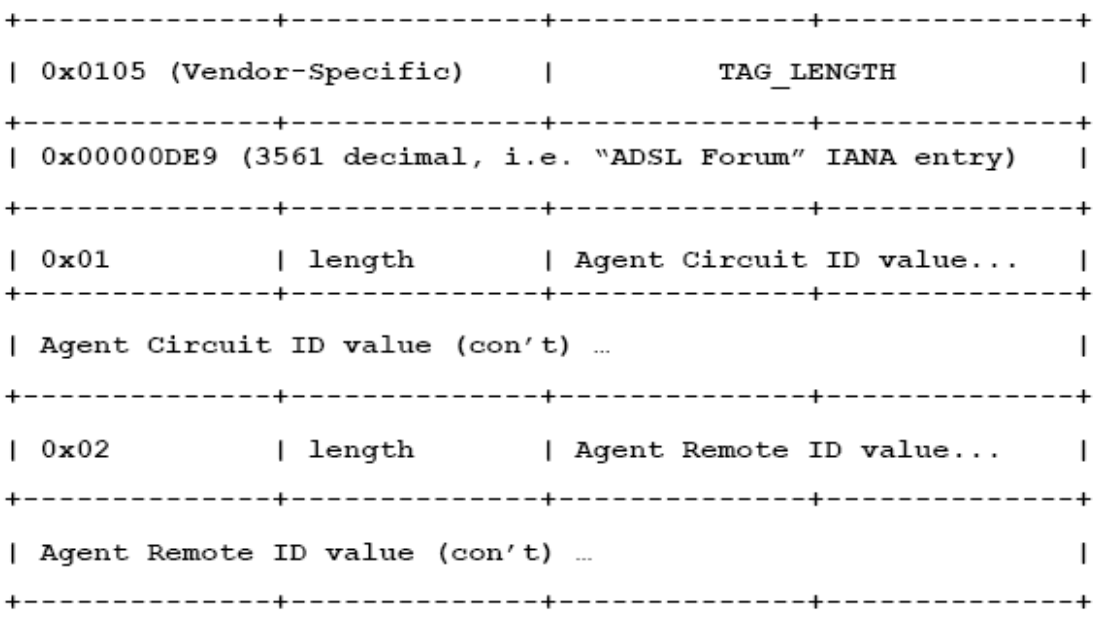


Figure 7-38 PPPoE IA Vendor Tag (4 Bytes per Row)

PPPoE IA adds a TLV tag of type 0x0105. TAG_LENGTH is the length field of the vendor tag; 0x00000DE9 is the fixed 4-byte "ADSL Forum" IANA entry; 0x01 is the type field of Agent Circuit ID, followed by its length field and Agent Circuit ID value field; 0x02 is the type field of Agent Remote ID, followed by its length field and Agent Remote ID value field.

PPPoE IA provides a default circuit ID value. The default circuit ID (shown in the following figure) contains 5 fields. ANI (Access Node Identifier) is user-configurable and must be shorter than 47 bytes; if no ANI is configured, the MAC address is used by default, occupying 6 bytes, followed by a space separator. "eth" occupies 3 bytes, followed by a space separator. "Slot ID" occupies 2 bytes, followed by a 1-byte "/" separator. "Port Index" occupies 3 bytes, followed by a 1-byte ":" separator. "Vlan ID" occupies 4 bytes. All fields are in ASCII. You can configure the circuit ID of each port as required.

ANI (n bytes)	Space (1 byte)	eth (3 bytes)	Space (1 byte)	Slot ID (2 bytes)	/ (1 byte)	Port Index (3 bytes)	: (1 byte)	Vlan ID (4 bytes)

Figure 7-39 Agent Circuit ID Value

The MAC address of the access switch is the default remote ID value of PPPoE IA. The remote ID value is user-configurable and must be shorter than 63 bytes.

7.13.1.2.4 Trust Port of PPPoE Intermediate Agent

Five kinds of packets are sent in the discovery stage: PADI and PADR packets are sent by the client to the server; PADO and PADS packets are sent by the server to the client; and PADT packets can be sent by either the server or the client.

In PPPoE IA, for security and to reduce traffic, the port connected to the server is set as a trust port, and the ports connected to clients are set as untrusted ports. A trust port can receive all packets, while an untrusted port can receive only PADI, PADR, and PADT packets sent to the server. To ensure correct client operation, the port connected to the server must be set as a trust port; each access device must have at least one trust port.

The PPPoE IA vendor tag must not appear in PPPoE packets sent by the server to the client, so any such vendor tags are stripped before the packets are forwarded. The strip function must be configured on a trust port; enabling it on an untrusted port has no effect.

7.13.2 PPPoE Intermediate Agent Configuration Task List

1. Enable PPPoE Intermediate Agent globally
2. Enable PPPoE Intermediate Agent on ports

Command	Explanation
Global Mode	
pppoe intermediate-agent **no pppoe intermediate-agent**	Enables PPPoE Intermediate Agent globally.
pppoe intermediate-agent type tr-101 circuit-id access-node-id <string> **no pppoe intermediate-agent type tr-101 circuit-id access-node-id**	Configures the access node ID field value of the circuit ID in the added vendor tag.
pppoe intermediate-agent type tr-101 circuit-id identifier-string <string> option {sp | sv | pv | spv} delimiter <WORD> [delimiter <WORD>] **no pppoe intermediate-agent type tr-101 circuit-id identifier-string option delimiter**	Configures the circuit ID in the added vendor tag.
pppoe intermediate-agent type self-defined circuit-id {vlan| port|id (switch-id (mac | hostname)| remote-mac) | string WORD} **no pppoe intermediate-agent type self-defined circuit-id**	Configures a self-defined circuit ID.
pppoe intermediate-agent type self-defined remoteid {mac | vlan-mac | hostname| string WORD} **no pppoe intermediate-agent type self-defined remote-id**	Configures a self-defined remote ID.
pppoe intermediate-agent delimiter <WORD> **no pppoe intermediate-agent delimiter**	Configures the delimiter between fields in the circuit ID and remote ID.
pppoe intermediate-agent format (circuit-id | remote-id) (hex | ascii) **no pppoe intermediate-agent format (circuit-id | remote-id)**	Configures the hex or ASCII format of the circuit ID and remote ID.
Port Mode	
pppoe intermediate-agent **no pppoe intermediate-agent**	Enables PPPoE Intermediate Agent on the port.

pppoe intermediate-agent vendor-tag strip **no pppoe intermediate-agent vendor-tag strip**	Enables the vendor tag strip function on the port.
pppoe intermediate-agent trust **no pppoe intermediate-agent trust**	Sets the port as a trust port.
pppoe intermediate-agent circuit-id <string> **no pppoe intermediate-agent circuit-id**	Sets the circuit ID of the port.
pppoe intermediate-agent remote-id <string> **no pppoe intermediate-agent remote-id**	Sets the remote ID of the port.

7.13.3 PPPoE Intermediate Agent Typical Application

A typical PPPoE Intermediate Agent application is shown below:

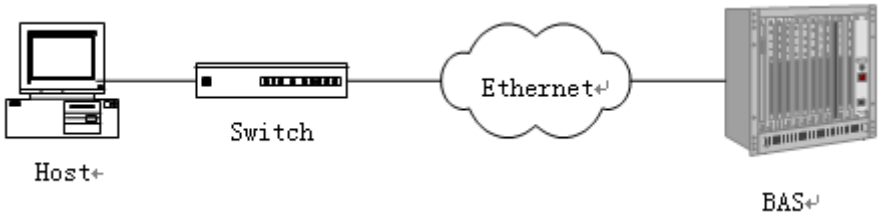


Figure 7-40 PPPoE IA Typical Application

The host and the BAS server both run PPPoE and are connected over Layer 2 Ethernet. PPPoE Intermediate Agent is enabled on the switch.

Typical configuration (1):

Step1: Enable PPPoE IA globally on the switch, whose MAC address is 0a0b0c0d0e0f.

```
Switch(config)# pppoe intermediate-agent
```

Step2: Configure port ethernet1/0/1, which connects to the server, as a trust port, and enable the vendor tag strip function.

```
Switch(config-if-ethernet1/0/1)#pppoe intermediate-agent trust
Switch(config-if-ethernet1/0/1)#pppoe intermediate-agent vendor-tag strip
```

Step3: Enable PPPoE IA on port ethernet1/0/2 in vlan1 and port ethernet1/0/3 in vlan 1234.

```
Switch(config-if-ethernet1/0/2)#pppoe intermediate-agent
Switch(config-if-ethernet1/0/3)#pppoe intermediate-agent
```

Step4: Set the PPPoE intermediate agent access-node-id to abcd.

```
Switch(config)#pppoe intermediate-agent type tr-101 circuit-id access-node-id
abcd
```

Step5: Set the circuit ID to aaaa and the remote ID to xyz on port ethernet1/0/3.

```
Switch(config-if-ethernet1/0/3)#pppoe intermediate-agent circuit-id aaaa
Switch (config-if-ethernet1/0/3)#pppoe intermediate-agent remote-id xyz
```

The added vendor tag of port ethernet1/0/2 has the circuit-id value "abcd eth 01/002:0001" and the remote-id value "0a0b0c0d0e0f".

The added vendor tag of port ethernet1/0/3 has the circuit-id value "aaaa" and the remote-id value "xyz".

Typical configuration (2):

Step1: Enable PPPoE IA globally on the switch, whose MAC address is 0a0b0c0d0e0f.

```
Switch(config)#pppoe intermediate-agent
```

Step2: Configure port ethernet1/0/1, which connects to the server, as a trust port, and enable the vendor tag strip function.

```
Switch(config-if-ethernet1/0/1)#pppoe intermediate-agent trust
Switch(config-if-ethernet1/0/1)#pppoe intermediate-agent vendor-tag strip
```

Step3: Enable PPPoE IA on port ethernet1/0/2 in vlan1 and port ethernet1/0/3 in vlan 1234.

```
Switch(config-if-ethernet1/0/2)#pppoe intermediate-agent
Switch(config-if-ethernet1/0/3)#pppoe intermediate-agent
```

Step4: Set the PPPoE intermediate agent access-node-id to abcd.

```
Switch(config)#pppoe intermediate-agent type tr-101 circuit-id access-node-id
abcd
```

Step5: Set the PPPoE intermediate agent identifier-string to "efgh", the option mode to spv, the delimiter between Slot ID and Port ID to "#", and the delimiter between Port ID and VLAN ID to "/".

```
Switch(config)#pppoe intermediate-agent type tr-101 circuit-id identifier-
string efgh option spv delimiter # delimiter /
```

Step6: Set the circuit-id value to bbbb on port ethernet1/0/2.

```
Switch(config-if-ethernet1/0/2)#pppoe intermediate-agent circuit-id bbbb
```

Step7: Set the remote-id to xyz on port ethernet1/0/3.

```
Switch(config-if-ethernet1/0/3)#pppoe intermediate-agent remote-id xyz
```

The added vendor tag of port ethernet1/0/2 has the circuit-id value "bbbb" and the remote-id value "0a0b0c0d0e0f".

The added vendor tag of port ethernet1/0/3 has the circuit-id value "efgh eth 01#003/1234" and the remote-id value "xyz".

7.13.4 PPPoE Intermediate Agent Troubleshooting

- PPPoE intermediate agent must be enabled globally before it can run on ports.
- Configure at least one trust port, which connects to the server.
- The vendor tag strip function must be configured on a trust port.
- The circuit ID override priority is: pppoe intermediate-agent circuit-id < pppoe intermediate-agent identifier-string option delimiter < pppoe intermediate-agent access-node-id.

7.14 QoS

7.14.1 QoS

7.14.1.1 Introduction to QoS

QoS (Quality of Service) is a set of capabilities that allow you to create differentiated services for network traffic, thereby providing better service for selected traffic. QoS guarantees consistent and predictable data transfer service to meet application requirements. QoS cannot generate extra bandwidth, but it provides more effective bandwidth management according to application requirements and network management policies.

7.14.1.1.1 QoS Terms

QoS: Quality of Service. It guarantees consistent and predictable data transfer service to meet application requirements. QoS cannot generate new bandwidth, but it provides more effective bandwidth management according to application requirements and network management policies.

QoS Domain: A network topology formed by QoS-capable devices that provides Quality of Service.

CoS: Class of Service. The classification information carried in Layer 2 802.1Q frames, occupying 3 bits of the tag field in the frame header. It is called the user priority and ranges from 0 to 7.

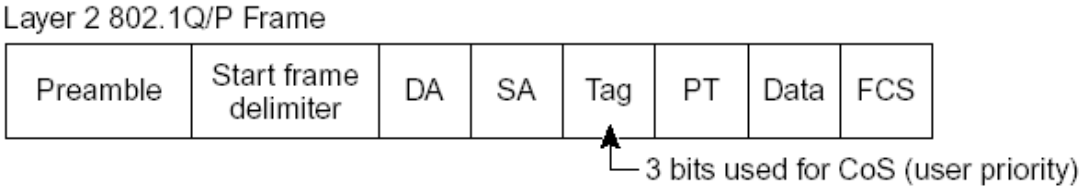


Figure 7-333 CoS Priority

ToS: Type of Service. A one-byte field in the Layer 3 IPv4 packet header that indicates the service type of IP packets. The ToS field can carry an IP Precedence value or a DSCP value.

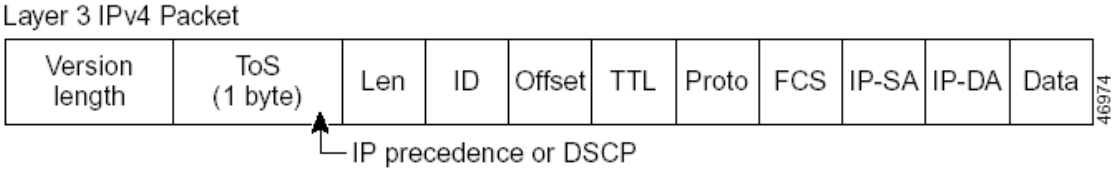


Figure 7-34 ToS Priority

IP Precedence: IP priority. Classification information carried in the Layer 3 IP packet header, occupying 3 bits, in the range of 0 to 7.

DSCP: Differentiated Services Code Point. Classification information carried in the Layer 3 IP packet header, occupying 6 bits, in the range of 0 to 63, and backward compatible with IP Precedence.

Internal Priority: The internal priority setting of the switch chip. Its valid range depends on the chip. It is abbreviated as Int-Prio or IntP.

Drop Precedence: When packets are processed, packets with a higher drop precedence are dropped first. The range is 0-1. It is abbreviated as Drop-Prec or DP.

Classification: The ingress action of QoS that classifies packet traffic according to the classification information carried in packets and ACLs.

Policing: The ingress action of QoS that specifies the policing policy and manages the classified packets.

Remark: The ingress action of QoS that permits, degrades, or discards packets according to the policing policies.

Scheduling: The QoS *egress* action. Packets are placed in the corresponding egress queues according to their internal priority, and then sent or dropped according to the drop precedence, the scheduling algorithm, and the egress queue weights.

7.14.1.1.2 QoS Implementation

To implement QoS in the switch software, a general and mature reference model is required. QoS cannot create new bandwidth, but it can make the best use of current bandwidth resources through adjustment and configuration. A fully implemented QoS can manage network traffic completely. The following describes QoS as accurately as possible.

IP data transfer specifications cover only the source and destination addresses and services, and ensure correct packet delivery by using OSI Layer 4 or higher protocols such as TCP. However, IP does not provide a mechanism for allocating and protecting packet transmission bandwidth; it provides bandwidth on a best-effort basis. This is acceptable for services such as Mail and FTP, but for the growing volume of multimedia and e-business data, best-effort delivery cannot meet bandwidth and low-latency requirements.

Based on differentiated services, QoS assigns a priority to each packet at the ingress. The classification information is carried in the Layer 3 IP packet header or the Layer 2 802.1Q frame header. QoS provides the same service to packets of the same priority and different handling for packets of different priorities. A QoS-enabled switch or router can provide different bandwidth according to the packet classification information, can remark the classification information according to the configured policing policies, and may discard some low-priority packets when bandwidth is insufficient.

If the devices at each hop in a network support differentiated services, an end-to-end QoS solution can be created. QoS configuration is flexible; its complexity depends on the network topology, the devices, and the analysis of incoming and outgoing traffic.

7.14.1.1.3 Basic QoS Model

The basic QoS model consists of four parts: Classification, Policing, Remark, and Scheduling. Classification, policing, and remark are sequential ingress actions, and queuing and scheduling are QoS egress actions.

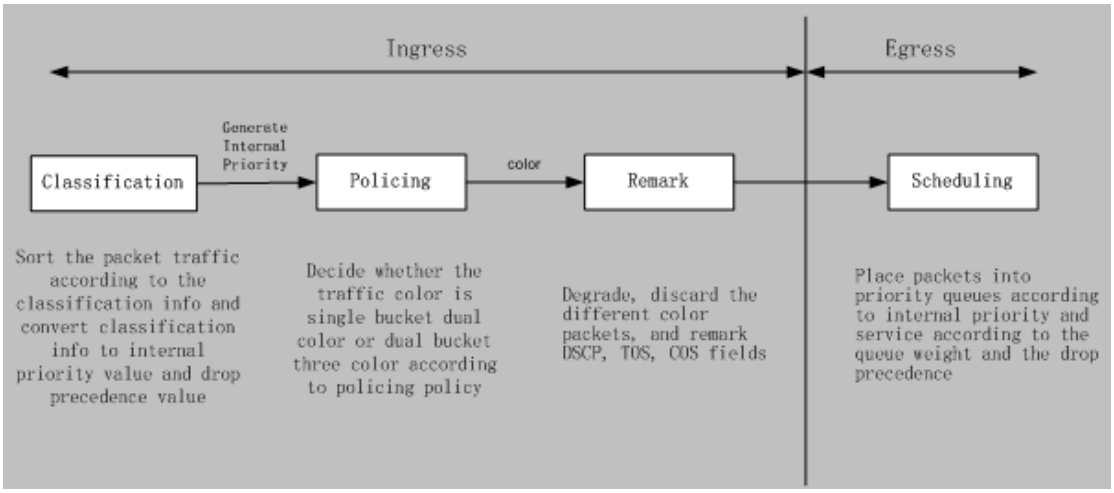


Figure 7-36 Basic QoS Model

Classification: Classifies traffic according to packet classification information and generates the internal priority based on that information. Classification differs by packet type; the following flowchart explains it in detail.

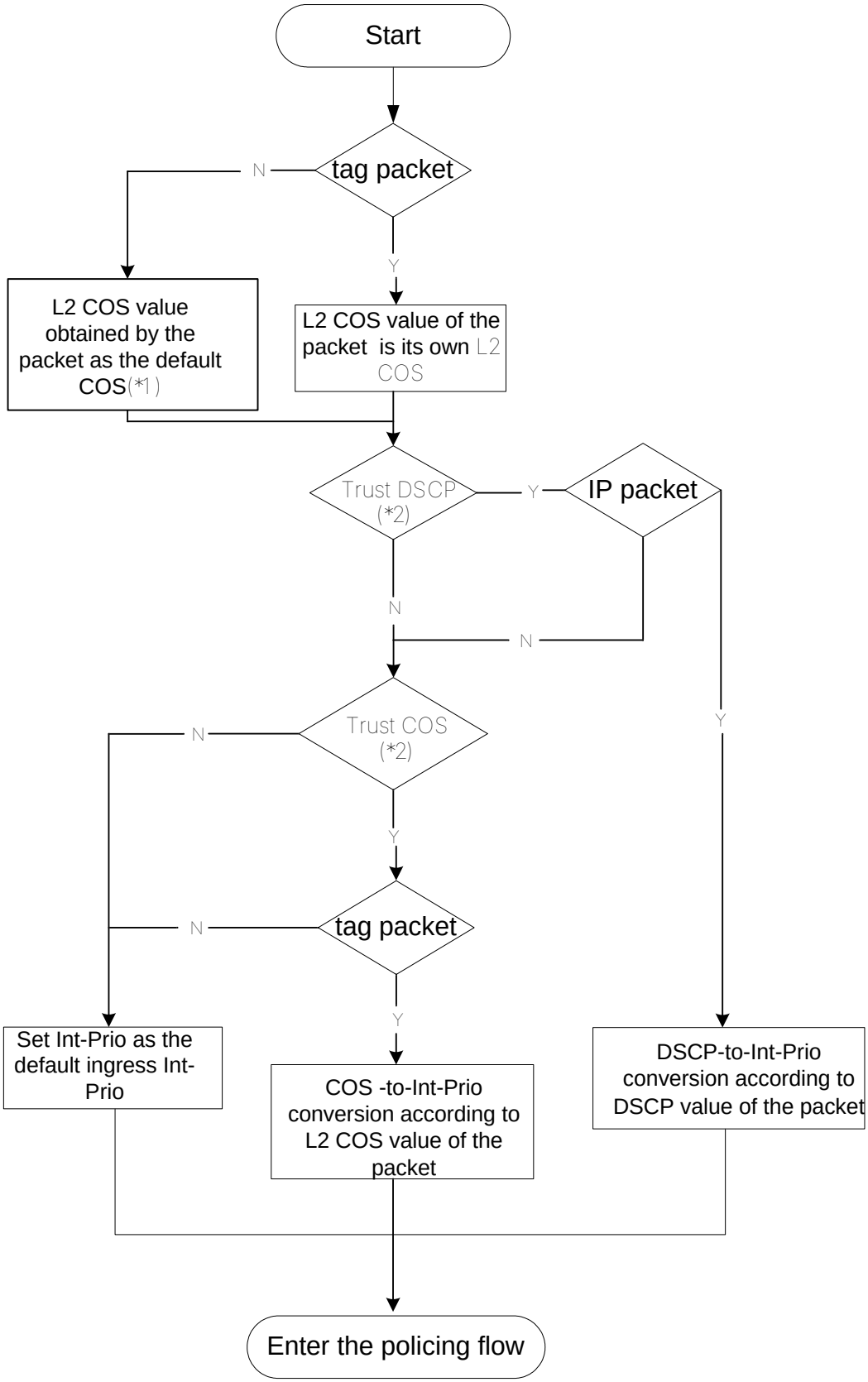


Figure 7-37 Classification Process

Note 1: The L2 CoS value is a property of the packet and is unrelated to the internal priority obtained in the following flow.

Note 2: Trust DSCP and Trust CoS can be configured at the same time, with the priority DSCP > CoS.

Policing and remark: Each packet in the classified ingress traffic is assigned an internal priority value and can be policed and remarked.

Policing can be performed per flow, with different policies allocating bandwidth to classified traffic. The bandwidth policy can be single bucket dual color or dual bucket three color. Traffic is assigned different colors and can be discarded or passed; a remarking action can be added for passed packets. Remarking replaces the original higher Int-Prio value of the packet with a new, lower Int-Prio value. The CoS and DSCP fields are modified at the egress according to the new Int-Prio. The following flowchart describes these operations.

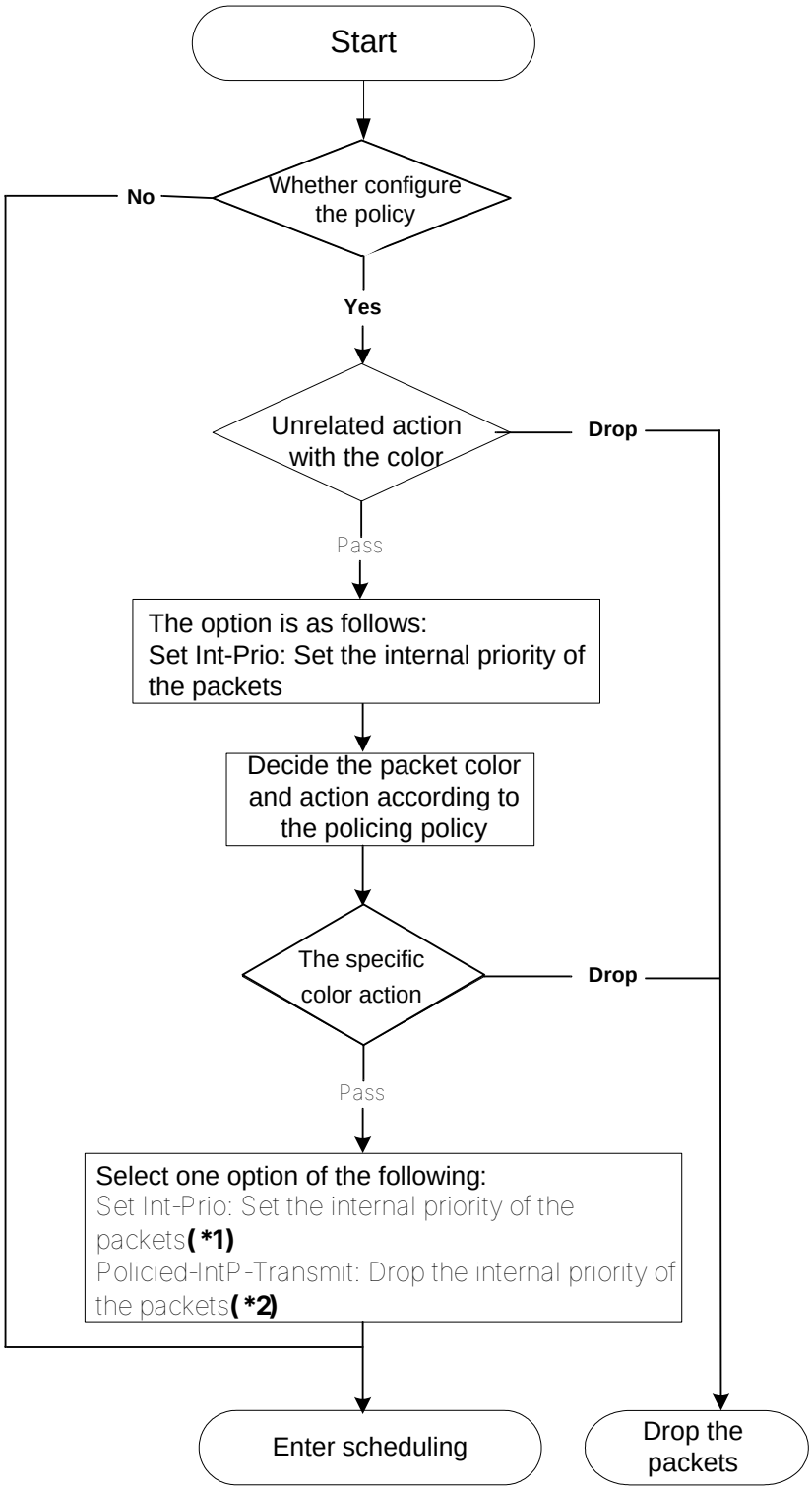


Figure 7-38 Policing and Remarking Process

Note 1: A later Int-Prio setting overrides an earlier one: Set Int-Prio in a color-specific action overrides Set Int-Prio in a color-independent action.

Note 2: The internal priority of packets is lowered according to the IntP-to-IntP map. Source Int-Prio refers to the Int-Prio obtained in the classification flow or set by a color-independent action.

Queuing and scheduling: Egress packets carry an internal priority. Scheduling assigns packets to different priority queues according to their internal priority, and then forwards them according to the priority queue weights and the drop precedence. The following flowchart describes the scheduling operation.

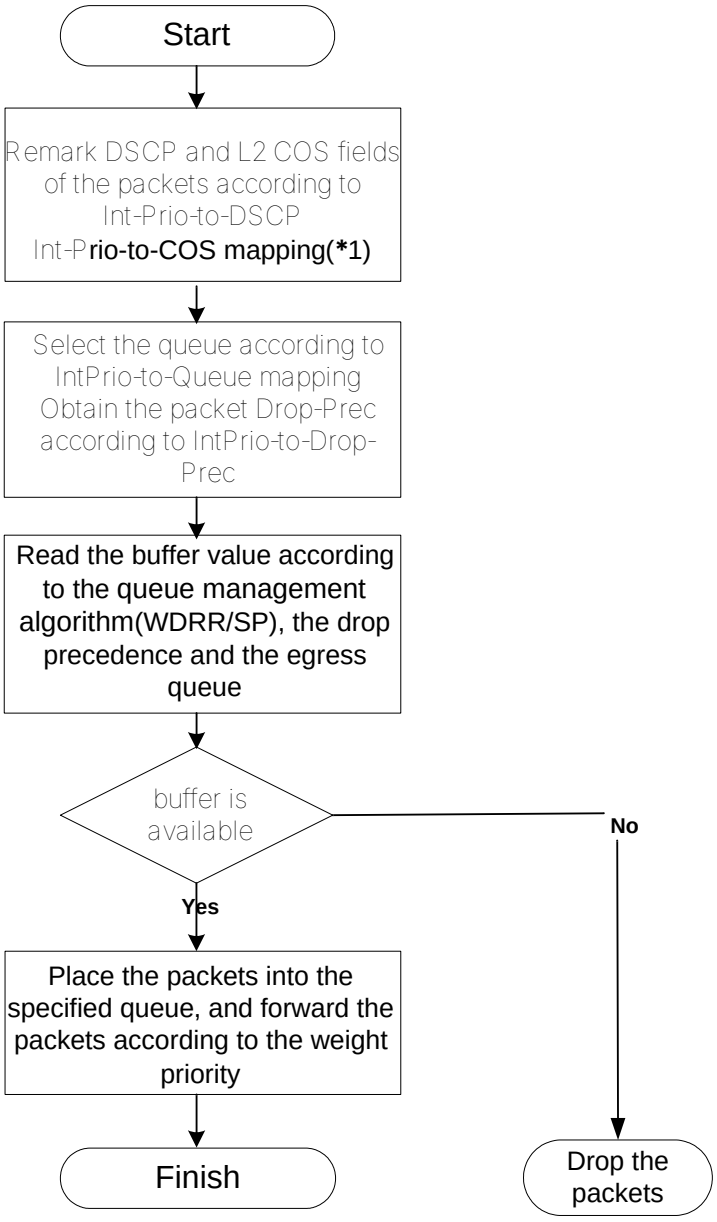


Figure 7-39 Queuing and Scheduling Process

Note 1: Configure pass-through-cos and pass-through-dscp at the ingress to prevent rewriting of the L2 CoS priority and DSCP value. At the egress, the L2 CoS priority and DSCP value are derived from the final Int-Prio of the packet, and pass-through-cos and pass-through-dscp determine whether they are rewritten.

7.14.1.2 QoS Configuration Task List

Configure a class map

Set up classification rules based on ACL, CoS, VLAN ID, IPv4 Precedence, DSCP, and IPv6 flow label to classify data streams. Different classes of data streams are processed with different policies.

Configure a policy map

After data streams are classified, create a policy map, associate it with a previously created class map, and enter class mode. Different policies (such as bandwidth limits, priority degrading, and assigning new DSCP values) can then be applied to different data streams. You can also define a policy set that several classes in a policy map can use.

Apply QoS to ports or VLAN interfaces

Configure the trust mode for ports or bind policies to ports. A policy takes effect on a port only when it is bound to that port.

A policy can also be bound to a specific VLAN.

Do not apply policy maps to a VLAN and its ports at the same time; otherwise, the policy map on the port takes priority.

Configure the queue management algorithm

Configure the queue management algorithm, such as sp or wdrr.

Configure QoS mapping

Configure the mappings from CoS to IntP and DSCP to IntP, and from IntP to DSCP, CoS, IntP, DP, or queue.

1. Configure class map.

Command	Explanation
Global Mode	
class-map <*class-map-name*> **no class-map <*class-map-name*>**	Creates a class map and enters class map mode. The **no class-map <*class-map-name*>** command deletes the specified class map.
match {access-group <acl-index-or-name> | ip dscp <dscp-list>| ip precedence <ip-precedence-list>| ipv6 access-group <acl-index-or-name> | ipv6 dscp <dscp-list>| ipv6 flowlabel <flowlabel-list>|vlan <vlan-list> | cos <*cos-list*>| vlan range <vlan-list>} **no match {access-group | ip dscp | ip precedence | ipv6 access-group | ipv6 dscp | ipv6 flowlabel | vlan | cos| vlan range}**	Sets the match criteria of the class map (classifying data streams by ACL, CoS, VLAN ID, IPv4 Precedence, IPv6 flow label, DSCP, and so on). The no command deletes the specified match criterion.

2. Configure a policy map

Command	Explanation
Global Mode	
policy-map <*policy-map-name*> **no policy-map <*policy-map-name*>**	Creates a policy map and enters policy map mode. The no command deletes the specified policy map.
class <*class-map-name*> [insert-before <class-map-name>] **no class <*class-map-name*>**	Associates a class with the policy map after the policy map is created. In class mode, different policies or new DSCP values can be applied to different data streams. The no command deletes the specified class.
set internal priority <new-intp> **no set internal priority**	Assigns a new internal priority to the classified traffic. The no command cancels the assigned value.
Single bucket mode: **policy <bits_per_second>**	Configures a policy for the classified traffic. The non-aggregate policy

<normal_burst_bytes> ({exceed-action ACTION}) **Dual bucket mode:** **policy <bits_per_second> <normal_burst_bytes> [pir <peak_rate_bps>]	<maximum_burst_bytes> [{exceed-action ACTION | violate-action ACTION }]** **ACTION definition:** **drop | transmit | set-internal-priority <intp_value> | policied-intp-transmit** **no policy**	command supports three colors. Depending on the token bucket working mode (single rate single bucket, single rate dual bucket, or dual rate dual bucket), different actions are set for packets of different colors. The no command deletes the configuration.
policy aggregate <*aggregate-policy-name*> **no policy aggregate**	Applies an aggregate policy to the classified traffic. The no command deletes the specified policy set.
accounting **no accounting**	Enables statistics for the classified traffic. When this function is enabled in policy class map mode, statistics are collected for the traffic of the policy class map. In single bucket mode, packets passing the policy can only be red or green, and the output shows packets of two colors (green and red). In dual bucket mode, packets have three colors (green, red, and yellow).
Policy Class Map Mode	
drop **no drop** **transmit** **no transmit**	Drops or transmits the traffic that matches the class. The no command cancels the assigned action.

3. Apply QoS to port or VLAN interface

Command	Explanation
Port Mode	
mls qos trust {cos | dscp} **no** mls **qos trust {cos | dscp}**	Configures the port trust mode. The no command disables the current trust mode of the port.
mls qos cos {<*default-cos*>} **no mls qos cos**	Configures the default CoS value of the port. The no command restores the default setting.
mls qos internal-priority {<*default-intp*>} **no mls qos internal-priority**	Configures the default internal priority of the port. The no command restores the default setting.
service-policy input <*policy-map-name*> **no service-policy input {<*policy-map-name*>}**	Applies a policy map to the port. The no command deletes the specified policy map applied to the port, or all policy maps applied in the ingress direction of the port. Egress policy maps are currently not supported.
pass-through-cos **no pass-through-cos**	Prevents the L2 CoS value of packets from being rewritten at the egress. The no command allows the L2 CoS value to be rewritten.
pass-through-dscp **no pass-through-dscp**	Prevents the DSCP value of packets from being rewritten at the egress. The no command allows the DSCP value to be rewritten.

Global Mode	
service-policy input <*policy-map-name*> vlan <*vlan-list*> **no service-policy input {<*policy-map-name*>} vlan <*vlan-list*>**	Applies a policy map to the specified VLAN interface. The no command deletes the specified policy map applied to the VLAN interface, or all policy maps applied in the ingress direction of the VLAN interface.

4. Configure queue management algorithm and weight

Command	Explanation
Port Mode	
mls qos queue algorithm {sp | wdrr} **no mls qos queue algorithm**	Sets the queue management algorithm. The default algorithm is wdrr.
Global Mode	
mls qos queue wdrr weight <weight0..weight7> **no mls qos queue wdrr weight**	Sets the WDRR queue weights for all ports globally. The default queue weights are 1 1 1 1 1 1 1 1.

5. Configure QoS mapping

Command	Explanation
Global Mode	
mls qos map {cos-intp <intp1…intp8> | dscp-intp <in-dscp list> to <intp> | intp-cos <intp list> to <out-cos> | intp-dp <intp list> to <out-dp> | intp-dscp <intp list> to <out-dscp>| intp-intp <color> <intp list> to <intp> | intp-queue <intp list> to <out-queue> } **no mls qos map {cos-intp | dscp-intp | intp-cos | intp-dp | intp-dscp | intp-intp <color> | intp-queue}**	Sets the QoS priority mappings. The no command restores the default mappings.

6. Clear accounting data of the specific ports or VLANs

Command	Explanation
Admin Mode	
clear mls qos statistics [interface <interface-name> | vlan <vlan-id>]	Clears the accounting data of the policy maps on the specified port or VLAN. If no parameter is specified, clears the accounting data of all policy maps.

7. Show configuration of QoS

Command	Explanation
Admin Mode	
show mls qos maps [cos-intp | dscp-intp | intp-intp | intp-cos | intp-dscp | intp-dp | intp-queue]	Displays the QoS mapping configuration.
show class-map [<class-map-name>]	Displays the QoS class map information.
show policy-map [<policy-map-name>]	Displays the QoS policy map information.
show mls qos aggregate-policy [<aggregate-policy-name>]	Displays the QoS aggregate policy configuration.
show mls qos interface [<interface-id>] [policy | queuing]	Displays the QoS configuration of a port.

show mls qos vlan <v-id>	Displays the QoS configuration of a VLAN interface.

7.14.1.3 QoS Example

Example 1:

Enable QoS, change the global egress queue weights to 1:1:2:2:4:4:8:8, set port ethernet 1/0/1 to trust CoS mode without changing the DSCP value, and set the default CoS value of the port to 5.

The configuration steps are as follows:

```
Switch#config
Switch(config)# mls qos queue weight 1 1 2 2 4 4 8 8
Switch(config)#interface ethernet 1/0/1
Switch(Config-If-Ethernet 1/0/1)#mls qos trust cos
Switch(Config-If-Ethernet 1/0/1)#pass-through-dscp
Switch(Config-If-Ethernet1/0/1)#mls qos cos 5
```

Configuration result:

When QoS is enabled in Global Mode, the egress queue bandwidth ratio of each port is 1:1:2:2:4:4:8:8. Packets with a CoS value received on port ethernet1/0/1 are mapped to the internal priority according to their CoS value; CoS values 0 to 7 correspond to egress queues 1, 2, 3, 4, 5, 6, 7, and 8 respectively. Packets without a CoS value are assigned the default CoS value 5 and placed in queue 6. The DSCP values of all packets passing through remain unchanged.

Example 2:

On port ethernet1/0/2, limit the bandwidth of packets from segment 192.168.1.0 to 10 Mb/s with a burst value of 4 MB; all packets exceeding this bandwidth are dropped.

The configuration steps are as follows:

```
Switch#config
Switch(config)#access-list 1 permit 192.168.1.0 0.0.0.255
Switch(config)#class-map c1
Switch(Config-ClassMap-c1)#match access-group 1
Switch(Config-ClassMap-c1)#exit
Switch(config)#policy-map p1
Switch(Config-PolicyMap-p1)#class c1
Switch(Config-PolicyMap-p1-Class-c1)#policy 10000 4000 exceed-action drop
Switch(Config-PolicyMap-p1-Class-c1)#exit
Switch(Config-PolicyMap-p1)#exit
Switch(config)#interface ethernet 1/0/2
Switch(Config-If-Ethernet1/0/2)#service-policy input p1
```

Configuration result:

ACL 1 is configured to match segment 192.168.1.0. QoS is enabled globally, a class map named c1 is created to match ACL 1, and a policy map named p1 is created that references c1 and sets the appropriate policy to limit the bandwidth and burst value. The policy map is applied to port ethernet1/0/2. After this configuration, the bandwidth of packets from segment 192.168.1.0 received on port ethernet 1/0/2 is limited to 10 Mb/s with a burst value of 4 MB, and all packets from that segment exceeding this bandwidth are dropped.

Example 3:

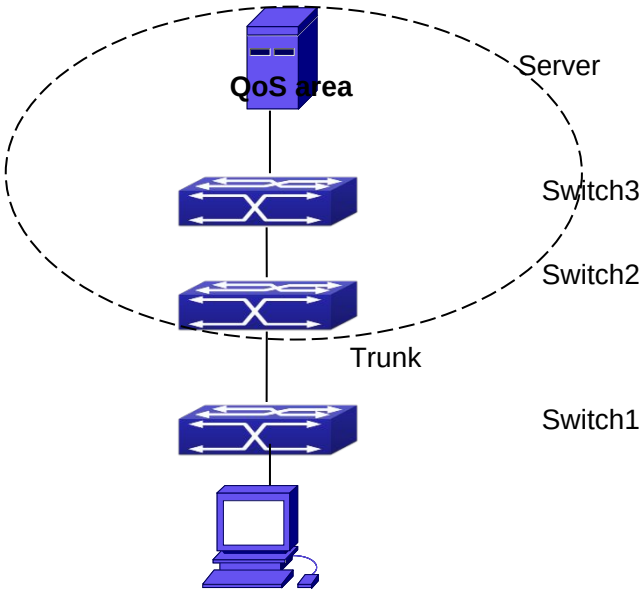


Figure 7-340 Typical QoS Topology

As shown in the figure, the area inside the box is a QoS domain. Switch1 classifies different traffic and assigns different IP precedences. For example, on port ethernet1/0/1, set the CoS precedence of packets from segment 192.168.1.0 to 5 (set the internal priority to 40; with the default intp-dscp mapping of 40-40, the corresponding IP precedence is 5). The port connecting to Switch2 is a trunk port. On Switch2, set port ethernet 1/0/1, which connects to Switch1, to trust DSCP. Thus, within the QoS domain, packets of different priorities go to different queues and receive different bandwidth.

The configuration steps are as follows:

QoS configuration in Switch1:

```
Switch#config
Switch(config)#access-list 1 permit 192.168.1.0 0.0.0.255
Switch(config)#class-map c1
Switch(Config-ClassMap-c1)#match access-group 1
Switch(Config-ClassMap-c1)#exit
Switch(config)#policy-map p1
Switch(Config-PolicyMap-p1)#class c1
Switch(Config-PolicyMap-p1-Class-c1)#set ip precedence 40
Switch(Config-PolicyMap-p1-Class-c1)#exit
Switch(Config-PolicyMap-p1)#exit
Switch(config)#interface ethernet 1/0/1
Switch(Config-If-Ethernet1/0/1)#service-policy input p1
```

QoS configuration on Switch2:

```
Switch#config
Switch(config)#interface ethernet 1/0/1
Switch(Config-If-Ethernet1/0/1)#mls qos trust dscp
```

7.14.1.4 QoS Troubleshooting

- trust cos and EXP can be used together with other trust modes or policy maps.
- trust dscp can be used together with other trust modes or policy maps. This configuration applies to IPv4 and IPv6 packets.

- ☝ trust exp, trust dscp, and trust cos can be configured at the same time, with the priority EXP > DSCP > COS.
- ☝ If a dynamic VLAN (MAC VLAN, voice VLAN, IP subnet VLAN, or protocol VLAN) is configured, the CoS value of the packet equals the CoS value of the dynamic VLAN.
- ☞ Currently, it is not recommended to apply policy maps to a VLAN and its ports at the same time.

7.14.2 PBR

7.14.2.1 Introduction to PBR

PBR (Policy-Based Routing) determines the next hop of data packets based on policy information such as source address, destination address, IP priority, ToS value, IP protocol, source port number, and destination port number.

7.14.2.2 PBR Configuration

1. Configure a class-map
2. Set the match criteria of the class-map
3. Configure a policy-map
4. Associate the policy map with a class map
5. Configure the next-hop IPv4 address
6. Bind the policy map to a port
7. Bind the policy map to a VLAN

1. Configure a class-map

Command	Explanation
Global Mode	
class-map <*class-map-name*> **no class-map <*class-map-name*>**	Creates or deletes a class map.

2. Set match standard of the class-map

Command	Explanation
Class Map Mode	
match ip {access-group <*acl-index-or-name*>} **no match ip {access-group}**	Sets the match criteria of the class map.

3. Configure a policy-map

Command	Explanation
Global Mode	
policy-map <*policy-map-name*> **no policy-map <*policy-map-name*>**	Creates or deletes a policy map.

4. Configure a policy map corresponding to a class map

Command	Explanation
Policy Map Mode	
class <*class-map-name*> **no class <*class-map-name*>**	Associates a class map and enters policy class map mode.

5. Configure nexthop IPv4 address

Command	Explanation
Policy Class Map Mode	

set ipv4 [default] nexthop [vrf <vrf>] <*nexthop-ip*> **no set ipv4 nexthop**	Sets the next-hop IP address for the classified traffic. The no command cancels the assigned value.

6. Configure the port binding policy map

Command	Explanation
Port Mode	
service-policy {input <*policy-map-name*> | output <*policy-map-name*>} **no service-policy {input <*policy-map-name*> | output <*policy-map-name*>}**	Applies a policy map to the specified port. Only one policy map can be applied in each direction of each port. Egress policy maps are not supported yet.

7. Configure the VLAN binding policy map

Command	Explanation
Global Mode	
service-policy input <*policy-map-name*> vlan <*vlan-list*> **no service-policy input <*policy-map-name*> vlan <*vlan-list*>**	Applies a policy map to the specified VLAN interface. The no command deletes the specified policy map applied to the VLAN interface.

7.14.2.3 PBR Examples

Example:

On port ethernet1/0/1, apply policy-based routing to packets from segment 192.168.1.0/24 and set the next hop to 218.31.1.119. The local network uses IP addresses within 192.168.0.0/16. To ensure normal communication within the local network, packets from 192.168.1.0/24 to local addresses in 192.168.0.0/16 are not policy-routed. The interface address of this device in 192.168.1.0/24 is 192.168.1.1.

Configuration procedure is as follows:

```
Switch#config
Switch(config)#ip access-list extended a1
Switch(Config-IP-Ext-Nacl-a1)# permit ip 192.168.1.0 0.0.0.255 192.168.0.0
0.0.255.255
Switch(Config-IP-Ext-Nacl-a1)#exit
Switch(config)#ip access-list extended a2
Switch(Config-IP-Ext-Nacl-a1)# permit ip 192.168.1.0 0.0.0.255 any-
destination
Switch(Config-IP-Ext-Nacl-a1)#exit
Switch(config)#mls qos
Switch(config)#class-map c1
Switch(Config-ClassMap-c1)#match access-group a1
Switch(Config-ClassMap-c1)# exit
Switch(config)#class-map c2
Switch(Config-ClassMap-c2)#match access-group a2
Switch(Config-ClassMap-c2)# exit
Switch(config)#policy-map p1
Switch(Config-PolicyMap-p1)#class c1
Switch(Config-PolicyMap-p1-Class-c1)#set ip nexthop 192.168.1.1
Switch(Config-PolicyMap-p1-Class-c1)#exit
Switch(Config-PolicyMap-p1)#class c2
Switch(Config-PolicyMap-p1-Class-c2)#set ip nexthop 218.31.1.119
Switch(Config-PolicyMap-p1-Class-c2)#exit
Switch(config)#interface ethernet 1/0/1
Switch(Config-If-Ethernet1/0/1)#service-policy input p1
```

Configuration results:

First, configure ACLs a1 and a2: a1 matches source IP segment 192.168.1.0/24 and destination IP segment 192.168.0.0/16, and a2 matches source IP segment 192.168.1.0/24. Enable QoS globally

and create two class maps: c1, which matches ACL a1, and c2, which matches ACL a2. Then create a policy map that references c1. The interface address of this device in 192.168.1.0/24 is set as 192.168.1.1, the next hop is set to 218.31.1.119, and the policy map is applied to port ethernet1/0/1. After that, all packets received on port ethernet 1/0/1 from segment 192.168.1.0/24 are forwarded through 218.31.1.119, except those from segment 192.168.0.0/16, which are still forwarded through normal L3 routing.

7.14.3 IPv6 PBR

7.14.3.1 Introduction to PBR (Policy-Based Routing)

Policy-based routing gives network administrators more powerful control over packet forwarding and storage than traditional routing protocols. Traditionally, routers forward packets according to destination addresses by using the routing table derived from routing protocols. Policy-based routing is more powerful and flexible because it allows administrators to choose the forwarding path based not only on the destination address but also on packet size or source IP address. Policies can be defined for load balancing across multiple routers or for the QoS of the total traffic forwarded on each line.

PBR (Policy-Based Routing) specifies, by policy, the next hop for forwarding a data packet according to the source address, destination address, IP priority, ToS value, IP protocol, source port, destination port, and other information in the IP packet.

7.14.3.2 PBR Configuration Task Sequence

1. Configure a class-map
2. Set the match criteria in the class-map
3. Configure a policy-map
4. Associate a policy with a class-map
5. Configure the next-hop IPv6 address
6. Bind the policy map to a port
7. Bind the policy map to a VLAN

1. Configure a class-map

Command	Explanation
Global Mode	
class-map <*class-map-name*> **no class-map <*class-map-name*>**	Creates or deletes a class map.

2. Set the match standard in the class-map

Command	Explanation
Class Map Mode	
match ipv6 {access-group <*acl-index-or-name*>} **no match ipv6 {access-group }**	Sets the match criteria in the class map.

3. Configure a policy-map

Command	Explanation
Global Mode	
policy-map <*policy-map-name*> **no policy-map <*policy-map-name*>**	Creates or deletes a policy map.

4. Configure to correlate a policy and a class-map

Command	Explanation

Policy Map Mode	
class <*class-map-name*> **no class <*class-map-name*>**	Associates a class map and enters policy class map mode.

5. Configure the next hop IPv6 address

Command	Explanation
Policy Class Map Mode	
set ipv6 [default] nexthop [vrf <vrf>] <*nexthop-ip*> **no set ipv6 nexthop**	Sets the next-hop IP address for the classified traffic. The no command cancels the assigned value.

6. Configure the port binding policy-map

Command	Explanation
Port Mode	
service-policy {input <*policy-map-name*> | output <*policy-map-name*>} **no service-policy {input <*policy-map-name*> | output <*policy-map-name*>}**	Applies a policy map to the specified port. Only one policy map can be applied in each direction of each port. Egress policy maps are not supported yet.

7. Configure the VLAN binding policy map

Command	Explanation
Global Mode	
service-policy input <*policy-map-name*> vlan <*vlan-list*> **no service-policy input <*policy-map-name*> vlan < *vlan-list* >**	Applies a policy map to the specified VLAN interface. The no command deletes the specified policy map applied to the VLAN interface.

7.14.3.3 PBR Examples

Example:

On port ethernet 1/0/1, where the default gateway address of this device is 3000::1, apply policy routing to packets whose source IP address is within segment 2000::/64, with the next hop 3100::2.

The configuration steps are as follows:

```
Switch#config
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ipv6 address 2000::1/64
Switch(Config-if-Vlan1)#ipv6 neighbor 2000::2 00-00-00-00-00-01 interface
Ethernet 1/0/1
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)#ipv6 address 3000::1/64
Switch(Config-if-Vlan2)#ipv6 neighbor 3000::2 00-00-00-00-00-02 interface
Ethernet 1/0/2
Switch(config)#interface vlan 3
Switch(Config-if-Vlan3)#ipv6 address 3100::1/64
Switch(Config-if-Vlan3)#ipv6 neighbor 3100::2 00-00-00-00-00-03 interface
Ethernet 1/0/5
Switch(config)# ipv6 access-list extended b1
Switch(Config-IPv6-Ext-Nacl-b1)# permit 2000:: /64 any-destination
Switch(Config-IPv6-Ext-Nacl-b1)#exit
Switch(config)#class-map c1
Switch(config-ClassMap)#match ipv6 access-group b1
Switch(config-ClassMap)# exit
Switch(config)#policy-map p1
Switch(config-PolicyMap)#class c1
Switch(config-Policy-Class)# set ipv6 nexthop 3100::2
Switch(config--Policy-Class)#exit
Switch(config-PolicyMap)#exit
```

```
Switch(config)#interface ethernet 1/0/1
Switch(Config-Ethernet1/0/1)#service-policy input p1
```

Configuration result:

First, configure an ACL named b1 containing one entry that matches (permits) source IP segment 2000::/64. Enable QoS globally, create class map c1, and match ACL b1 in the class map. Create policy map p1, reference c1 in p1, and set the next hop to 3100::2. Apply this policy map to port ethernet 1/0/1. After that, packets received on port ethernet 1/0/1 whose source IP addresses are within segment 2000::/64 are forwarded through 3100::2.

7.14.3.4 PBR Troubleshooting Help

- ☞ Currently, a policy map can be bound only to the ingress of a port, not to the egress.
- ☞ Because hardware resources are limited, the switch displays a related message if the configured policy is too complex.

7.15 Flow-Based Redirection

7.15.1 Introduction to Flow-Based Redirection

Flow-based redirection enables the switch to send data frames that meet specific conditions (specified by an ACL) to another specified port. Frames that meet the same condition form a class of flow. The ingress port of the frames is called the source port of redirection, and the specified egress port is called the destination port of redirection. Flow-based redirection typically has two applications: 1. connecting a protocol analyzer or an RMON monitor to the destination port of redirection to monitor and manage the network and diagnose network problems; 2. applying a special forwarding policy to a specific type of data frame.

On a source port of redirection, the switch can designate only one destination port of redirection for the same class of flow, but it can designate different destination ports for different classes of flows. The same class of flow can be applied to different source ports.

7.15.2 Flow-Based Redirection Configuration Task Sequence

1. Configure flow-based redirection
2. Display the current flow-based redirection configuration
3. Configure VLAN redirection on the port
4. Configure the maximum number of redirected VLANs on each port

1. Flow-based redirection configuration

Command	Explanation
Port Mode	
access-group <*aclname*> redirect to interface [ethernet <*IFNAME*>|<*IFNAME*>] **no access-group <*aclname*> redirect**	Specifies flow-based redirection for the port. The **no access-group <*aclname*> redirect** command deletes flow-based redirection.

2. Check the current flow-based redirection configuration

Command	Explanation
Global Mode/Admin Mode	
show flow-based-redirect {interface [ethernet <*IFNAME*> |<*IFNAME*>]}	Displays the current flow-based redirection information of the system or port.

3. Configure the vlan redirection function of the port

Command	Explanation
Port Mode	
match vlan <1-4096> redirect interface (ethernet|) IFNAME **no match vlan <1-4096> redirect**	Configures VLAN redirection on the port.
port-redirect match vlan <1-4094> source-port interface (ethernet|) IFNAME destination-port interface (ethernet|) IFNAME **no port-redirect match vlan <1-4094> source-port interface (ethernet|) IFNAME destination-port interface (ethernet|) IFNAME**	Configures VLAN redirection on the port.

4. Configure the maximum number of vlan of redirect on each port

Command	Explanation
Global Mode	
vlan-port-redirect vlan maximum <1-1000> **no vlan-port-redirect vlan maximum**	Configures the maximum number of redirected VLANs on each port.

7.15.3 Flow-Based Redirection Examples

Example:

Configuration requirement: redirect frames with source IP 192.168.1.111 received on port 1 to port 6, that is, send frames with source IP 192.168.1.111 received on port 1 out through port 6.

Configuration steps:

1: Configure an ACL that matches source IP 192.168.1.111;

2: Apply flow-based redirection based on this flow to port 1.

The following is the configuration procedure:

```
Switch(config)#access-list 1 permit host 192.168.1.111
Switch(config)#interface ethernet 1/0/1
Switch(Config-If-Ethernet1/0/1)# access-group 1 redirect to interface
ethernet 1/0/6
```

7.15.4 Flow-Based Redirection Troubleshooting Help

If flow-based redirection configuration fails, check whether it is caused by the following:

- ☞ The flow type (ACL) can only be a numbered standard IP ACL, numbered extended IP ACL, named standard IP ACL, named extended IP ACL, numbered standard IPv6 ACL, or named standard IPv6 ACL.
- ☞ The **Timerange** and **Portrange** parameters cannot be set in the ACL, and the ACL type must be Permit.
- ☞ The redirection port must be a 1000Mb port in flow-based redirection.
- ☞ Do not use flow-based redirection to forward across VLANs.

7.16 Egress QoS

7.16.1 Introduction to Egress QoS

In traditional IP networks, all packets are treated the same way. All network devices handle them on a first-in, first-out basis and make a best effort to deliver them to the destination, but this does not guarantee performance such as reliability and transmission delay. As networks develop rapidly and new applications emerge, new demands are placed on the quality of service of IP networks. For

example, delay-sensitive services such as VoIP and video have strict requirements on packet transmission delay, and users cannot accept long delays (by contrast, E-mail and FTP services are not sensitive to delay). To support services with different requirements, such as voice, video, and data, the network must be able to distinguish different communications and provide appropriate service for each. A traditional best-effort IP network cannot identify and distinguish different kinds of communications, yet this ability is the prerequisite for providing differentiated services. Therefore, the best-effort service mode of traditional networks cannot meet application demands. QoS techniques were developed to solve this problem.

Egress PolicyMap is the QoS policy at the egress. It performs QoS control of packets in the egress direction and uses various techniques to provide better service for specified network communications. Egress PolicyMap includes a class-map and a policy-map: the class-map selects the packets to be processed, and the policy-map specifies the operations to be applied. Currently, not all devices support Egress QoS.

7.16.1.1 Egress QoS Terms

Egress QoS: QoS implemented at the egress of a port.

Inner_vid: The VLAN ID carried by the tag nearer the network layer header when two tags exist.

Outer_vid: The VLAN ID carried by the tag nearer the link layer header when two tags exist. When only one tag exists, it is considered the outer tag by default.

Outer_tpid: The protocol type in the link layer header that indicates the type of the outer tag.

7.16.1.2 Basic Egress QoS Model

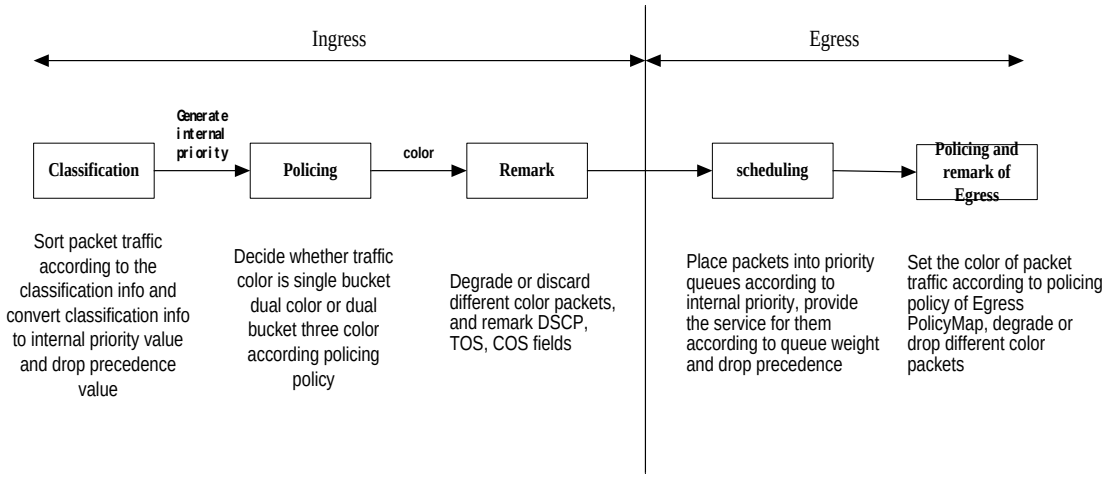


Figure 6-48

Based on the characteristics of upstream packets (including field values such as CoS and DSCP), egress policing and rewriting make the final QoS changes to a packet before it leaves the egress.

Policing applies different policing policies per flow and allocates bandwidth to classified flows. The bandwidth allocation policy can be dual bucket dual color or dual bucket three color. Different colors can be assigned to different flows, and packets can be discarded or passed accordingly; a rewriting action can be added for passed packets. The following flowchart describes Egress QoS in detail:

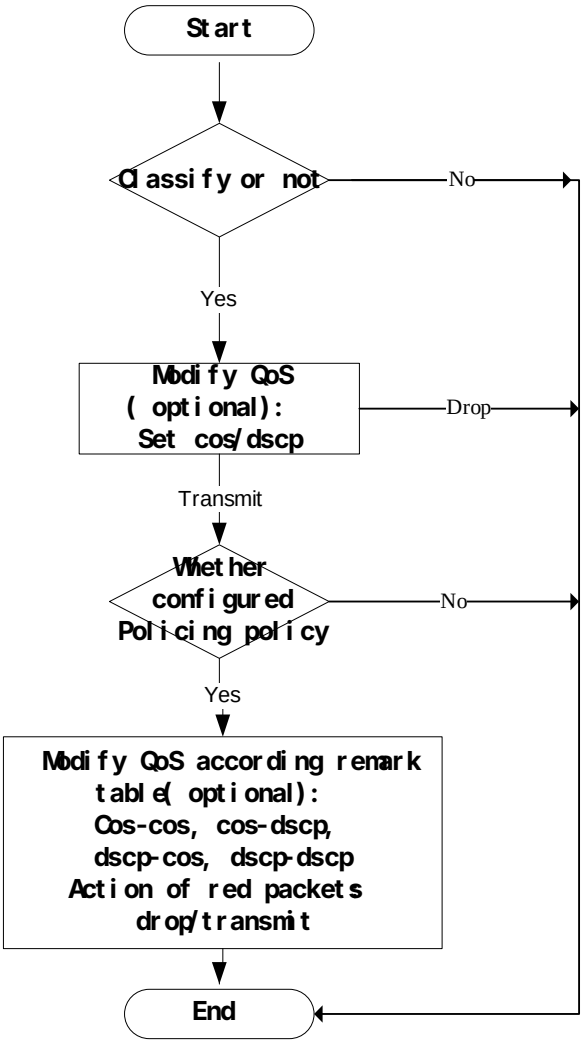


Figure 6-49

The actions that modify QoS attributes according to the egress remark table are described as follows:

cos-cos: modifies the CoS value of packets according to their CoS value and the CoS table for QoS remarking

cos-dscp: modifies the DSCP value of packets according to their CoS value and the CoS table for QoS remarking

dscp-cos: modifies the CoS value of packets according to their DSCP value and the DSCP table for QoS remarking

dscp-dscp: modifies the DSCP value of packets according to their DSCP value and the DSCP table for QoS remarking

7.16.2 Egress QoS Configuration

Egress QoS Configuration Task List:

Configure a class map

Set up classification rules based on ACL, CoS, VLAN ID, IPv4 Precedence, DSCP, and IPv6 DSCP to classify data streams. Different classes of data streams are processed with different policies.

Configure a policy map

After data streams are classified, create a policy map, associate it with a previously created class map, and enter policy class mode. Different policies (such as bandwidth limits and assigning new DSCP values) can then be applied to different data streams.

Apply Egress QoS to a port or VLAN

Configure the trust mode or bind policies to ports. A policy takes effect on a port only when it is bound to that port.

A policy can also be bound to a specific VLAN.

Set the Egress QoS remark mapping

If Egress QoS remark is used in a policy to modify QoS attributes, set the corresponding mapping. For the mapping to take effect on green packets, enable remarking of green packets, and the ingress must trust the corresponding QoS attribute (qos/dscp/exp).

1. Configure a class-map

Command	Explanation
Global Mode	
class-map <*class-map-name*> **no class-map <*class-map-name*>**	Creates a class map and enters class map mode. The no command deletes the specified class map.
match {access-group <*acl-index-or-name*> | ip dscp <*dscp-list*> | ip precedence <*ip-precedence-list*> | ipv6 dscp <*dscp-list*> | vlan <*vlan-list*> | cos <*cos-list*> | ipv6 access-group <*acl-index-or-name*>} **no match {access-group | ip dscp | ip precedence | ipv6 dscp | vlan | cos | ipv6 access-group}**	Configures the match criteria of the class map to classify data streams by ACL, CoS, VLAN ID, IPv4 Precedence, DSCP, or IPv6 DSCP priority. The no command deletes the specified match criterion.

2. Configure a policy-map

Command	Explanation
Global Mode	
policy-map <*policy-map-name*> **no policy-map <*policy-map-name*>**	Creates a policy map and enters policy map mode. The no command deletes the specified policy map.
class <*class-map-name*> [insert-before <*class-map-name*>] **no class <*class-map-name*>**	Associates a class map with the policy map and enters policy class map mode, so that different policies and new DSCP values can be applied to different data streams. The no command deletes the specified policy class map.
set {ip dscp <*new-dscp*> | ip precedence <*new-precedence*> | cos <*new-cos*> | c-vid <*new-c-vid*> | s-vid <*new-s-vid*> | s-tpid <*new-s-tpid*>} **no set {ip dscp | ip precedence | cos | c-vid | s-vid | s-tpid}**	Assigns a new DSCP, CoS, or IP Precedence value to the classified flow. The no command cancels the operation.
Single bucket mode: **policy <*bits_per_second*> <*normal_burst_bytes*> ({action ACTION} | exceed-action drop | transmit})** **Dual bucket mode:** **policy <*bits_per_second*> <*normal_burst_bytes*> [pir <*peak_rate_bps*>] | <*maximum_burst_bytes*> [{action ACTION | violate-action drop | transmit}]**	Configures a policy for the classified flow. The non-aggregate policy command supports three colors. Depending on the token bucket working mode (single rate single bucket, single rate dual bucket, or dual rate dual bucket), different actions are set for packets of different colors. The no command deletes the configuration. Single bucket mode is supported only on specific switches.

ACTION definition: **policied-cos-to-cos-transmit | policied-cos-to-dscp-transmit | policied-dscp-exp-to-cos-transmit | policied-dscp-exp-to-dscp-transmit** **no policy**	
accounting **no accounting**	Enables statistics for the classified flow. When this function is enabled in policy class map mode, statistics are collected for the flow of the policy class map. In single bucket mode, packets passing the policy can only be red or green; in the output, in-profile means green and out-profile means red. In dual bucket mode, packets have three colors: in-profile means green, and out-profile means red and yellow.

3. Apply policy to port or VLAN

Command	Explanation
Interface Mode	
service-policy output <*policy-map-name*> **no service-policy output {<*policy-map-name*>}**	Applies a policy map to the egress of the port. The no command deletes the specified policy map applied to the port, or all policy maps applied in the egress direction of the port.
Global Mode	
service-policy output <*policy-map-name*> vlan <*vlan-list*> **no service-policy output {<*policy-map-name*>} vlan <*vlan-list*>**	Applies a policy map to the egress of the VLAN. The no command deletes the specified policy map applied to the VLAN interface, or all policy maps applied in the egress direction of the VLAN interface.

4．Set Egress QoS remark mapping

Command	Explanation
Global Mode	
mls qos map {cos-cos | cos-dscp} {green | yellow | red} <value1> <value2>…<value8> **no mls qos map {cos-cos | cos-dscp} {green | yellow | red}**	Sets the egress CoS mapping. The no command restores the default configuration.
mls qos map {dscp-cos | dscp-dscp} {green | yellow | red} <dscp list> to <value> **no mls qos map {dscp-cos | dscp-dscp} {green | yellow | red}**	Sets the egress DSCP mapping; <dscp-list> specifies 1 to 8 DSCP values. The no command restores the default configuration.
mls qos egress green remark **no mls qos egress green remark**	Enables the Egress QoS remark mapping for green packets. The no command disables it for green packets.

5. Clear accounting data of the specific ports or VLANs

Command	Explanation
Admin Mode	
clear mls qos statistics [interface <interface-name> | vlan <vlan-id>]	Clears the accounting data of the policy maps on the specified port or VLAN. If no parameter is specified, clears the accounting data of all policy maps.

6. Show QoS configuration

Command	Explanation

Admin Mode	
show mls qos {interface [<*interface-id*>] [policy | queuing] | vlan <*vlan-id*>}	Displays the QoS configuration of the port.
show class-map [<*class-map-name*>]	Displays the QoS class map information.
show policy-map [<*policy-map-name*>]	Displays the QoS policy map information.
show mls qos maps {cos-cos | cos-dscp | dscp-cos | dscp-exp} {green | yellow | red |}	Displays the Egress QoS remark mappings.

7.16.3 Egress QoS Examples

Example1:

At the egress of port 1, change the CoS value to 4 for packets with DSCP value 0.

Create a class map:

```
switch(config)#class-map 1
switch(config-classmap-1)#match ip dscp 0
switch(config-classmap-1)#exit
```

Create a policy map:

```
switch(config)#policy-map 1
switch(config-policymap-1)#class 1
switch(config-policymap-1-class-1)#set cos 4
switch(config-policymap-1-class-1)#exit
switch(config-policymap-1)#exit
```

Bind the policy to the port:

```
switch(config)#in e 1/0/1
switch(config-if-ethernet1/0/1)#service-policy output 1
```

Example2:

At the egress of vlan10, change the CoS value to 4 for packets with IPv6 DSCP value 7.

Create a class map:

```
switch(config)#class-map 1
switch(config-classmap-1)#match ipv6 dscp 7
switch(config-classmap-1)#exit
```

Create a policy map:

```
switch(config)#policy-map 1
switch(config-policymap-1)#class 1
switch(config-policymap-1-class-1)#set cos 4
switch(config-policymap-1-class-1)#exit
switch(config-policymap-1)#exit
```

Bind the policy to the VLAN:

```
switch(config)#service-policy output 1 vlan 10
```

Example 3:

At the egress of port 1, rate-limit packets: set the bandwidth to 1 Mb/s, with a normal burst of 1 MB and a maximum burst of 4 MB; change DSCP value 1 to 10 for green packets and to 9 for yellow packets, and drop red packets.

Create a class map:

```
switch(config)#class-map c1
switch(config-classmap-c1)#match ip dscp 1
switch(config-classmap-c1)#exit
```

Create a policy map:

```
switch(config)#policy-map p1
switch(config-policymap-p1)#class c1
switch(config-policymap-p1-class-c1)#policy 1000 1000 4000 action policied-
dscp-exp-to-dscp-transmit violate-action drop
switch(config-policymap-p1-class-c1)#exit
switch(config-policymap-p1)#exit
```

Set the egress DSCP remark mapping:

```
switch(config)#mls qos map dscp-dscp green 1 to 10
switch(config)#mls qos map dscp-dscp yellow 1 to 9
```

Enable egress remarking for green packets:

```
switch(config)#mls qos egress green remark
```

Set the ingress to trust DSCP mode:

```
switch(config-if-port-range)#mls qos trust dscp
```

Bind the policy to the egress of port 1:

```
switch(config-if-ethernet1/0/1)#service-policy output p1
```

7.16.4 Egress QoS Troubleshooting Help

- Not all devices currently support Egress QoS; make sure that the current device supports this function.
- If the configured policy cannot be bound to a port or VLAN, check whether the match options in the classification table are supported by the current device.
- If the terminal reports insufficient resources, make sure that enough resources are available to apply the current policy.
- If a policy with match acl configured cannot be bound to a port or VLAN, make sure that the ACL contains permit rules.
- If modifying QoS attributes through Egress QoS remark does not take effect, make sure that the ingress trusts the corresponding QoS attribute.
- If the egress sets QoS attributes (set cos/ip dscp) for all packets and also uses Egress remark to modify the QoS attributes of packets by color, the former modification takes precedence.

7.17 Flexible QinQ

7.17.1 Introduction to Flexible QinQ

7.17.1.1 QinQ Technique

Dot1q-tunnel, also called QinQ (802.1Q-in-802.1Q), is an extension of 802.1Q. Its main idea is to encapsulate the customer VLAN tag (CVLAN tag) in the service provider VLAN tag (SPVLAN tag). Packets with two VLAN tags are transmitted through the ISP backbone network, providing a simple Layer 2 tunnel for users. QinQ is simple and easy to manage, requires only static configuration, and is especially suitable for small office networks or small metropolitan area networks that use Layer 3 switches as backbone equipment.

There are two kinds of QinQ: basic QinQ and flexible QinQ. Flexible QinQ has a higher priority than basic QinQ.

7.17.1.2 Basic QinQ

Basic QinQ is port-based. After QinQ is configured on a port, the device adds the default VLAN tag to every received packet, whether tagged or not. Basic QinQ is simple to use, but its VLAN tag setting is inflexible.

7.17.1.3 Flexible QinQ

Flexible QinQ is flow-based. It determines whether to add an outer tag, and which outer tag to add, by matching data flows. For example, flexible QinQ can be applied according to the user's VLAN tag, MAC address, IPv4/IPv6 address, IPv4/IPv6 protocol, application port number, and so on. It can thus encapsulate outer tags for packets and apply different schemes to different users or methods.

7.17.2 Flexible QinQ Configuration Task List

Flexible QinQ data flows are matched by QoS policy-map rules. The configuration task list is as follows:

1. Create a class map to classify different data flows

2. Create a flexible QinQ policy map, associate it with the class map, and set the corresponding operation

3. Bind the flexible QinQ policy map to a port

1. Configure class map

Command	Explanation
Global Mode	
class-map <*class-map-name*> **no class-map <*class-map-name*>**	Creates a class map and enters class map mode. The no command deletes the specified class map.
match {access-group <*acl-index-or-name*> | ip dscp <*dscp-list*>| ip precedence <*ip-precedence-list*>| ipv6 access-group <*acl-index-or-name*>| ipv6 dscp <*dscp-list*> | ipv6 flowlabel <flowlabel-list> | vlan <vlan-list> | cos <*cos-list*>} **no match {access-group | ip dscp | ip precedence|ipv6 access-group| ipv6 dscp | ipv6 flowlabel | vlan | cos}**	Sets the match criteria of the class map (classifying data flows by ACL, CoS, VLAN ID, IPv4 Precedence, DSCP, and so on). The no command deletes the specified match criterion.

2. Configure policy-map of flexible QinQ

Command	Explanation
Global Mode	
policy-map <*policy-map-name*> **no policy-map <*policy-map-name*>**	Creates a policy map and enters policy map mode. The no command deletes the specified policy map.
class <*class-map-name*> [insert-before <class-map-name>] **no class <*class-map-name*>**	Associates a class with the policy map after the policy map is created. In class mode, different policies or new DSCP values can be applied to different data flows. The no command deletes the specified class.
set s-vid < vid> **no set s-vid**	Sets the outer VLAN tag for the classified traffic. The no command cancels the operation.
add s-vid <vid> **no add s-vid <vid>**	Adds an outer VLAN tag to the classified traffic. The no command cancels the operation.

3. Bind flexible QinQ policy-map to port

Command	Explanation
Port Mode	
service-policy input<*policy-map-name*> **no service-policy input<*policy-map-name*>**	Applies a policy map to the port. The no command deletes the specified policy map applied to the port.
Global Mode	
service-policy input<*policy-map-name*> vlan<vid> **no service-policy input<*policy-map-name*> vlan <*vid*>**	Applies a policy map to the VLAN. The no command deletes the specified policy map applied to the VLAN.

4. Show flexible QinQ policy-map bound to port

Command	Explanation
Admin Mode	
show mls qos {interface [<interface-id>]	Displays the flexible QinQ configuration on the port.

7.17.3 Flexible QinQ Example

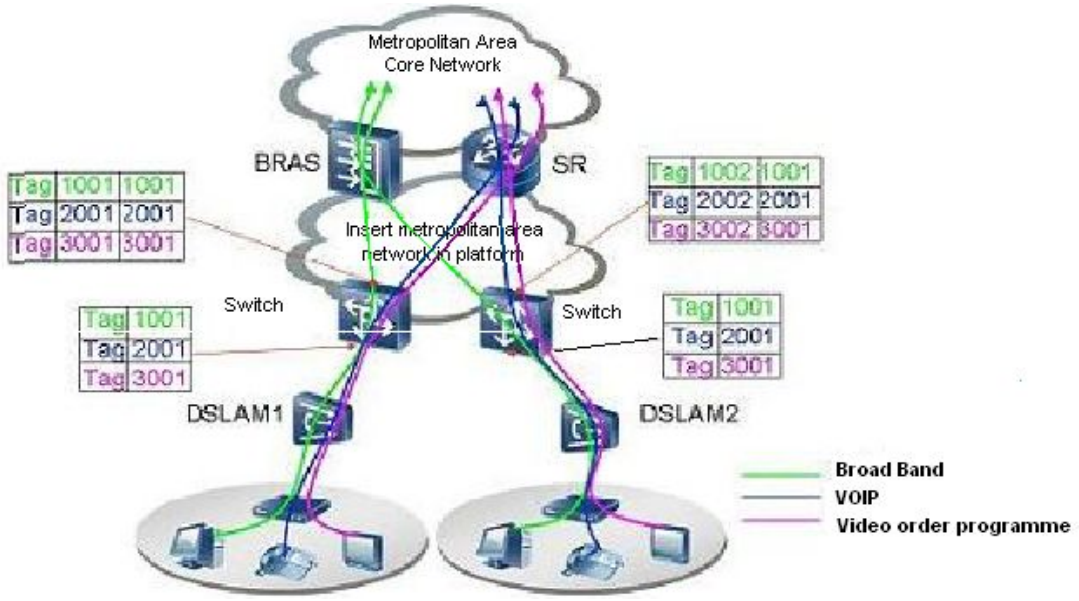


Figure 7-350 Flexible QinQ Application Topology

As shown in the figure, the first user is assigned three VLANs in DSLAM1, with tag values 1001, 2001, and 3001. VLAN1001 corresponds to the broadband network, VLAN2001 to VoIP, and VLAN3001 to VoD. After flexible QinQ is enabled on the downlink port, packets are given different outer tags according to the user's VLAN ID. Packets with tag 1001 are given outer tag 1001 directly (this tag is unique in the public network), enter Broad Band Network-VLAN1001, and are forwarded to the BRAS device. Packets with tag 2001 (or 3001) are given outer tag 2001 (or 3001) and are forwarded to the SR device according to the flow rules. The second user can be assigned different VLAN tags for different VLANs in DSLAM2. Note: the VLAN tags assigned to the second user may be the same as those of the first user, and these tagged packets are also given an outer tag. In the figure above, the outer tags of the second user differ from those of the first user, to distinguish DSLAM locations and ultimately locate users.

The configuration is as follows:

If the data flow from DSLAM1 enters downlink port 1 of the switch, configure as follows:

```
Switch(config)#class-map c1
Switch(config-classmap-c1)#match vlan 1001
Switch(config-classmap-c1)#exit
Switch(config)#class-map c2
Switch(config-classmap-c2)#match vlan 2001
Switch(config-classmap-c2)#exit
Switch(config)#class-map c3
Switch(config-classmap-c3)#match vlan 3001
Switch(config-classmap-c3)#exit
Switch(config)#policy-map p1
Switch(config-policymap-p1)#class c1
Switch(config-policymap-p1-class-c1)# set s-vid 1001
Switch(config-policymap-p1)#class c2
Switch(config-policymap-p1-class-c2)# set s-vid 2001
Switch(config-policymap-p1)#class c3
Switch(config-policymap-p1-class-c3)# set s-vid 3001
Switch(config-policymap-p1-class-c3)#exit
Switch(config-policymap-p1)#exit
Switch(config)#interface ethernet 1/0/1
Switch(config-if-ethernet1/0/1)#service-policy input p1
```

If the data flow from DSLAM2 enters downlink port 1 of the switch, configure as follows:

```
Switch(config)#class-map c1
Switch(config-classmap-c1)#match vlan 1001
Switch(config-classmap-c1)#exit
Switch(config)#class-map c2
Switch(config-classmap-c2)#match vlan 2001
Switch(config-classmap-c2)#exit
Switch(config)#class-map c3
Switch(config-classmap-c3)#match vlan 3001
Switch(config-classmap-c3)#exit
Switch(config)#policy-map p1
Switch(config-policymap-p1)#class c1
Switch(config-policymap-p1-class-c1)# set s-vid 1002
Switch(config-policymap-p1)#class c2
Switch(config-policymap-p1-class-c2)# set s-vid 2002
Switch(config-policymap-p1)#class c3
Switch(config-policymap-p1-class-c3)# set s-vid 3002
Switch(config-policymap-p1-class-c3)#exit
Switch(config-policymap-p1)#exit
Switch(config)#interface ethernet 1/0/1
Switch(config-if-ethernet1/0/1)# service-policy input p1
```

7.17.4 Flexible QinQ Troubleshooting

If a flexible QinQ policy cannot be bound to a port, check whether the problem is caused by the following:

- ☞ Make sure that flexible QinQ supports the configured class map and policy map.
- ☞ Make sure that the ACL includes a permit rule if the class map matches an ACL.
- ☞ Make sure that the switch has enough TCAM resources to apply the binding.
- ☞ The priority of flexible QinQ and VLAN ingress filtering for processing packets is: flexible QinQ > VLAN ingress filtering.

Chapter 8 Reliability Configuration

8.1 MSTP

8.1.1 Introduction to MSTP

MSTP (Multiple Spanning Tree Protocol) is a spanning-tree protocol based on STP and RSTP. It runs on all bridges of a bridged LAN. It calculates a common and internal spanning tree (CIST) for a bridged LAN that consists of bridges running MSTP, RSTP, and STP. It also calculates independent multiple spanning-tree instances (MSTIs) for each MST region (MSTP region). MSTP adopts RSTP for rapid convergence of the spanning tree and allows multiple VLANs to be mapped to the same spanning-tree instance, which is independent of other spanning-tree instances. MSTP provides multiple forwarding paths for data traffic and enables load balancing. Moreover, because multiple VLANs share the same MSTI, MSTP reduces the number of spanning-tree instances, which consumes fewer CPU resources and less bandwidth.

8.1.1.1 MSTP Region

Because multiple VLANs can be mapped to a single spanning-tree instance, the IEEE 802.1s committee introduced the MST concept. MST is used to associate specific VLANs with specific spanning-tree instances.

An MSTP region consists of one or more bridges with the same MCID (MST Configuration Identification) and the bridged LAN (a bridge in the MSTP region is the designated bridge of the LAN, and the bridges attached to the LAN do not run STP). All bridges in the same MSTP region have the same MCID.

The MCID consists of three attributes:

- Configuration Name: Consists of digits and letters
- Revision Level
- Configuration Digest: The mapping of VLANs to spanning-tree instances

Bridges with the same three attributes above are considered to be in the same MST region.

When MSTP calculates the CIST in a bridged LAN, an MSTP region is treated as a single bridge. See the figure below:

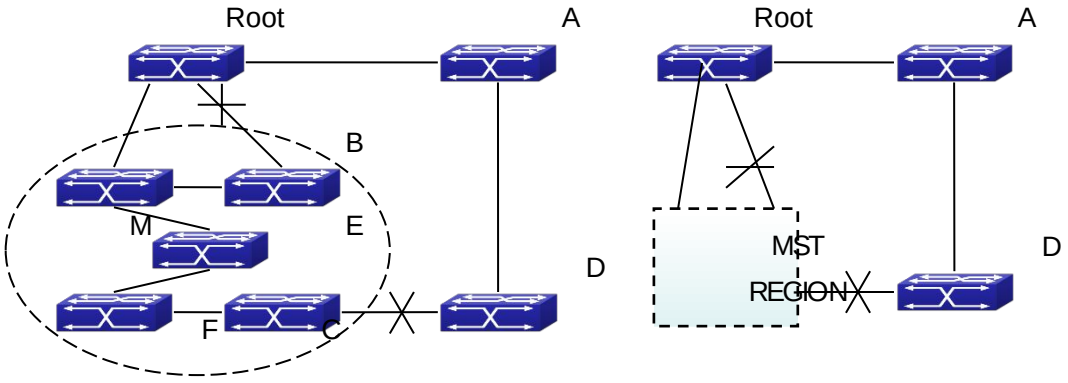


Figure 8-21 Example of CIST and MST Region

In the network above, if the bridges run STP or RSTP, one port between Bridge M and Bridge B must be blocked. However, if the bridges in the yellow area run MSTP and are configured in the same MST region, MSTP treats this region as a single bridge. Therefore, one port between Bridge B and Root is blocked, and one port on Bridge D is blocked.

8.1.1.1.1 Operations Within an MSTP Region

The IST connects all MSTP bridges in a region. When the IST converges, the root of the IST becomes the IST master, which is the switch within the region with the lowest bridge ID and path cost to the CST root. The IST master is also the CST root if there is only one region in the network. If the CST root is outside the region, one of the MSTP bridges at the region boundary is selected as the IST master.

When an MSTP bridge initializes, it sends BPDUs claiming itself as the root of the CST and the IST master, with the path costs to both the CST root and the IST master set to zero. The bridge also initializes all of its MST instances and claims to be the root for all of them. If the bridge receives MST root information on a port that is superior (lower bridge ID, lower path cost, and so on) to the information currently stored for that port, it relinquishes its claim as the IST master.

Within an MST region, the IST is the only spanning-tree instance that sends and receives BPDUs. Because the MST BPDU carries information for all instances, the number of BPDUs that a switch must process to support multiple spanning-tree instances is significantly reduced.

All MST instances within the same region share the same protocol timers, but each MST instance has its own topology parameters, such as root switch ID and root path cost.

8.1.1.1.2 Operations Between MST Regions

If there are multiple regions or legacy 802.1D bridges in the network, MSTP establishes and maintains the CST, which includes all MST regions and all legacy STP bridges in the network. The MST instances combine with the IST at the region boundary to form the CST.

An MSTI is valid only within its MST region and is independent of MSTIs in other MST regions. Bridges in an MST region receive MST BPDUs from other regions through boundary ports. They process only CIST-related information and discard MSTI information.

8.1.1.2 Port Roles

An MSTP bridge assigns a port role to each port that runs MSTP.

- ☞ CIST port roles: Root Port, Designated Port, Alternate Port, and Backup Port.
- ☞ In addition to these roles, each MSTI port has one new role: Master Port.

The port roles in the CIST (Root Port, Designated Port, Alternate Port, and Backup Port) are defined in the same way as in RSTP.

8.1.1.3 MSTP Load Balancing

In an MSTP region, VLANs can be mapped to different instances, which can form different topologies. Each instance is independent of the others, and each instance can have its own attributes, such as bridge priority and port cost. Consequently, VLANs in different instances have their own paths, and VLAN traffic is load-balanced.

8.1.2 MSTP Configuration Task List

MSTP configuration task list:

1. Enable MSTP and set the running mode
2. Configure instance parameters
3. Configure MSTP region parameters
4. Configure MSTP timer parameters
5. Configure the fast migration feature for MSTP
6. Configure the port packet format
7. Configure the spanning-tree attributes of a port

8. Configure the snooping attribute of the authentication key

9. Configure the flush mode for topology changes

1. Enable MSTP and set the running mode

Command	Explanation
Global Mode and Port Mode	
spanning-tree **no spanning-tree**	Enables or disables MSTP.
Global Mode	
spanning-tree mode {mstp|stp|rstp} **no spanning-tree mode**	Sets the MSTP running mode.
Port Mode	
spanning-tree mcheck	Forces the port to migrate to MSTP mode.

2. Configure instance parameters

Command	Explanation
Global Mode	
spanning-tree mst <*instance-id*> priority <*bridge-priority*> **no spanning-tree mst <*instance-id*> priority**	Sets the bridge priority for the specified instance.
spanning-tree priority <*bridge-priority*> **no spanning-tree priority**	Configures the spanning-tree priority of the switch.
Port Mode	
spanning-tree mst <*instance-id*> cost <*cost*> **no spanning-tree mst <*instance-id*> cost**	Sets the port path cost for the specified instance.
spanning-tree mst <*instance-id*> port-priority <*port-priority*> **no spanning-tree mst <*instance-id*> port-priority**	Sets the port priority for the specified instance.
spanning-tree mst <*instance-id*> rootguard **no spanning-tree mst <*instance-id*> rootguard**	Configures whether root guard runs on the current port in the specified instance; a port with root guard cannot become a root port.
spanning-tree rootguard **no spanning-tree rootguard**	Configures whether root guard runs on the current port in instance 0; a port with root guard cannot become a root port.
spanning-tree [mst <*instance-id*>] loopguard **no spanning-tree [mst <*instance-id*>] loopguard**	Enables loop guard on the specified instance. The no command disables this function.

3. Configure MSTP region parameters

Command	Explanation
Global Mode	
spanning-tree mst configuration **no spanning-tree mst configuration**	Enters MSTP Region Mode. The no command restores the default setting.
MSTP Region Mode	
show	Displays information about the current running system.

instance <*instance-id*> vlan <*vlan-list*> **no instance <*instance-id*> [vlan <*vlan-list*>]**	Creates an instance and sets the mapping between VLANs and the instance.
name <*name*> **no name**	Sets the MSTP region name.
revision-level <*level*> **no revision-level**	Sets the MSTP region revision level.
abort	Exits MSTP Region Mode and returns to Global Mode without saving the MSTP region configuration.
exit	Exits MSTP Region Mode and returns to Global Mode, saving the MSTP region configuration.
no	Cancels a command or restores its initial value.

4. Configure MSTP timer parameters

Command	Explanation
Global Mode	
spanning-tree forward-time <*time*> **no spanning-tree forward-time**	Sets the forward delay time of the switch.
spanning-tree hello-time <*time*> **no spanning-tree hello-time**	Sets the hello time for sending BPDU messages.
spanning-tree maxage <*time*> **no spanning-tree maxage**	Sets the aging time for BPDU messages.
spanning-tree max-hop <*hop-count*> **no spanning-tree max-hop**	Sets the maximum number of hops for BPDU messages in the MSTP region.

5. Configure the fast migration feature for MSTP

Command	Explanation
Port Mode	
spanning-tree link-type p2p {auto|force-true| force-false} **no spanning-tree link-type**	Sets the port link type.
spanning-tree portfast [bpdufilter| bpduguard] [recovery <30-3600>] **no spanning-tree portfast**	Sets or cancels the port as a boundary port. With bpdufilter, received BPDUs are discarded; with bpduguard, receiving a BPDU disables the port; with no parameter, the port becomes a non-boundary port when it receives a BPDU.

6. Configure the MSTP packet format

Command	Explanation
Port Mode	
spanning-tree format standard **spanning-tree format privacy** **spanning-tree format auto** **no spanning-tree format**	Configures the format of spanning-tree packets on the port: standard is the IEEE-defined format, privacy is a private format compatible with other vendors' devices, and auto means the format is determined from received packets.

7. Configure the spanning-tree attributes of a port

Command	Explanation
Port Mode	

spanning-tree cost **no spanning-tree cost**	Sets the port path cost.
spanning-tree port-priority **no spanning-tree port-priority**	Sets the port priority.
spanning-tree rootguard **no spanning-tree rootguard**	Configures root guard on the port so that the port cannot become a root port.
Global Mode	
spanning-tree transmit-hold-count <*tx-hold-count-value*> **no spanning-tree transmit-hold-count**	Sets the maximum transmit hold count of the port.
spanning-tree cost-format {dot1d | dot1t}	Sets the port cost format to dot1d or dot1t.

8. Configure the snooping attribute of authentication key

Command	Explanation
Port Mode	
spanning-tree digest-snooping **no spanning-tree digest-snooping**	Sets the port to use the authentication string of the partner port. The no command restores the use of the locally generated string.

9. Configure the flush mode for topology changes

Command	Explanation
Global Mode	
spanning-tree tcflush {enable| disable| protect} **no spanning-tree tcflush**	enable: Spanning tree flushes entries each time the topology changes. disable: Spanning tree does not flush entries when the topology changes. protect: Spanning tree flushes entries no more than once every ten seconds. The no command restores the default setting (flush each time the topology changes).
Port Mode	
spanning-tree tcflush {enable| disable| protect} **no spanning-tree tcflush**	Configures the flush mode of the port. The no command restores the use of the globally configured flush mode.

8.1.3 MSTP Example

The following is a typical MSTP application example:

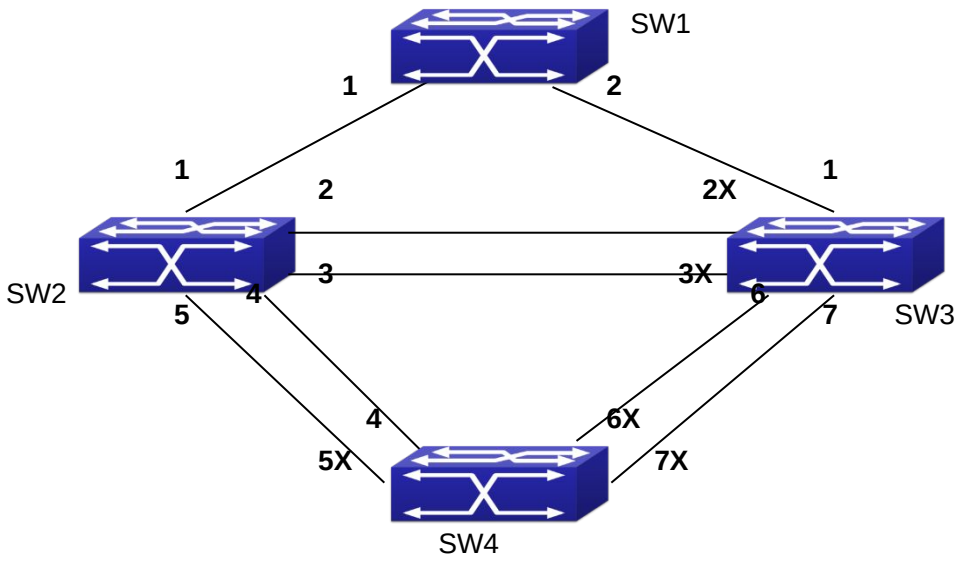


Figure 8-22 Typical MSTP Application Scenario

The connections among the switches are shown in the figure above. All switches run in MSTP mode by default, and their bridge priority, port priority, and port route cost are all at default (equal) values. The default switch configuration is listed below:

Bridge Name		SW1	SW2	SW3	SW4
Bridge MAC Address		…00-00-01	…00-00-02	…00-00-03	…00-00-04
Bridge Priority		32768	32768	32768	32768
Port Priority	port 1	128	128	128	
	port 2	128	128	128	
	port 3		128	128	
	port 4		128		128
	port 5		128		128
	port 6			128	128
	port 7			128	128
Route Cost	port 1	200000	200000	200000	
	port 2	200000	200000	200000	
	port 3		200000	200000	
	port 4		200000		200000
	port 5		200000		200000
	port 6			200000	200000
	port 7			200000	200000

By default, MSTP establishes a tree topology (blue lines) rooted at Switch1. Ports marked with "x" are in the discarding state, and the other ports are in the forwarding state.

Configuration steps:

Step 1: Configure port-to-VLAN mapping:

- ☞ Create VLANs 20, 30, 40, and 50 on Switch2, Switch3, and Switch4.

- ☞ Set ports 1-7 as trunk ports on Switch2, Switch3, and Switch4.

Step 2: Place Switch2, Switch3, and Switch4 in the same MSTP region:

- ☞ Set the region name of Switch2, Switch3, and Switch4 to mstp.
- ☞ Map VLAN 20 and VLAN 30 on Switch2, Switch3, and Switch4 to Instance 3; map VLAN 40 and VLAN 50 on Switch2, Switch3, and Switch4 to Instance 4.

Step 3: Set Switch3 as the root bridge of Instance 3 and Switch4 as the root bridge of Instance 4:

- ☞ Set the bridge priority of Instance 3 on Switch3 to 0.
- ☞ Set the bridge priority of Instance 4 on Switch4 to 0.

The detailed configuration is listed below:

```
Switch2:
Switch2(config)#vlan 20
Switch2(Config-Vlan20)#exit
Switch2(config)#vlan 30
Switch2(Config-Vlan30)#exit
Switch2(config)#vlan 40
Switch2(Config-Vlan40)#exit
Switch2(config)#vlan 50
Switch2(Config-Vlan50)#exit
Switch2(config)#spanning-tree mst configuration
Switch2(Config-Mstp-Region)#name mstp
Switch2(Config-Mstp-Region)#instance 3 vlan 20;30
Switch2(Config-Mstp-Region)#instance 4 vlan 40;50
Switch2(Config-Mstp-Region)#exit
Switch2(config)#interface e1/0/1-7
Switch2(Config-Port-Range)#switchport mode trunk
Switch2(Config-Port-Range)#exit
Switch2(config)#spanning-tree
Switch3:
Switch3(config)#vlan 20
Switch3(Config-Vlan20)#exit
Switch3(config)#vlan 30
Switch3(Config-Vlan30)#exit
Switch3(config)#vlan 40
Switch3(Config-Vlan40)#exit
Switch3(config)#vlan 50
Switch3(Config-Vlan50)#exit
Switch3(config)#spanning-tree mst configuration
Switch3(Config-Mstp-Region)#name mstp
Switch3(Config-Mstp-Region)#instance 3 vlan 20;30
Switch3(Config-Mstp-Region)#instance 4 vlan 40;50
Switch3(Config-Mstp-Region)#exit
Switch3(config)#interface e1/0/1-7
Switch3(Config-Port-Range)#switchport mode trunk
Switch3(Config-Port-Range)#exit
Switch3(config)#spanning-tree
Switch3(config)#spanning-tree mst 3 priority 0
Switch4:
Switch4(config)#vlan 20
Switch4(Config-Vlan20)#exit
Switch4(config)#vlan 30
Switch4(Config-Vlan30)#exit
Switch4(config)#vlan 40
Switch4(Config-Vlan40)#exit
Switch4(config)#vlan 50
Switch4(Config-Vlan50)#exit
Switch4(config)#spanning-tree mst configuration
Switch4(Config-Mstp-Region)#name mstp
Switch4(Config-Mstp-Region)#instance 3 vlan 20;30
```

```
Switch4(Config-Mstp-Region)#instance 4 vlan 40;50
Switch4(Config-Mstp-Region)#exit
Switch4(config)#interface e1/0/1-7
Switch4(Config-Port-Range)#switchport mode trunk
Switch4(Config-Port-Range)#exit
Switch4(config)#spanning-tree
Switch4(config)#spanning-tree mst 4 priority 0
```

After the above configuration, Switch1 is the root bridge of instance 0 for the entire network. In the MSTP region to which Switch2, Switch3, and Switch4 belong, Switch2 is the regional root of instance 0, Switch3 is the regional root of instance 3, and Switch4 is the regional root of instance 4. Traffic of VLAN 20 and VLAN 30 is sent through the topology of instance 3, traffic of VLAN 40 and VLAN 50 is sent through the topology of instance 4, and traffic of other VLANs is sent through the topology of instance 0. Port 1 on Switch2 is the master port of instance 3 and instance 4.

The MSTP calculation generates three topologies: instance 0, instance 3, and instance 4 (marked with blue lines). Ports marked with "x" are in the discarding state, and the other ports are in the forwarding state. Because instance 3 and instance 4 are valid only within the MSTP region, the following figures show only the topology of the MSTP region.

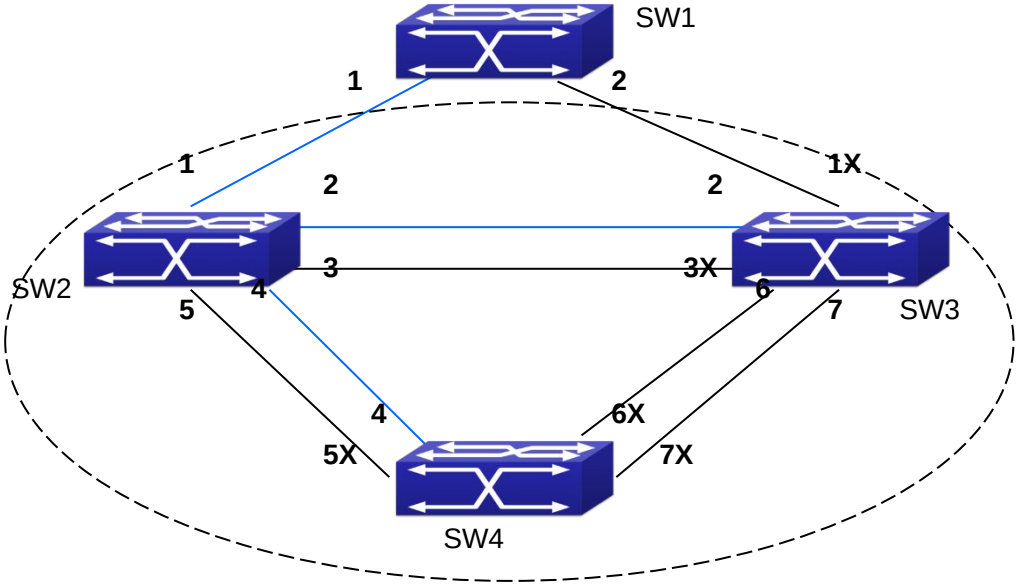


Figure 8-3 Topology of Instance 0 after MSTP Calculation

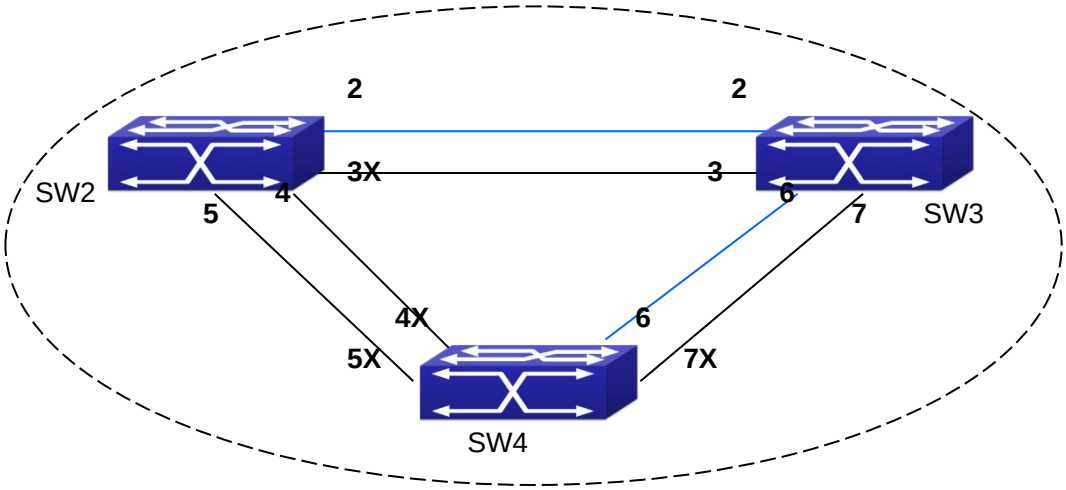


Figure 8-4 Topology of Instance 3 after MSTP Calculation

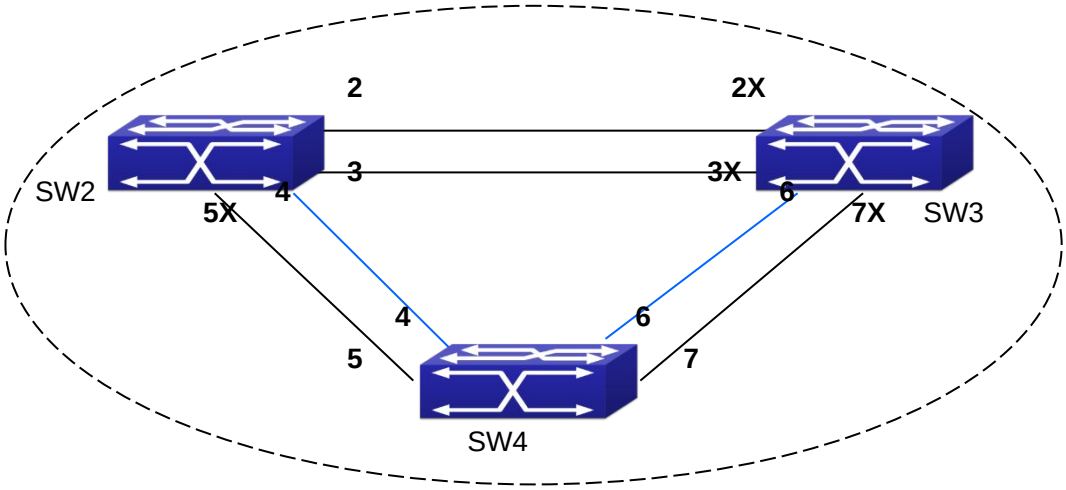


Figure 8-5 Topology of Instance 4 after MSTP Calculation

8.1.4 MSTP Troubleshooting

- ☞ To run MSTP on a switch port, MSTP must be enabled globally. If MSTP is not enabled globally, it cannot be enabled on the port.
- ☞ MSTP parameters interact with each other, so they must meet the following conditions; otherwise, MSTP may not work correctly:

2×(Bridge_Forward_Delay -1.0 seconds) >= Bridge_Max_Age

Bridge_Max_Age >= 2 ×(Bridge_Hello_Time + 1.0 seconds)

- ☞ Before modifying MSTP parameters, make sure you understand the resulting topology changes. Global configuration applies to the bridge; other configuration applies to individual instances.

8.2 VRRP

8.2.1 Introduction to VRRP

VRRP (Virtual Router Redundancy Protocol) is a fault-tolerant protocol designed to improve connection reliability between routers (or Layer 3 Ethernet switches) and external devices. It was developed by the IETF for local area networks (LANs) with multicast/broadcast capability, such as Ethernet, and is widely used.

All hosts in a LAN generally have a default route to a specified default gateway, and any packet destined for an address outside the local segment is sent to the default gateway through this default route. This allows the hosts in the LAN to communicate with external networks. However, if the link between the router serving as the default gateway and the external networks fails, all hosts using that gateway as the default next hop cannot communicate with external networks.

VRRP was developed to solve this problem. VRRP runs on multiple routers in a LAN and simulates a "virtual" router (also called a "standby cluster") from the multiple routers. A standby cluster has one active router (the "Master") and one or more backup routers (the "Backups"). The work of the virtual router is actually performed by the active router, while the backup routers serve as backups for the active router.

The virtual router has its own "virtual" IP address (which can be identical to the IP address of a router in the standby cluster), and the routers in the standby cluster also have their own IP addresses. Because VRRP runs only on routers or Ethernet switches, the standby cluster is transparent to the hosts in the segment. The hosts see only the IP address of the virtual router, not the actual IP addresses of the Master and Backups, and all hosts use the IP address of the virtual router as their default gateway. Therefore, hosts in the LAN communicate with other networks through the virtual

router, but in fact through the Master. If the Master of the standby cluster fails, a Backup takes over its tasks and becomes the Master to serve all hosts in the LAN, so communication between LAN hosts and external networks is not interrupted.

In summary, in a VRRP standby cluster, one router or Ethernet switch always serves as the active router (Master), while the rest of the standby cluster act as backup routers (Backups; there can be several) and continuously monitor the Master. If the Master fails, the Backups elect a new Master to take over the work and continue serving the hosts in the segment. Because the election and takeover are brief and smooth, hosts in the segment can continue to use the virtual router normally without communication interruption.

8.2.2 VRRP Configuration Task List

Configuration task list:

1. Create or remove the virtual router (required)
2. Configure the VRRP virtual IP address and interface (required)
3. Activate or deactivate the virtual router (required)
4. Configure VRRP sub-parameters (optional)
   （1） Configure the preemptive mode for VRRP
   （2） Configure VRRP priority
   （3） Configure VRRP timer intervals
   （4） Configure VRRP interface monitor
   （5） Configure the status of the VRRP session monitor port
   （6） Configure VRRP authentication

1. Create or remove the virtual router

Command	Explanation
Global Mode	
router vrrp <*vrid*> **no router vrrp <*vrid*>**	Creates or removes the virtual router.

2. Configure the VRRP virtual IP address and interface

Command	Explanation
VRRP Configuration Mode	
virtual-ip <*ip*> **no virtual-ip**	Configures the VRRP virtual IP address. The **no virtual-ip** command removes the virtual IP address.
interface {IFNAME | Vlan <*ID*>} **no interface**	Configures the VRRP interface. The **no interface** command removes the interface.

3. Activate or deactivate the virtual router

Command	Explanation
VRRP Configuration Mode	
enable	Activates the virtual router.
disable	Deactivates the virtual router.

4. Configure VRRP sub-parameters

(1) Configure the preemptive mode for VRRP

Command	Explanation
VRRP Configuration Mode	
preempt-mode {true| false}	Configures the preemptive mode for VRRP.

(2) Configure VRRP priority

Command	Explanation
VRRP Configuration Mode	
priority <*priority*>	Configures VRRP priority.

(3) Configure VRRP timer intervals

Command	Explanation
VRRP Configuration Mode	
advertisement-interval <*time*>	Configures the VRRP advertisement timer value (in seconds).

(4) Configure VRRP interface monitor

Command	Explanation
VRRP Configuration Mode	
circuit-failover {IFNAME | Vlan <*ID*> } <*value_reduced*> **no circuit-failover**	Configures VRRP interface monitoring. The **no circuit-failover** command removes monitoring of the interface.

(5) Configure the status of the VRRP session monitor port

Command	Explanation
VRRP Configuration Mode	
vrrp track interface {ethernet IFNAME| IFNAME} priority <priority_value> **no vrrp track interface {ethernet IFNAME|IFNAME}**	Configures the status of the VRRP session monitor port.

(6) Configure VRRP authentication

Command	Explanation
VRRP Configuration Mode	
authentication string {0|7} <*string*> **no authentication**	Configures VRRP authentication information. The **no authentication** command restores the default settings.

8.2.3 VRRP Typical Examples

As shown in the figure below, SwitchA and SwitchB are Layer 3 Ethernet switches in the same group that provide redundancy for each other.

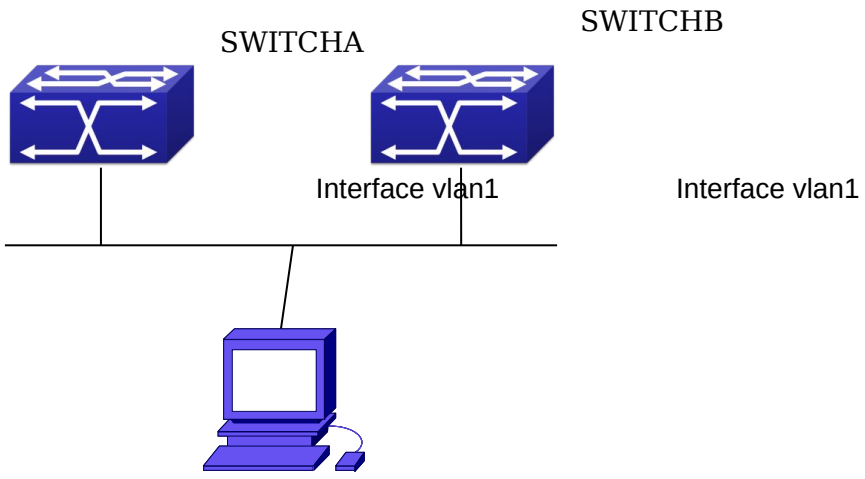


Figure 8-23 VRRP Network Topology

Configuration of SwitchA:

```
SwitchA(config)#interface vlan 1
SwitchA (Config-if-Vlan1)# ip address 10.1.1.1 255.255.255.0
```

```
SwitchA (config)#router vrrp 1
SwitchA(Config-Router-Vrrp)# virtual-ip 10.1.1.5
SwitchA(Config-Router-Vrrp)# interface vlan 1
SwitchA(Config-Router-Vrrp)# enable
```

Configuration of SwitchB:

```
SwitchB(config)#interface vlan 1
SwitchB (Config-if-Vlan1)# ip address 10.1.1.7 255.255.255.0
SwitchB(config)#router vrrp 1
SwitchB (Config-Router-Vrrp)# virtual-ip 10.1.1.5
SwitchB(Config-Router-Vrrp)# interface vlan 1
SwitchB(Config-Router-Vrrp)# enable
```

8.2.4 VRRP Troubleshooting

VRRP may fail to run properly due to causes such as physical connection failures or incorrect configuration. Check the following:

- ☞ The physical connection is in good condition.
- ☞ All interfaces and link protocols are in the UP state (use the **show interface** command).
- ☞ VRRP is enabled on the interface, and the authentication mode is the same on all routers (or Layer 3 Ethernet switches) in the same standby cluster.
- ☞ The timer values are the same on all routers (or Layer 3 Ethernet switches) in the same standby cluster.
- ☞ The virtual IP address is in the same network segment as the actual IP address of the interface.
- ☞ If the problem remains unsolved, use **debug vrrp** and other debugging commands, record the debug messages for 3 minutes, and send them to EnGenius technical support.

8.3 MRPP

8.3.1 Introduction to MRPP

MRPP (Multi-layer Ring Protection Protocol) is a link-layer protocol for Ethernet ring protection. It prevents broadcast storms caused by data loops on an Ethernet ring and restores communication among all nodes on the ring when a link on the ring breaks. MRPP is an extension of EAPS (Ethernet Automatic Protection Switching).

MRPP is similar to STP in function. Compared with STP, MRPP has the following characteristics:

<1> MRPP is designed specifically for Ethernet ring topologies.

<2> Fast convergence: less than 1 s; ideally 100-50 ms.

8.3.1.1 Concepts

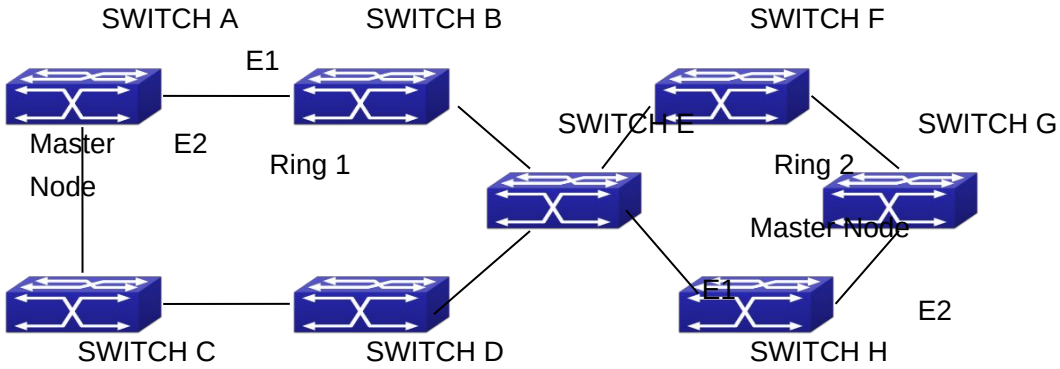


Figure 8-7 MRPP Diagram

1. Control VLAN

The control VLAN is a virtual VLAN used only to identify MRPP protocol packets transferred on the link. To avoid confusion with other configured VLANs, do not configure the control VLAN ID to be the same as any other configured VLAN ID. Different MRPP rings must use different control VLAN IDs.

2. Ethernet Ring (MRPP Ring)

An Ethernet network with a ring topology.

Each MRPP ring has two states:

Health state: All physical links of the ring network are connected.

Break state: One or more physical links in the ring network are broken.

3. Nodes

Each switch on the Ethernet ring is called a node. There are two node types:

Primary node: Each ring has one primary node, which is the main node for detecting and protecting the ring.

Transfer node: All nodes on a ring other than the primary node are transfer nodes.

The node role is determined by user configuration. As shown in Figure 8-7, Switch A is the primary node of Ring 1, and Switch B, Switch C, Switch D, and Switch E are transfer nodes of Ring 1.

4. Primary port and secondary port

The primary node and each transfer node have two ports connected to the Ethernet ring: one is the primary port and the other is the secondary port. The port role is determined by user configuration.

Primary port and secondary port of the primary node:

The primary port of the primary node sends ring health check packets (Hello), and the secondary port receives the Hello packets sent from the primary node. When the Ethernet ring is in the health state, the secondary port of the primary node logically blocks other data and allows only MRPP packets to pass. When the Ethernet ring is in the break state, the secondary port of the primary node leaves the blocked state and forwards data packets.

The primary port and secondary port of a transfer node do not differ in function.

The port role is determined by user configuration. As shown in Figure 8-7, E1 of Switch A is the primary port and E2 is the secondary port.

5. Timer

The primary node uses two timers to send and receive MRPP protocol packets: the Hello timer and the Fail timer.

Hello timer: Defines the interval at which the primary port of the primary node sends health check packets.

Fail timer: Defines the timeout for receiving health check packets on the primary node. The Fail timer value must be greater than or equal to 3 times the Hello timer value.

8.3.1.2 MRPP Protocol Packet Types

Packet Type	Explanation
Hello packet (health check packet)	Sent from the primary port of the primary node to check the ring. If the secondary port of the primary node receives the Hello packet within the configured timeout, the ring is normal.
LINK-DOWN (link down event packet)	When a transfer node detects a Down event on a port, it immediately sends a LINK-DOWN packet to the primary node to report the ring failure.
LINK-DOWN-FLUSH_FDB packet	After the primary node detects a ring failure or receives a LINK-DOWN packet, it unblocks the secondary port and sends this packet through both ports to instruct each transfer node to refresh its MAC address table.
LINK-UP-FLUSH_FDB packet	After the primary node detects that the ring has recovered from a failure, it sends this packet from the primary port to instruct each transfer node to refresh its MAC address table.

8.3.1.3 MRPP Protocol Operation

1. Link Down Alarm Mechanism

When a transfer node detects that one of its MRPP ring ports is down, it immediately sends a LINK-DOWN packet to the primary node. On receiving the LINK-DOWN packet, the primary node immediately unblocks its secondary port and sends a LINK-DOWN-FLUSH-FDB packet to instruct all transfer nodes to refresh their MAC address forwarding tables.

2. Polling Mechanism

The primary port of the primary node periodically sends Hello packets to its neighbors according to the configured Hello timer.

If the ring is healthy, the secondary port of the primary node receives the health check packets, and the primary node keeps the secondary port blocked.

If the ring is broken, the secondary port of the primary node does not receive the health check packets before the timer expires. The primary node then unblocks the secondary port and sends a LINK-DOWN-FLUSH_FDB packet to instruct all transfer nodes to refresh their MAC address forwarding tables.

3. Ring Recovery

After a ring failure, if the secondary port of the primary node receives a Hello packet sent from the primary node, the ring has recovered. The primary node then blocks its secondary port and sends a LINK-UP-Flush-FDB packet to its neighbors.

After an MRPP ring port on a transfer node comes back up, the primary node may detect the ring recovery only after some time. During this period, a temporary loop may form for normal data VLANs and cause a broadcast storm. To avoid a temporary loop, when a transfer node detects that its ring port has come up, it immediately blocks the port temporarily (allowing only control VLAN packets to pass), and unblocks the port only after it receives the LINK-UP-FLUSH-FDB packet from the primary node.

8.3.2 MRPP Configuration Task List

1) Enable MRPP globally

2) Configure MRPP ring

3) Configure the MRPP query time

4) Configure the compatibility mode

5) Display and debug MRPP information

1) Enable MRPP globally

Command	Explanation
Global Mode	
mrpp enable **no mrpp enable**	Enables or disables MRPP globally.

2) Configure MRPP ring

Command	Explanation
Global Mode	
mrpp ring <*ring-id*> **no mrpp ring <*ring-id*>**	Creates an MRPP ring. The no command deletes the MRPP ring and its configuration.
MRPP Ring Mode	
control-vlan <*vid*> **no control-vlan**	Configures the control VLAN ID. The no command deletes the configured control VLAN ID.
node-mode {master | transit}	Configures the node type of the MRPP ring (primary node or transfer node).
hello-timer < *timer*> **no hello-timer**	Configures the timer for sending Hello packets from the primary node of the MRPP ring. The no command restores the default timer value.
fail-timer <*timer*> **no fail-timer**	Configures the timeout timer for Hello packets sent from the primary node of the MRPP ring. The no command restores the default timer value.
enable **no enable**	Enables the MRPP ring. The no command disables the MRPP ring.
Port Mode	
mrpp ring <*ring-id*> primary-port **no mrpp ring <*ring-id*> primary-port**	Specifies the primary port of the MRPP ring.
mrpp ring <*ring-id*> secondary-port **no mrpp ring <*ring-id*> secondary-port**	Specifies the secondary port of the MRPP ring.

3) Configure the MRPP query time

Command	Explanation
Global Mode	
mrpp poll-time <20-2000>	Configures the MRPP query interval.

4) Configure the compatibility mode

Command	Explanation
Global Mode	
mrpp errp compatible **no mrpp errp compatible**	Enables ERRP compatibility mode. The no command disables the compatibility mode.

mrpp eaps compatible **no mrpp eaps compatible**	Enables EAPS compatibility mode. The no command disables the compatibility mode.
errp domain <*domain-id*> **no errp domain <*domain-id*>**	Creates an ERRP domain. The no command deletes the configured ERRP domain.

5) Display and debug MRPP information

Command	Explanation
Admin Mode	
debug mrpp **no debug mrpp**	Enables MRPP module debugging. The no command disables MRPP debug output.
show mrpp {<*ring-id*>}	Displays MRPP ring configuration information.
show mrpp statistics {<*ring-id*>}	Displays statistics of data packets received on the MRPP ring.
clear mrpp statistics {<*ring-id*>}	Clears statistics of data packets received on the MRPP ring.

8.3.3 MRPP Typical Scenario

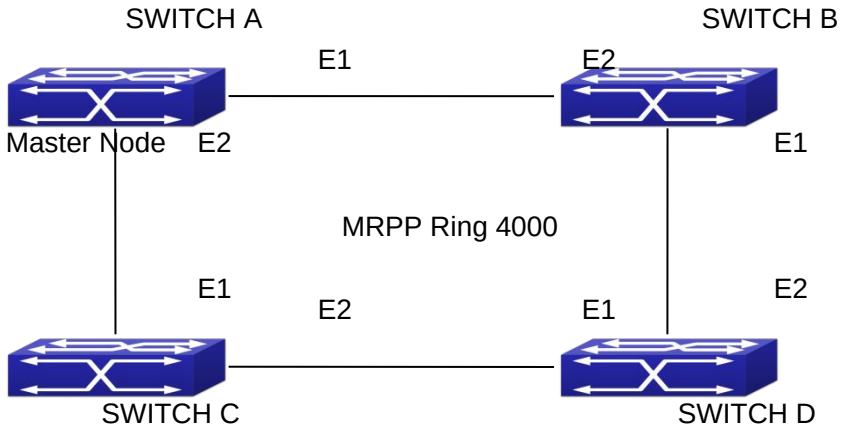


Figure 8-8 MRPP Typical Configuration Scenario

The topology above is common in MRPP deployments. Multiple switches form a single MRPP ring: all switches are configured with only MRPP ring 4000.

In this configuration, SWITCH A is the primary node of MRPP ring 4000, with E1/0/1 as the primary port and E1/0/2 as the secondary port. The other switches are transfer nodes of the MRPP ring, each with its own primary port and secondary port configured.

To avoid a loop, temporarily disable one of the ports of the primary node while enabling MRPP on each node of the ring, and re-enable the port after all nodes are configured.

Before disabling an MRPP ring, make sure that the ring is broken (no loop exists).

SWITCH A configuration task sequence:

```
Switch(Config)#mrpp enable
Switch(Config)#mrpp ring 4000
Switch(mrpp-ring-4000)#control-vlan 4000
Switch(mrpp-ring-4000)#fail-timer 18
Switch(mrpp-ring-4000)#hello-timer 5
Switch(mrpp-ring-4000)#node-mode master
Switch(mrpp-ring-4000)#enable
Switch(mrpp-ring-4000)#exit
Switch(Config)#interface ethernet 1/0/1
Switch(config-If-Ethernet1/0/1)#mrpp ring 4000 primary-port
```

```
Switch(config-If-Ethernet1/0/1)#interface ethernet 1/0/2
Switch(config-If-Ethernet1/0/2)#mrpp ring 4000 secondary-port
Switch(config-If-Ethernet1/0/2)#exit
Switch(Config)#
```

SWITCH B configuration task sequence:

```
Switch(Config)#mrpp enable
Switch(Config)#mrpp ring 4000
Switch(mrpp-ring-4000)#control-vlan 4000
Switch(mrpp-ring-4000)#enable
Switch(mrpp-ring-4000)#exit
Switch(Config)#interface ethernet 1/0/1
Switch(config-If-Ethernet1/0/1)#mrpp ring 4000 primary-port
Switch(config-If-Ethernet1/0/1)#interface ethernet 1/0/2
Switch(config-If-Ethernet1/0/2)#mrpp ring 4000 secondary-port
Switch(config-If-Ethernet1/0/2)#exit
Switch(Config)#
```

SWITCH C configuration task sequence:

```
Switch(Config)#mrpp enable
Switch(Config)#mrpp ring 4000
Switch(mrpp-ring-4000)#control-vlan 4000
Switch(mrpp-ring-4000)#enable
Switch(mrpp-ring-4000)#exit
Switch(Config)#interface ethernet 1/0/1
Switch(config-If-Ethernet1/0/1)#mrpp ring 4000 primary-port
Switch(config-If-Ethernet1/0/1)#interface ethernet 1/0/2
Switch(config-If-Ethernet1/0/2)#mrpp ring 4000 secondary-port
Switch(config-If-Ethernet1/0/2)#exit
Switch(Config)#
```

SWITCH D configuration task sequence:

```
Switch(Config)#mrpp enable
Switch(Config)#mrpp ring 4000
Switch(mrpp-ring-4000)#control-vlan 4000
Switch(mrpp-ring-4000)#enable
Switch(mrpp-ring-4000)#exit
Switch(Config)#interface ethernet 1/0/1
Switch(config-If-Ethernet1/0/1)#mrpp ring 4000 primary-port
Switch(config-If-Ethernet1/0/1)#interface ethernet 1/0/2
Switch(config-If-Ethernet1/0/2)#mrpp ring 4000 secondary-port
Switch(config-If-Ethernet1/0/2)#exit
Switch(Config)#
```

8.3.4 MRPP Troubleshooting

Normal operation of MRPP depends on correct configuration of every switch on the MRPP ring; otherwise, loops and broadcast storms are very likely to occur:

- When configuring an MRPP ring, disconnect the ring first, wait until all switches are configured, and then close the ring.
- Before disabling MRPP on a switch in an MRPP ring, make sure that the ring has been disconnected.
- If a broadcast storm occurs on an MRPP ring, first disconnect the ring and check whether the MRPP configuration of each switch on the ring is correct. If it is correct, restore the ring and then observe whether the ring operates normally.
- The convergence time of an MRPP ring depends on the port up/down response mode. In poll mode, the convergence time is several hundred milliseconds in a simple ring network; in interrupt mode, it is within 50 milliseconds.

- Generally, ports are configured in poll mode; interrupt mode applies only to environments that require higher performance, but poll mode is more reliable than interrupt mode. Refer to the port-scan-mode {interrupt | poll} command.
- If a ring broadcast storm or ring blocking still occurs with a correct configuration, enable MRPP debugging on the primary node, use the show MRPP statistics command to check whether the states and statistics of the primary node and transfer nodes are normal, and send the results to EnGenius technical support.

8.4 ULPP

8.4.1 Introduction to ULPP

Each ULPP group has two uplink ports: the master port and the slave port. Each port can be a physical port or a port channel. Member ports of a ULPP group have three states: Forwarding, Standby, and Down. Normally, only one port is in the Forwarding state, and the other port is blocked in the Standby state. When the master port has a link failure, it enters the Down state, and the slave port switches to the Forwarding state.

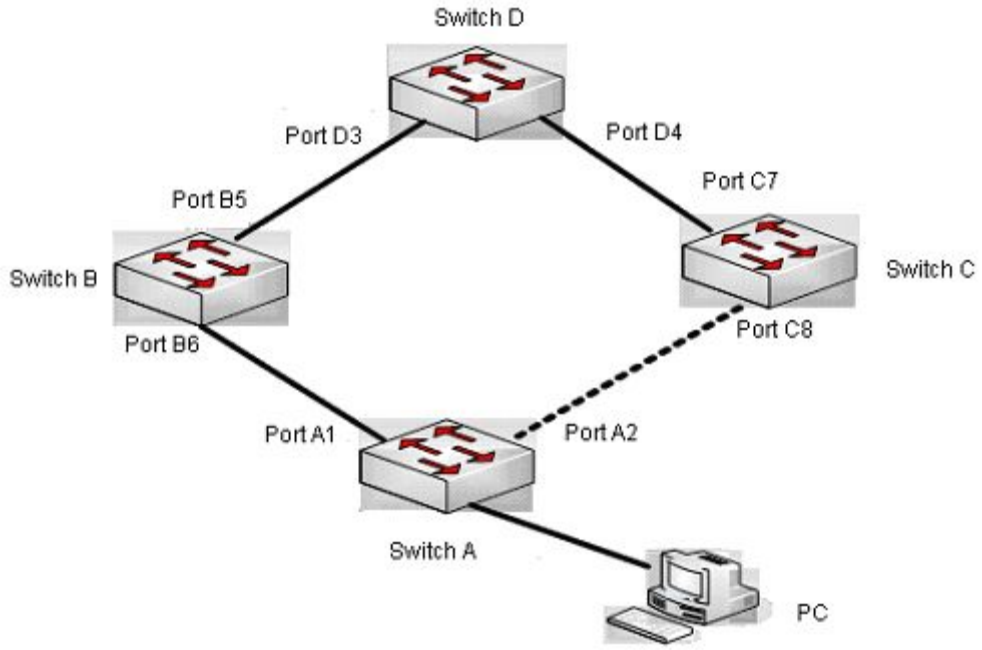


Figure 8-9 ULPP Application Scenario

The figure above shows a dual-uplink network, which is the typical ULPP application scenario. SwitchA connects upstream to SwitchD through SwitchB and SwitchC, with port A1 and port A2 as the uplink ports. ULPP is configured on SwitchA, with port A1 as the master port and port A2 as the slave port. When port A1, which is in the Forwarding state, fails, the uplink switches immediately and port A2 enters the Forwarding state. Later, when the master port recovers, if preemption mode is not configured, port A2 stays in the Forwarding state and port A1 enters the Standby state.

When preemption mode is enabled, the master port preempts the slave port after it recovers from the failure. To avoid frequent uplink switchovers caused by unstable faults, a preemption delay mechanism is used: the master port waits for a period of time before preempting the slave port. To keep traffic continuous, by default the master port does not preempt and enters the Standby state instead.

When configuring ULPP, specify the VLANs protected by the ULPP group through MSTP instances. ULPP does not protect other VLANs.

When an uplink switchover occurs, the existing forwarding entries on the devices no longer match the new network topology. In the figure, ULPP is configured on SwitchA, with port A1 as the master port in the Forwarding state, and SwitchD learns the MAC address of the PC on port D3. If port A1 then fails, traffic switches to port A2. Data sent by SwitchD to the PC is still forwarded from port D3 and is lost. Therefore, during an uplink switchover, the device running ULPP must send flush packets through the port that has switched to the Forwarding state to update the MAC address tables and ARP tables of other devices in the network. ULPP uses two types of flush packets to update these entries: MAC address update packets and ARP delete packets.

To make full use of bandwidth, ULPP can be configured for VLAN load balancing. As shown in the figure below, SwitchA is configured with two ULPP groups: in group 1, port A1 is the master port and port A2 is the slave port; in group 2, port A2 is the master port and port A1 is the slave port. Group 1 protects VLANs 1-100, and group 2 protects VLANs 101-200. Both port A1 and port A2 are in the Forwarding state; the master and slave ports back each other up and forward packets for different VLAN ranges. When port A1 fails, traffic for VLANs 1-200 is forwarded by port A2. Later, when port A1 recovers, port A2 continues to forward data for VLANs 101-200, and data for VLANs 1-100 switches back to port A1.

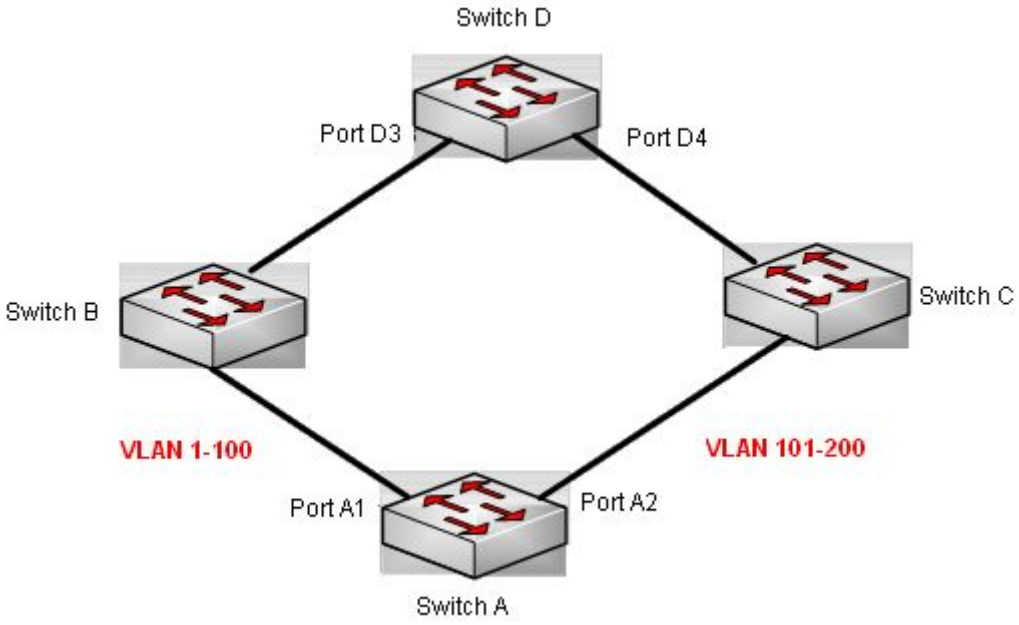


Figure 8-10 VLAN Load Balancing

8.4.2 ULPP Configuration Task List

1. Create a ULPP group globally
2. Configure the ULPP group
3. Display and debug ULPP information

1. Create a ULPP group globally

Command	Explanation
Global Mode	
ulpp group <*integer*> **no ulpp group <*integer*>**	Creates or deletes a ULPP group globally.

2. Configure the ULPP group

Command	Explanation
ULPP Group Mode	
preemption mode **no preemption mode**	Configures the preemption mode of the ULPP group. The no command deletes the preemption mode.
preemption delay <*integer*> **no preemption delay**	Configures the preemption delay. The no command restores the default value of 30s.
control vlan <*integer*> **no control vlan**	Configures the control VLAN used for sending. The no command restores the default value of 1.
protect vlan-reference-instance <*instance-list*> **no protect vlan-reference-instance <*instance-list*>**	Configures the protected VLANs. The no command deletes the protected VLANs.
flush enable mac **flush disable mac**	Enables or disables sending flush packets that update MAC addresses.
flush enable arp **flush disable arp**	Enables or disables sending flush packets that delete ARP entries.
flush enable mac-vlan **flush disable mac-vlan**	Enables or disables sending flush packets that delete dynamic unicast MAC addresses by VLAN.
description <*string*> **no description**	Configures or deletes the ULPP group description.
Port Mode	
ulpp control vlan <*vlan-list*> **no ulpp control vlan <*vlan-list*>**	Configures the control VLANs used for receiving. The no command restores the default value of 1.
ulpp flush enable mac **ulpp flush disable mac**	Enables or disables receiving flush packets that update MAC addresses.
ulpp flush enable arp **ulpp flush disable arp**	Enables or disables receiving flush packets that delete ARP entries.
ulpp flush enable mac-vlan **ulpp flush disable mac-vlan**	Enables or disables receiving flush packets of the mac-vlan type.
ulpp group <*integer*> master **no ulpp group <*integer*> master**	Configures or deletes the master port of the ULPP group.
ulpp group <*integer*> slave **no ulpp group <*integer*> slave**	Configures or deletes the slave port of the ULPP group.

3. Display and debug ULPP information

Command	Explanation
Admin Mode	
show ulpp group [group-id]	Displays the configuration of the configured ULPP groups.
show ulpp flush counter interface {ethernet <IFNAME> | <IFNAME>}	Displays statistics of flush packets.
show ulpp flush-receive-port	Displays the flush types and control VLANs received by the port.
clear ulpp flush counter interface <*name*>	Clears statistics of flush packets.

debug ulpp flush {send | receive} interface *<name>* **no debug ulpp flush {send | receive} interface *<name>***	Displays information about received and sent flush packets. The no command disables the display.
debug ulpp flush content interface *<name>* **no debug ulpp flush content interface *<name>***	Displays the contents of received flush packets. The no command disables the display.
debug ulpp error **no debug ulpp error**	Displays ULPP error information. The no command disables the display.
debug ulpp event **no debug ulpp event**	Displays ULPP event information. The no command disables the display.

8.4.3 ULPP Typical Examples

8.4.3.1 ULPP Typical Example 1

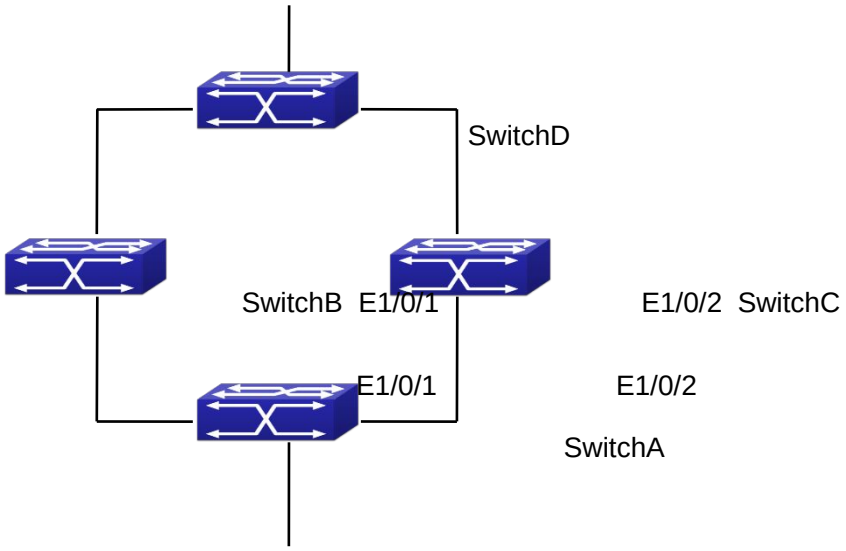


Figure 8-11 ULPP Typical Example 1

The topology above is a typical ULPP application environment.

SwitchA has two uplinks, to SwitchB and SwitchC. When no protocol is enabled, this topology forms a loop. To avoid the loop, configure ULPP on SwitchA with a master port and a slave port in a ULPP group. When both the master port and the slave port are up, the slave port is set to the Standby state and does not forward data packets. When the master port goes down, the slave port is set to the Forwarding state and takes over the uplink. Enable receiving of flush packets on SwitchB and SwitchC so that they cooperate with ULPP on SwitchA to switch the uplink immediately and reduce the switchover delay.

To configure ULPP on SwitchA, first create a ULPP group and configure VLAN 10 as its protected VLAN, then configure interface Ethernet 1/0/1 as the master port, interface Ethernet 1/0/2 as the slave port, and VLAN 10 as the control VLAN. On SwitchB and SwitchC, configure receiving of ULPP flush packets.

SwitchA configuration task list:

```
Switch(Config)#vlan 10
Switch(Config-vlan10)#switchport interface ethernet 1/0/1; 1/0/2
Switch(Config-vlan10)#exit
Switch(Config)#spanning-tree mst configuration
Switch(Config-Mstp-Region)#instance 1 vlan 10
```

```
Switch(Config-Mstp-Region)#exit
Switch(Config)#ulpp group 1
Switch(ulpp-group-1)#protect vlan-reference-instance 1
Switch(ulpp-group-1)#control vlan 10
Switch(ulpp-group-1)#exit
Switch(Config)#interface ethernet 1/0/1
Switch(config-If-Ethernet1/0/1)# ulpp group 1 master
Switch(config-If-Ethernet1/0/1)#exit
Switch(Config)#interface Ethernet 1/0/2
Switch(config-If-Ethernet1/0/2)# ulpp group 1 slave
Switch(config-If-Ethernet1/0/2)#exit
```

SwitchB configuration task list:

```
Switch(Config)#vlan 10
Switch(Config-vlan10)#switchport interface ethernet 1/0/1
Switch(Config-vlan10)#exit
Switch(Config)#interface ethernet 1/0/1
Switch(config-If-Ethernet1/0/1)# ulpp flush enable mac
Switch(config-If-Ethernet1/0/1)# ulpp flush enable arp
Switch(config-If-Ethernet1/0/1)# ulpp control vlan 10
```

SwitchC configuration task list:

```
Switch(Config)#vlan 10
Switch(Config-vlan10)#switchport interface ethernet 1/0/2
Switch(Config-vlan10)#exit
Switch(Config)#interface ethernet 1/0/2
Switch(config-If-Ethernet1/0/2)# ulpp flush enable mac
Switch(config-If-Ethernet1/0/2)# ulpp flush enable arp
Switch(config-If-Ethernet1/0/2)# ulpp control vlan 10
```

8.4.3.2 ULPP Typical Example 2

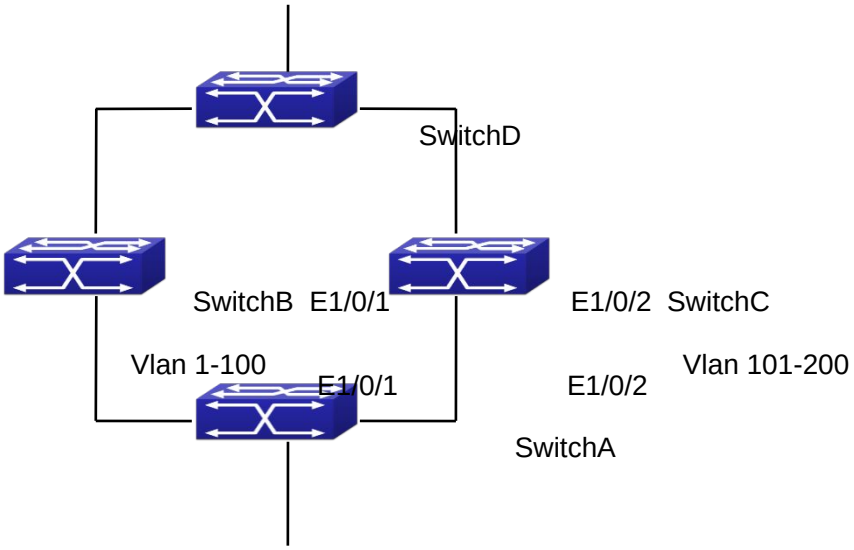


Figure 8-12 ULPP Typical Example 2

ULPP can implement VLAN-based load balancing. As shown in the figure above, SwitchA is configured with two ULPP groups: in group 1, port E1/0/1 is the master port and port 1/0/2 is the slave port; in group 2, port 1/0/2 is the master port and port 1/0/1 is the slave port. Group 1 protects VLANs 1-100, and group 2 protects VLANs 101-200. Both port E1/0/1 and port E1/0/2 are in the Forwarding state; the master and slave ports back each other up and forward packets for different VLAN ranges. When port E1/0/1 fails, traffic for VLANs 1-200 is forwarded by port E1/0/2. When port E1/0/1 recovers, port E1/0/2 continues to forward data for VLANs 101-200, and data for VLANs 1-100 switches back to port E1/0/1.

SwitchA configuration task list:

```
Switch(Config)#spanning-tree mst configuration
Switch(Config-Mstp-Region)#instance 1 vlan 1-100
Switch(Config-Mstp-Region)#instance 2 vlan 101-200
Switch(Config-Mstp-Region)#exit
Switch(Config)#ulpp group 1
Switch(ulpp-group-1)#protect vlan-reference-instance 1
Switch(ulpp-group-1)#preemption mode
Switch(ulpp-group-1)#exit
Switch(Config)#ulpp group 2
Switch(ulpp-group-2)#protect vlan-reference-instance 2
Switch(ulpp-group-1)#preemption mode
Switch(ulpp-group-2)#exit
Switch(Config)#interface ethernet 1/0/1
Switch(config-If-Ethernet1/0/1)#switchport mode trunk
Switch(config-If-Ethernet1/0/1)#ulpp group 1 master
Switch(config-If-Ethernet1/0/1)#ulpp group 2 slave
Switch(config-If-Ethernet1/0/1)#exit
Switch(Config)#interface Ethernet 1/0/2
Switch(config-If-Ethernet1/0/2)#switchport mode trunk
Switch(config-If-Ethernet1/0/2)# ulpp group 1 slave
Switch(config-If-Ethernet1/0/2)# ulpp group 2 master
Switch(config-If-Ethernet1/0/2)#exit
```

SwitchB configuration task list:

```
Switch(Config)#interface ethernet 1/0/1
Switch(config-If-Ethernet1/0/1)#switchport mode trunk
Switch(config-If-Ethernet1/0/1)# ulpp flush enable mac
Switch(config-If-Ethernet1/0/1)# ulpp flush enable arp
```

SwitchC configuration task list:

```
Switch(Config)#interface ethernet 1/0/2
Switch(config-If-Ethernet1/0/2)# switchport mode trunk
Switch(config-If-Ethernet1/0/2)# ulpp flush enable mac
Switch(config-If-Ethernet1/0/2)# ulpp flush enable arp
```

8.4.4 ULPP Troubleshooting

- ☞ More than two uplinks can currently be configured, but this may cause loops and is not recommended.
- ☞ If a broadcast storm occurs or communication along the ring is broken with a correct configuration, enable ULPP debugging, record 3 minutes of debug information together with the configuration, and send them to EnGenius technical support.

8.5 ULSM

8.5.1 Introduction to ULSM

ULSM (Uplink State Monitor) synchronizes port states. Each ULSM group consists of uplink ports and downlink ports; there can be multiple uplink ports and multiple downlink ports. Each port can be a physical port or a port channel, but not a member port of a port channel, and each port can belong to only one ULSM group.

An uplink port is a monitored port of the ULSM group. When all uplink ports are down, or the ULSM group has no uplink port, the ULSM group state is down. The ULSM group state is up as long as at least one uplink port is up.

A downlink port is a controlled port. Its state follows the up/down state of the ULSM group and is always the same as the ULSM group state.

ULSM works with ULPP so that a downstream device can detect a link failure on the upstream device and respond correctly. As shown in the figure below, ULPP is configured on SwitchA, and traffic is forwarded through port A1. If the link between SwitchB and SwitchD fails, SwitchA cannot detect the upstream link failure and continues to forward traffic through port A1, causing traffic loss.

Configuring ULSM on SwitchB solves this problem: set port B5 as the uplink port of a ULSM group and port B6 as the downlink port. When the link between SwitchB and SwitchD fails, both the ULSM group and the downlink port B6 go down. This causes SwitchA, on which ULPP is configured, to perform an uplink switchover and avoid data loss.

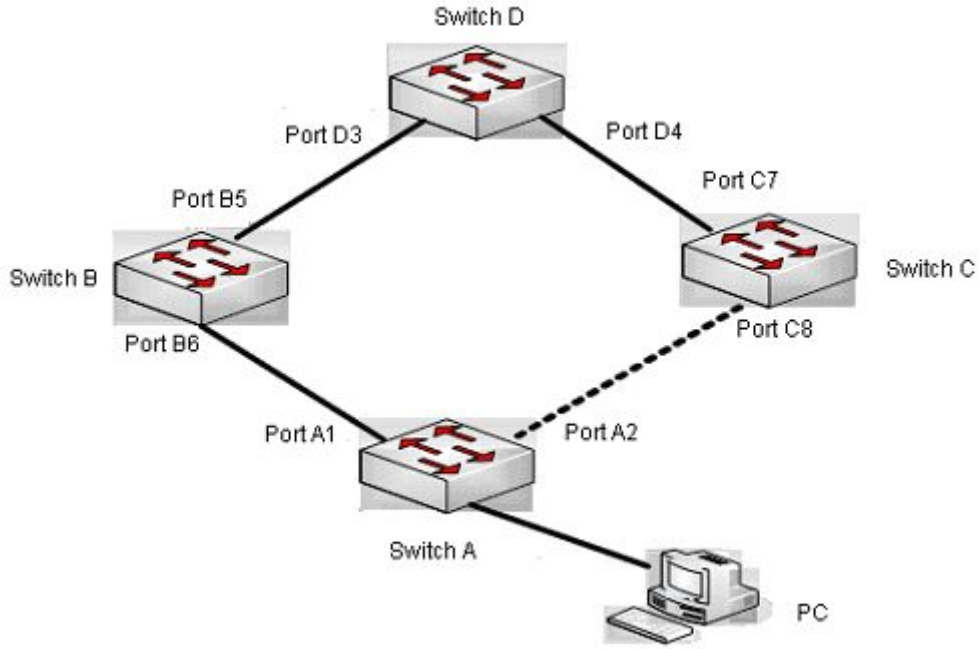


Figure 8-13 ULSM Application Scenario

8.5.2 ULSM Configuration Task List

1. Create a ULSM group globally
2. Configure the ULSM group
3. Display and debug ULSM information

1. Create a ULSM group globally

Command	Explanation
Global Mode	
ulsm group <*group-id*> **no ulsm group <*group-id*>**	Creates or deletes a ULSM group globally.

2. Configure the ULSM group

Command	Explanation
Port Mode	
ulsm group <*group-id*> {uplink | downlink} **no ulsm group <*group-id*> {uplink | downlink}**	Configures the uplink or downlink port of the ULSM group. The no command deletes the uplink or downlink port.

3. Display and debug ULSM information

Command	Explanation

Admin Mode	
show ulsm group [group-id]	Displays the configuration of the ULSM group.
debug ulsm event **no debug ulsm event**	Displays ULSM event information. The no command disables the display.

8.5.3 ULSM Typical Example

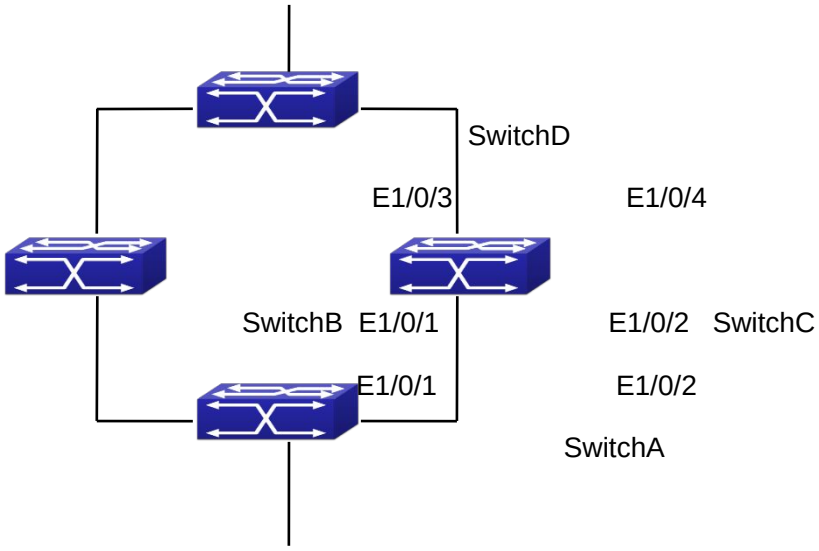


Figure 8-14 ULSM Typical Example

The topology above is a typical application environment for ULSM and ULPP.

ULSM synchronizes port states and is not useful on its own, so it is usually used together with ULPP. In this topology, SwitchA runs ULPP to switch the uplink. SwitchB and SwitchC run ULSM to monitor whether the uplink is down. If it is down, ULSM brings down the downlink port, so that ULPP on SwitchA performs the uplink switchover.

SwitchA configuration task list:

```
Switch(Config)#spanning-tree mst configuration
Switch(Config-Mstp-Region)#instance 1 vlan 1
Switch(Config-Mstp-Region)#exit
Switch(Config)#ulpp group 1
Switch(ulpp-group-1)#protect vlan-reference-instance 1
Switch(ulpp-group-1)#exit
Switch(Config)#interface ethernet 1/0/1
Switch(config-If-Ethernet1/0/1)# ulpp group 1 master
Switch(config-If-Ethernet1/0/1)#exit
Switch(Config)#interface Ethernet 1/0/2
Switch(config-If-Ethernet1/0/2)# ulpp group 1 slave
Switch(config-If-Ethernet1/0/2)#exit
```

SwitchB configuration task list:

```
Switch(Config)#ulsm group 1
Switch(Config)#interface ethernet 1/0/1
Switch(config-If-Ethernet1/0/1)#ulsm group 1 downlink
Switch(config-If-Ethernet1/0/1)#exit
Switch(Config)#interface ethernet 1/0/3
Switch(config-If-Ethernet1/0/3)#ulsm group 1 uplink
Switch(config-If-Ethernet1/0/3)#exit
```

SwitchC configuration task list:

```
Switch(Config)#ulsm group 1
Switch(Config)#interface ethernet 1/0/2
Switch(config-If-Ethernet1/0/2)#ulsm group 1 downlink
Switch(config-If-Ethernet1/0/2)#exit
Switch(Config)#interface ethernet 1/0/4
Switch(config-If-Ethernet1/0/4)#ulsm group 1 uplink
Switch(config-If-Ethernet1/0/4)#exit
```

8.5.4 ULSM Troubleshooting

☞ If the downlink port does not respond to the down event of the uplink port with a correct configuration, enable ULSM debugging, record 3 minutes of debug information together with the configuration, and send them to EnGenius technical support.

8.6 ERPS

8.6.1 Introduction to ERPS

ERPS (Ethernet Ring Protection Switching) is a Layer 2 loop-prevention protocol defined by ITU-T in standard ITU-T G.8032/Y1344, also known as G.8032. G.8032 incorporates the advantages of ring protection technologies such as EAPS, RPR, SDH, and STP. It optimizes the detection mechanism, can detect bidirectional and unidirectional faults, and supports multi-ring and multi-domain structures. In addition to 50 ms switchover, it supports master/slave operation and load sharing. It is the newest Ethernet ring technology standard.

ERPS is a loop-prevention protocol for ring network protection. It covers link switchover on a ring fault, notification, and ring switchback after the ring is restored, but it does not cover link fault discovery. Fault discovery can use the CCM function defined by 802.1ag or physical link fault detection. Whichever detection mechanism is used, a link fault must be detected quickly and reported to the ERPS module. ERPS requires that traffic interruption caused by a link fault be no more than 50 ms for a ring of up to 1200 km with 16 nodes, which is demanding for both link fault discovery and protection switching time.

8.6.1.1 ERPS Terminology

Ethernet ring: A closed physical ring network made up of multiple ring nodes. Each node on the ring has exactly two ports connected to the ring network.

Ring protection link: The RPL is a link on the ring network that is blocked by its nodes and does not carry data traffic when the ring network is healthy.

RPL owner node: The node connected to the RPL that blocks the RPL when the ring network is healthy. When configured as revertive, it also initiates switchback after the ring network recovers.

RPL neighbor node: The other node connected to the RPL. It also blocks the RPL when the ring network is healthy.

Interconnection node: When multiple rings intersect, the node at the intersection. On an interconnection node, one or more rings are connected through two ports. A ring connected through one port is a sub-ring, and a ring connected through two ports is a major ring.

R-APS virtual channel: The link that connects the two interconnection nodes of a sub-ring outside the sub-ring path. Its transmission characteristics depend on the network outside the ring.

Major ring: The ring connected through two ports on an interconnection node.

Sub-ring: A ring connected to another network through two interconnection nodes. It is not a closed ring by itself; it forms a ring only when connected through the interconnection nodes.

ERP instance: A set of protected VLANs. Packets of the VLANs in this instance are transmitted over the same ring network links, and each VLAN belongs to only one instance.

Revertive switch: After learning that the ring network fault has been cleared, the RPL owner node blocks the RPL again and restores the traffic path to the one used before the fault.

Non-revertive switch: After learning that the ring network fault has been cleared, the RPL owner node does not block the RPL, and the traffic path remains unchanged.

8.6.1.2 ERPS Function

8.6.1.2.1 Fault Switchover

The following shows a single-ring, single-link fault.

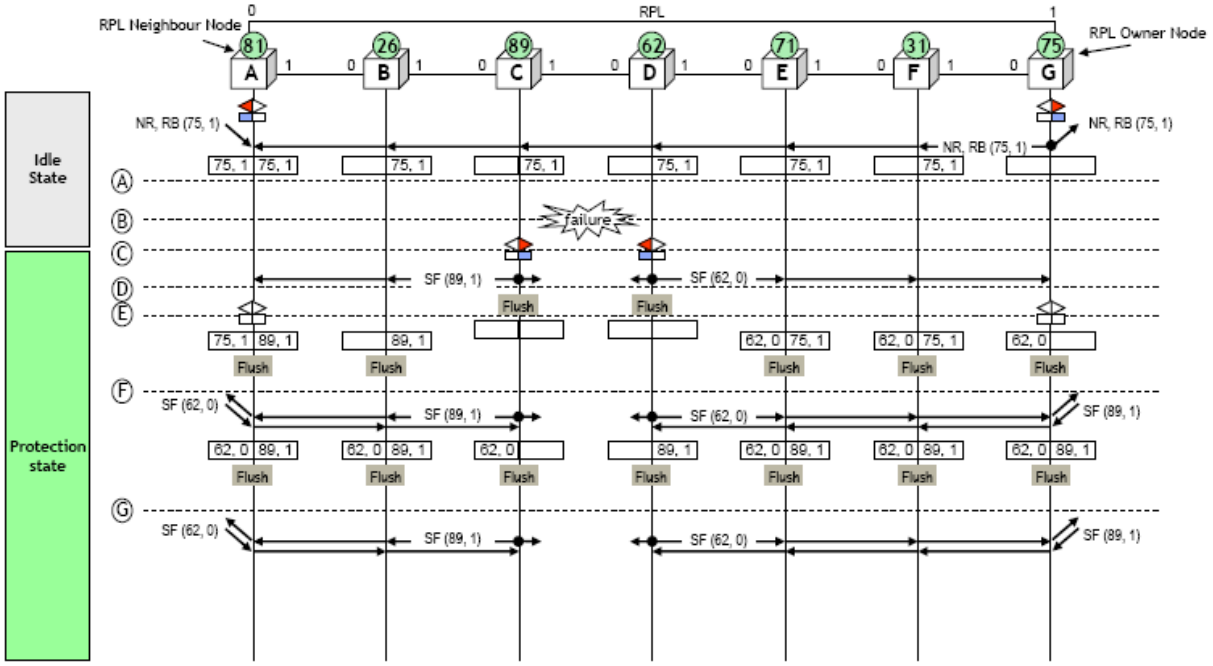


Figure 7-15 Single-Link Fault

The fault switchover steps are:

a) The ring network is normal; RPL owner node G periodically sends R-APS (NR, RB) packets, indicating that the RPL is blocked and the ring network is healthy.

b) A fault occurs on the link between nodes C and D.

c) Nodes C and D detect the fault, each blocks its port connected to the faulty link, and each performs a flush FDB.

d) At the same time, nodes C and D each send R-APS (SF) fault notification packets through their ports connected to the ring network.

e) All nodes that receive the R-APS (SF) packets perform a flush FDB. At the same time, RPL owner node G and RPL neighbor node A set their RPL ports to forwarding, and node G stops sending R-APS (NR, RB) packets.

f) Because the RPL is unblocked, all nodes can receive two R-APS (SF) packets (sent by nodes C and D). After receiving a new R-APS (SF) packet, a node performs a flush FDB.

g) The R-APS (SF) link fault messages continue to be transmitted in the ring network.

8.6.1.2.2 Failure Recovery

When the ring link recovers, ring nodes use one of two methods. One is revertive switching: after the ring link returns to normal, the ring network blocks the RPL again and restores the forwarding state of the previously faulty link, so the data forwarding path is the same as before the fault. The other is

non-revertive switching: after the ring link returns to normal, the link remains blocked, and data packets continue to be forwarded along the current path.

The two methods suit different environments. When blocking the RPL gives the best traffic path, use revertive switching. When the path costs are similar, so it makes no difference which path is blocked, use non-revertive switching to prevent a second interruption of traffic.

1. **Revertive switch**

The following shows a single-ring, single-link fault.

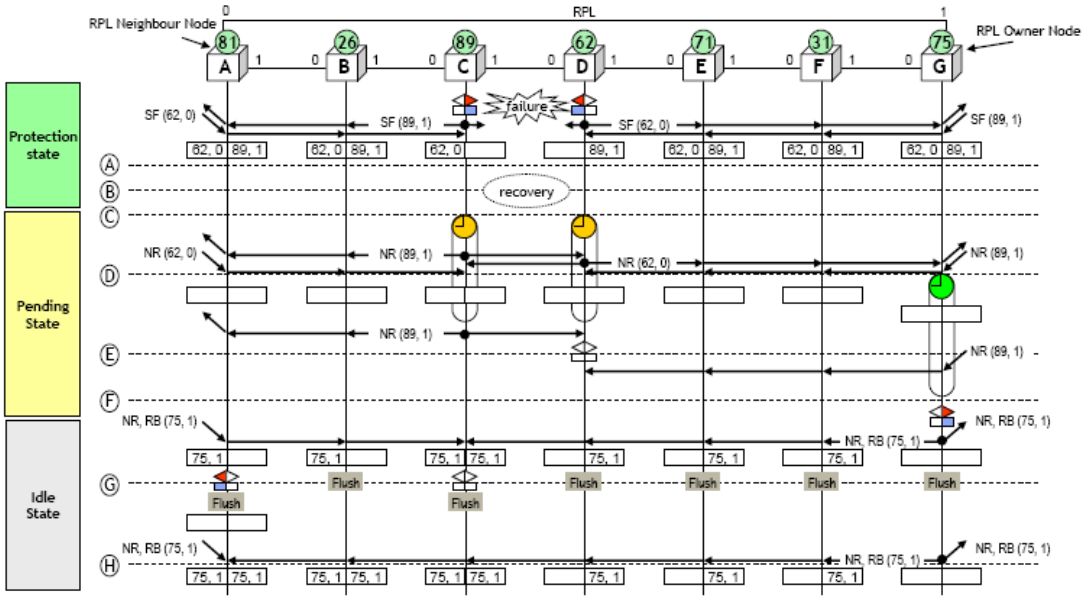


Figure 7-16 Revertive Switch Fault Recovery on a Single Link

The steps are:

a) While the fault persists, the nodes that detected the fault periodically send R-APS (SF) packets with the fault message.

b) The link fault is cleared.

c) Nodes C and D detect the fault recovery, start the guard timer, and at the same time send R-APS (NR) fault recovery packets on their ring ports.

d) When the RPL owner node receives the R-APS (NR) packets, it starts the WTR timer and at the same time clears the local node fault message.

e) After their timers expire, nodes C and D receive R-APS (NR) packets from each other. Node D determines that node C has higher priority, so it stops sending R-APS (NR) packets with its local message and unblocks its port.

f) When the WTR timer of RPL owner node G expires, node G blocks its RPL port and sends R-APS (NR, RB) packets through its ring ports to notify other nodes that the RPL has been blocked. At the same time, node G performs a flush FDB.

g) When node C receives the R-APS (NR, RB) packets sent by the RPL owner node, it unblocks its local port and stops sending NR packets. When RPL neighbor node A receives this packet, it blocks its RPL port. In addition, all nodes perform a flush FDB after receiving the R-APS (NR, RB) packets.

2. **Non-revertive switch**

The following example shows a single-ring, single-link fault, as shown in Figure 7-17.

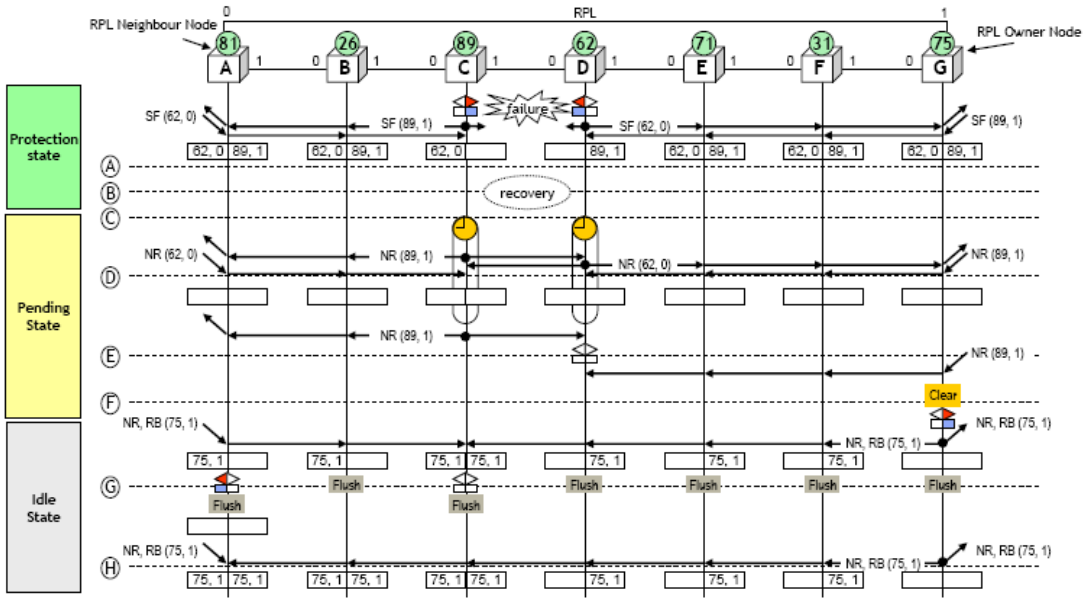


Figure 7-17 Non-Revertive Switch Fault Recovery on a Single Link

The steps are:

a) While the fault persists, the nodes that detected the fault periodically send R-APS (SF) packets with the fault message.

b) The link fault is cleared.

c) Nodes C and D detect the fault recovery, start the guard timer, and at the same time send R-APS (NR) fault recovery packets on their ring ports.

d) When RPL owner node G receives the R-APS (NR) packets, it clears the local node fault message but does not start the WTR timer, because non-revertive mode is configured.

e) After their timers expire, nodes C and D receive R-APS (NR) packets from each other. Node D determines that node C has higher priority, so it stops sending R-APS (NR) packets and unblocks its local port.

f) If the clear command is run on RPL owner node G, node G reverts to revertive mode, blocks its RPL port, and sends R-APS (NR, RB) packets through its ring ports to notify other nodes that the RPL has been blocked. At the same time, it performs a flush FDB.

g) When node C receives the packets sent by the RPL owner node, it unblocks its local port and stops sending R-APS (NR) packets. When RPL neighbor node A receives this packet, it blocks its RPL port. In addition, all nodes perform a flush FDB after receiving the packets.

8.6.1.2.3 Interconnection Ring Model

ERPS supports protection switching of interconnected rings. There are two interconnected ring models: with a virtual channel and without a virtual channel.

1. Interconnected ring model with a virtual channel

As shown in Figure 7-18, three ring networks are interconnected. Ring 1 is a major ring made up of nodes A, B, G, and H and the links between them; when ring 1 is healthy, the link between nodes A and B is blocked. Ring 2 is another major ring made up of nodes C, D, E, and F and the links between them; when ring 2 is healthy, the link between nodes C and D is blocked. Ring 3 is a sub-ring made up of nodes B, C, F, and G and links B-C and G-F; when ring 3 is healthy, link B-C is blocked. Link B-G is the interconnection link of ring 1 and ring 3 and belongs to ring 1. Link C-F is the interconnection link of ring 2 and ring 3 and belongs to ring 2. Ring 1 and ring 2 are closed rings; ring 3 is not. If ring 1 is treated as the link (virtual channel) between interconnection nodes B and G of ring

3, and ring 2 is treated as the link (virtual channel) between interconnection nodes C and F of ring 3, ring 3 becomes a ring.

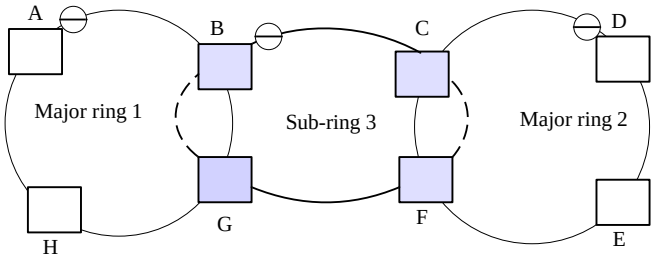


Figure 7-18 Interconnected Ring Topology with a Virtual Channel

The R-APS virtual channel provided by ring 1 and ring 2 treats ring 3 protocol packets as data packets and transmits them in the same way as data packets. Node B of ring 3 sends ring 3 ERPS protocol packets to node G and receives those sent by node G, and vice versa. To distinguish the ERPS packets of ring 3 from those of the major rings in ring 1 and ring 2, a different control VLAN can be used for the protocol packets of each ring.

When sub-ring 3 changes, it notifies ring 1 and ring 2, and the nodes on the major rings perform a flush FDB. Topology changes in major ring 1 and ring 2 do not affect sub-ring 3, and topology changes in major ring 1 and ring 2 do not affect each other.

2. Interconnected ring model without a virtual channel

Consider the ring network shown in Figure 7-19 differently: ring 1 is a major ring made up of nodes A, B, G, and H and the links between them; when ring 1 is healthy, the link between nodes A and B is blocked. Ring 2 is a sub-ring made up of nodes C, D, E, and F and links C-D, D-E, and E-F; when ring 2 is healthy, the link between nodes C and D is blocked. Ring 3 is another sub-ring made up of nodes B, C, F, and G and links B-C, C-F, and F-G; when ring 3 is healthy, link B-C is blocked. Link B-G is the interconnection link of ring 1 and ring 3 and belongs to ring 1. Link C-F is the interconnection link of ring 2 and ring 3 and belongs to ring 3. Ring 1 is a closed ring; ring 2 and ring 3 are not.

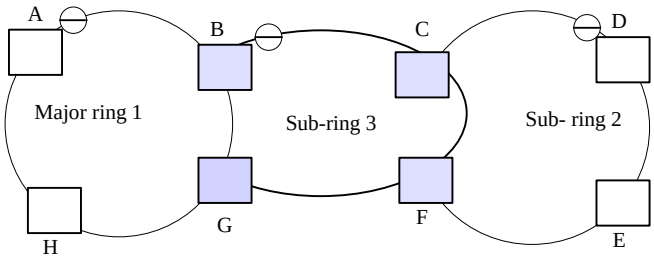


Figure 7-19 Interconnected Ring Topology without a Virtual Channel

Although ring 2 and ring 3 are not closed rings, the ERPS packets of these two sub-rings must reach all ring nodes. Therefore, even if links B-C and C-D are blocked, they must still transmit ERPS protocol packets, and nodes B, C and C, D of the blocked links must still receive and send ERPS protocol packets.

When sub-ring 3 changes, it notifies ring 1, and the nodes on the major ring perform a flush FDB; the change does not affect sub-ring 2. When sub-ring 2 changes, it affects sub-ring 3 and major ring 1, and the nodes on the major ring must perform a flush FDB. However, topology changes in major ring 1 do not affect sub-ring 2 or ring 3.

8.6.1.3 ERPS Application

ERPS is used in ring networks at the aggregation layer. The aggregation ring performs Layer 2 aggregation of services and connects to the Layer 3 network for service processing. The aggregation ring runs ERPS to provide Layer 2 redundancy protection switching.

8.6.2 ERPS Configuration

ERPS configuration task list:

1) Create an instance and map the VLANs to be protected to it

2) Create an ERPS ring and configure the member port information. Default configuration: version V2 is supported, the ring is a major (closed) ring, and the physical status of the ports is monitored

3) Configure the ERPS ring instance: configure the protected instance and port roles; configure the ERPS ring instance name, R-APS level, and timers; finally, configure the control VLAN and select the ports to configure as RPL owner and RPL neighbor

1. Create the MSTP instance

Command	Explanation
Global Mode	
spanning-tree mst configuration **no spanning-tree mst configuration**	Enters MSTP Region Mode to configure MSTP region parameters. The no command restores the default.
MSTP Region Mode	
instance <instance value > vlan <vlan-list> **no instance [instance-value]**	Configures the VLANs to be protected by the instance and their mapping. The no command deletes the specified instance.

2. Create an ERPS ring and configure the member port information

Command	Explanation
Global Mode	
erps ring <ring-name> **no erps ring <ring-name>**	Creates an ERPS ring and enters ERPS Ring Configuration Mode. The no command deletes the specified ERPS ring.
Port Mode	
erps-ring <ring-name> port0 **erps-ring <ring-name> port1** **erps-ring <ring-name> port0** **erps-ring <ring-name> port1**	Configures the port as port0 or port1 of the ring node. The no command deletes the property.

3. Configure the ERPS ring instance

Command	Explanation
Global Mode	
erps-ring <ring-name> **no erps-ring <ring-name>**	Creates an ERPS ring and enters ERPS Ring Configuration Mode. The no command deletes the specified ERPS ring.
ERPS Ring Configuration Mode	
eprs-instance <instance-id> **no eprs-instance <instance-id>**	Creates an ERPS ring instance and enters its configuration mode. The no command deletes the specified ring instance.
description <*instance-name*> ***no* description**	Configures the description string of the ERPS instance. The no command deletes the string.
rpl {port0 | port1} {owner | neighbour} **no rpl {port0 | port1}**	Configures a member port of the ERPS ring instance as the RPL owner or RPL neighbor. The no command deletes the specified owner or neighbor setting.

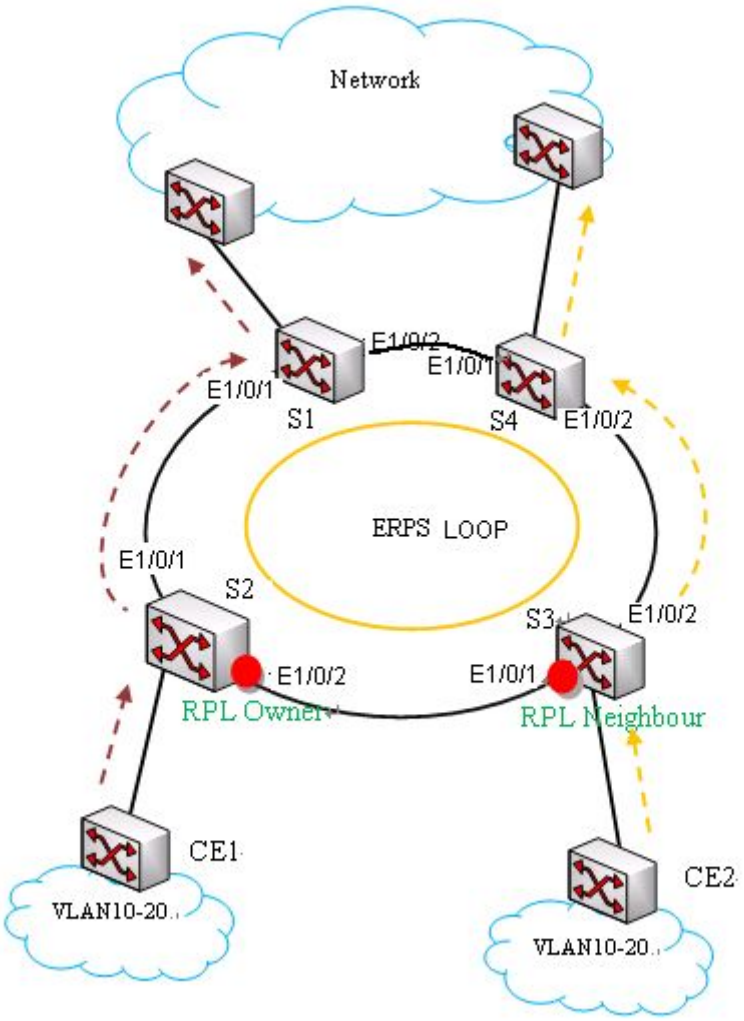
raps-mel <level-value> **no raps-mel**	Configures the R-APS channel level. The MEL field in protocol packets is used to determine whether the current packet can pass. The no command deletes the R-APS channel level.
protected-instance <*instance-list*> **no protected-instance**	Configures the protected instances of the ERPS ring instance. The no command deletes the protected instances.
wtr-timer <*wtr-times*> **no wtr-timer**	Configures the WTR timer. The WTR timer prevents frequent switchovers by the RPL owner node caused by intermittent faults. The no command deletes the WTR timer.
guard-timer <*guard-times*> **no guard-timer**	Configures the Guard timer. The Guard timer prevents an Ethernet ring node from acting on outdated R-APS packets and prevents closed loops. The no command deletes the Guard timer.
holdoff –timer <*holdoff-times*> **no holdoff –timer**	Configures the Holdoff timer. The Holdoff timer sets how long an Ethernet ring node delays reporting a fault. The no command deletes the Holdoff timer.
control-vlan <*vlan-id*> **no control-vlan**	Configures the control VLAN used by the R-APS channel to transmit R-APS packets. In the ERPS ring instance, this VLAN carries only ERPS protocol packets, not user service packets, which improves the security of ERPS. The no command deletes the control VLAN.

4. Display ERPS configuration information

Command	Explanation
Global Mode	
show erps ring {<ring-name> | brief}	Displays ERPS ring information.
show erps instance [ring <ring-name> [instance <instance-id>]]	Displays ERPS ring instance information.
show erps status [ring <ring-name> [instance <instance-id>]]	Displays ERPS ring instance status information.
show erps statistics [ring <ring-name> [instance <instance-id>]]	Displays ERPS ring instance statistics.

8.6.3 ERPS Examples

Case 1:



The figure above illustrates ERPS configuration and application. S1 to S4 form a ring network that provides Layer 2 redundancy protection switching. To prevent loops for packets in VLAN 10 to VLAN 20, deploy ERPS on the devices that form the ring. The forwarding path for user data accessing through CE1 is S2-S1, and for user data accessing through CE2 it is S3-S4. To protect against Ethernet ring failures, configure as follows:

1. Configuration approach

Configure ERPS ring redundancy protection as follows:

Create the ERPS ring maijor_ring1 and configure the ring member ports.

Configure instance 1 on ERPS ring maijor_ring1, and configure the protected instance, member port roles, timers, and control VLAN.

8.6.4 ERPS Troubleshooting

If the configured ERPS ring does not provide Ethernet ring protection switching, check for the following causes:

- ☞ Check that the basic configuration is correct and that the protected instance, control-vlan, wtr-timer, guard-timer, and raps-mel are consistent on every node.
- ☞ Check that the VLAN of user data traffic is different from the control VLAN. In an ERPS ring instance, the control VLAN transmits only ERPS protocol packets, not user service packets, which improves the security of ERPS; make sure this configuration is unique. This VLAN is

used as the VLAN tag when R-APS packets are sent. The protected VLAN configuration must be consistent on all nodes in the instance.

- ☞ We recommend configuring the ports of ERPS nodes as trunk ports and making sure that the data VLANs and control VLAN are in the protected instance, because ERPS protects only data and protocol packets in the instance. For example, if the switch enables a protocol (CFM, EFM, or a Layer 3 interface) that sends protocol packets, and the VLAN ID used to send those packets is not in the protected instance, a loop will occur in the topology.
- ☞ If the port status detection method is configured as fastlink, the hardware must support the fastlink function; port status change notifications are interrupted, and MAC software learning is disabled at the same time.
- ☞ If ERPS is associated with CFM, the hardware must support the CC function and be able to send CCM packets at 3.3 ms intervals.

Chapter 9 Debugging and Diagnosis Configuration

9.1 Monitor and Debug

When configuring the switch, you need to verify that the configuration is correct and that the switch is operating as expected. When a network failure occurs, you also need to diagnose the problem. The switch provides various diagnostic commands, such as ping, telnet, show, and debug, to check the system configuration and operating status and to locate the causes of problems.

9.1.1 Ping

The Ping command sends ICMP query packets from the switch to a remote device to check whether the remote device is reachable from the switch. For descriptions of the Ping command parameters and options, see the Ping command chapter in the Command Guide.

9.1.2 Ping6

The Ping6 command sends ICMPv6 query packets from the switch to a remote device to check whether the remote device is reachable from the switch. For descriptions of the Ping6 command parameters and options, see the Ping6 command chapter in the Command Guide.

9.1.3 Traceroute

The Traceroute command tests the gateways that data packets pass through from the source device to the destination device. It is used to check network reachability and locate network failures.

The Traceroute command works as follows: First, a packet with a TTL of 1 is sent to the destination address. If the first hop returns an ICMP error message indicating that the packet cannot be forwarded (because the TTL expired), a packet with a TTL of 2 is sent. The second hop may also return a TTL-expired message, and the procedure continues until a packet reaches the destination. The procedure records the source address of each ICMP TTL-expired message to describe the path that the IP packets travel to reach the destination.

For descriptions of the Traceroute command parameters and options, see the traceroute command chapter in the Command Guide.

9.1.4 Traceroute6

The Traceroute6 command tests the gateways that data packets pass through from the source device to the destination device, to verify reachability and locate network failures. Traceroute6 in IPv6 works on the same principle as Traceroute in IPv4 and uses the hop limit field of the ICMPv6 and IPv6 headers. First, Traceroute6 sends an IPv6 datagram (including the source address, destination address, and the time the packet was sent) with the HOPLIMIT set to 1. When the first router on the path receives this datagram, it decrements the HOPLIMIT by 1 to 0. The router then discards the datagram and returns an "ICMPv6 time exceeded" message (including the source address of the IPv6 packet, the entire content of the IPv6 packet, and the IPv6 address of the router). On receiving this message, Traceroute6 sends another datagram with the HOPLIMIT increased to 2 to discover the second router. Traceroute6 increases the HOPLIMIT by 1 each time to discover the next router and repeats this action until a datagram reaches the destination.

For descriptions of the Traceroute6 command parameters and options, see the traceroute6 command chapter in the Command Guide.

9.1.5 Show

The **show** command displays information about the system, ports, and protocol operation. This section describes the **show** commands that display system information; other **show** commands are described in other chapters.

Command	Explanation
Admin Mode	
show debugging	Displays the debugging status.
show flash	Displays the files saved in flash memory and their sizes.
show history	Displays the commands recently entered by the user.
show history all-users [detail]	Displays the recent command history of all users. Use the **clear history all-users** command to clear the command history of all users saved by the system. The maximum number of history entries can be set with the **history all-users max-length** command.
show memory	Displays the contents of the specified memory area.
show running-config	Displays the switch configuration currently in effect.
show running-config current-mode	Displays the configuration in the current mode.
show startup-config	Displays the switch configuration saved in flash memory, which is normally the configuration file applied the next time the switch starts up.
show switchport interface [ethernet *<IFNAME>*]	Displays the VLAN port mode and VLAN membership of switch ports, as well as trunk port information.
show tcp **show tcp ipv6**	Displays the status of the TCP connections currently established on the switch.
show udp **show udp ipv6**	Displays the status of the UDP connections currently established on the switch.
show telnet login	Displays information about the Telnet clients that currently have a Telnet connection with the switch.
show tech-support	Displays the operating information and status of each task running on the switch. Technicians use it to diagnose whether the switch is operating properly.
show version	Displays the version of the switch.
show temperature	Displays the CPU temperature of the switch.
show fan	This command is not supported by switch.

9.1.6 Debug

All protocols supported by the switch have corresponding debug commands, which provide information for troubleshooting. The debug commands for each protocol are described in the later chapters.

9.2 Reload Switch After Specified Time

9.2.1 Introduction to Reloading the Switch After a Specified Time

This function reboots the switch after a specified period without powering it off. It is typically used when upgrading the switch software: the switch can be rebooted after a specified period instead of immediately after the new version is installed successfully.

9.2.2 Reload Switch After Specified Time Task List

1. Reload the switch after a specified time

Command	Explanation
Admin Mode	
reload after {[<*HH:MM:SS*>] [days <*days*>]}	Reloads the switch after the specified time period.
reload cancel	Cancels the scheduled reload of the switch.

9.3 Debugging and Diagnosis for Packets Received and Sent by CPU

9.3.1 Introduction to Debugging and Diagnosis for Packets Received and Sent by CPU

The following commands are used to debug and diagnose packets received and sent by the CPU. Use them with the help of technical support.

9.3.2 Debugging and Diagnosis for Packets Received and Sent by CPU Task List

Command	Explanation
Global Mode	
cpu-rx-ratelimit total <*packets*> **no cpu-rx-ratelimit total**	Sets the total rate at which the CPU receives packets. The no command restores the default total rate.
cpu-rx-ratelimit protocol <*protocol-type*> <*packets*> **no cpu-rx-ratelimit protocol [<*protocol- type*>]**	Sets the maximum rate at which the CPU receives packets of the specified protocol type. The no command restores the default maximum rate.
clear cpu-rx-stat protocol [<*protocol-type*>]	Clears the statistics of packets of the specified protocol type received by the CPU.
Admin Mode	
show cpu-rx protocol [<*protocol-type*>]	Displays information about packets of the specified protocol type received by the CPU.
debug driver {receive|send} [interface {<*interface-name*> | all}] [protocol {<*protocol-type*> |discard |all}] [detail]	Enables the display of information about packets received or sent by the CPU.
no debug driver {receive |	Disables the display of information about packets received

send}	or sent by the CPU.
Command	Explanation
Admin Mode	
protocol filter {protocol-type} **no Protocol filter {protocol-type}**	Enables or disables processing of packets of the specified protocol. The supported protocols are: **{arp|bgp|dhcp|dhcpv6|hsrp|http|igmp|ip|ldp|mpls|ospf|pim|rip|snmp|telnet|vrrp}**

9.4 Debugging for CPU-Protect

9.4.1 Introduction to CPU-Protect

During normal operation of the switch, only a small portion of packets is sent to the CPU for processing, mainly protocol packets and some packets that require software forwarding. Protocol packet attacks are generally handled by the protection module of each protocol, but not all protocols implement a protection module, and each protection module takes effect only after the kernel has received the packet and before the protocol module processes it. By then, the attack packets have already consumed some CPU resources. Intercepting attack packets at the front end, before they are sent to the CPU, provides better protection against network attacks and keeps the CPU running stably.

9.4.2 CPU-Protect Configuration

Command	Explanation
Global Mode	
cpu-protect enable **no cpu-protect enable**	Enables the cpu-protect function. The no command disables it.
cpu-protect log enable **no cpu-protect log enable**	Enables cpu-protect logging. The no command disables it.
cpu-protect trap enable **no cpu-protect trap enable**	Enables cpu-protect trap alarms. The no command disables them.
cpu-protect interval <time> **no cpu-protect interval**	Configures the cpu-protect sampling period.
cpu-protect recovery-time <time> **no cpu-protect recovery-time**	Configures the cpu-protect policy recovery period.
cpu-protect (protocol <WORD>|per-ip|per-mac) threshold <pps> action (none|speed-limit) **no cpu-protect (protocol <WORD>|per-ip|per-mac) threshold**	Configures the upper limit of the rate at which a flow (identified by protocol type, IP address, or MAC address) is sent to the CPU.

cpu-protect (protocol <WORD>|per-ip| per-mac) limit-speed <pps> **no cpu-protect (protocol <WORD>|per-ip|per-mac) limit-speed**	Configures the rate limit applied to a flow (identified by protocol type, IP address, or MAC address) sent to the CPU after the threshold is exceeded.
Admin Mode	
show cpu-protect attack-list (detail|top)	Displays the attack source information detected by the cpu-protect module.
show cpu-protect running-config	Displays the configuration and operating information of the cpu-protect module.

9.5 Info-center

9.5.1 Overview

9.5.1.1 Basic Concepts

The information center is a set of mechanisms that outputs system information (hereafter referred to as information sources).

To output the desired log information, configure and enable the corresponding output direction and set the matching conditions. The configuration process is simple and flexible.

9.5.1.2 Information Sources

Information sources have eight severity levels.

Table 1-1

Level	Level Name	Description
1	emergencies	Fatal errors
2	alerts	Errors that must be corrected immediately
3	critical	Critical errors
4	errors	Errors that require attention but are not critical
5	warnings	Warnings of possible errors
6	notifications	Notices
7	informational	General information
8	debugging	Debugging information

9.5.1.3 Output Directions

Sixteen output directions are currently supported, as shown in Table 1-2:

Table 1-2

Output direction	Description
console	Console
monitor	Monitor terminal
loghost <1-8>	Log host
trapbuffer	Trap (alarm) buffer
logbuffer	Log buffer
logfile <1-4>	Log file

The output directions are configured independently of each other, but the information center must be enabled before the settings take effect.

9.5.1.4 Source Information Format

The format of the source information is as follows:

```
<priority>timestamp sysname module/level/digest:content
```

When the log prefix is configured, the angle brackets (<>), spaces, slashes (/), and colon (:) in the format above are required. The angle brackets are valid only when the output direction is the log host.

The following is an example of a log output to the log host:

```
<188>Sep 28 15:33:46:235 2005 MyDevice SHELL/5/LOGIN: Console login from con0
```

The following is an example of a log without a prefix:

```
Console login from con0
```

Each field is described below:

1) Priority

The priority is calculated as follows: facility*8+severity-1.

The facility can be local0 to local7 (values 16 to 23). The default is local7.

The severity ranges from 1 to 8. For details about severity, see Table 1-1.

There is no character between the priority and the timestamp. The priority is valid only when the information is sent to the log host.

The origin of the facility is worth noting. The facility is an attribute used only when information is output to the log host and has no meaning for other output directions. Its values, local0 to local7, come from the log types that the early syslog protocol defined for UNIX log hosts. All types are listed below:

Table 1-3

facility	code
kernel messages	0
user-level messages	1
mail system	2
system daemons	3
security/authorization messages (note 1)	4
messages generated internally by syslogd	5
line printer subsystem	6
network news subsystem	7
UUCP subsystem	8
clock daemon (note 2)	9
security/authorization messages (note 1)	10
FTP daemon	11
NTP subsystem	12
log audit (note 1)	13
log alert (note 1)	14
clock daemon (note 2)	15
local use 0 (local0)	16
local use 1 (local1)	17
local use 2 (local2)	18
local use 3 (local3)	19
local use 4 (local4)	20
local use 5 (local5)	21

local use 6 (local6)	22
local use 7 (local7)	23

As shown, local0 to local7 are reserved for local use, with values 16 to 23.

2) Timestamp

The timestamp records the time when the system information is generated, making it easy to view and locate system events. The timestamp and the host name are separated by a space.

Date format

mmm + space + dd + space + hh + colon +mm + colon + ss + colon + sss+ space+ yyyy

mmm = Jan | Feb | Mar | Apr | May | Jun | Jul | Aug | Sep | Oct | Nov | Dec

Indicates the month.

dd = 1 to 31, indicates the day. Values 1 to 9 are padded with a leading 0; for example, 1 becomes "01".

hh = 0 to 23, indicates the hour.

mm = 0 to 59, indicates the minute.

ss = 0 to 59, indicates the second.

sss = 0 to 999, indicates the millisecond.

yyyy = 1970 to 9999, indicates the year.

3) Sysname

The sysname is the system name of the device.
The sysname and the module name are separated by a space.

4) Module

This field indicates the name of the module that generated the message.
The module and the level are separated by a slash (/).

5) Severity

System information falls into three categories: log, debug, and alarm. Each category is divided into eight levels according to urgency. The smaller the level value, the higher the urgency.

The system filters information by blocking the output of information whose level value is greater than the threshold.

When the threshold level is 1, the system outputs only emergencies information.

When the threshold level is 8, the system outputs information of all levels.

If exact matching is configured, the system outputs only information at the configured level.

The level is separated from the digest by a slash (/).

6) Digest

The digest is a string of up to 32 characters that summarizes the content of the message.

The digest is separated from the content by a colon (:).

7) Content

The content field indicates the specific content of the system information.

9.5.2 Configuration Sequence

1. Enable the information center globally
2. Enable the output directions of the information center
3. Enable or disable the log information prefix
4. Set the log information matching conditions

5. Configure the IP address and facility of the log host
6. Configure the log file capacity and storage path
7. Configure the recording of user operation commands
8. Delete logs recorded in the logbuffer or trapbuffer
9. View the storage areas that support saving files
10. Perform one-key collection

1. Enable the information center globally

Command	Explanation
Global Mode	
info-center enable	Enables the information center globally.
no info-center enable	Disables the information center globally.

2. Enable the output directions of the information center

Command	Explanation
Global Mode	
info-center (console | logbuffer | monitor | trapbuffer) output-enable info-center (logfile <1-4> | loghost <1-8>) output-enable (member <1-16> |)	Enables the specified output direction of the information center.
no info-center (console | logbuffer | monitor | trapbuffer) output-enable no info-center (logfile <1-4> | loghost <1-8>) output-enable (member <1-16> |)	Disables the specified output direction of the information center.

3. Enable or disable the log information prefix

Command	Explanation
Global Mode	
info-center (console | logbuffer | monitor | trapbuffer) prefix (on|off) info-center (logfile <1-4> | loghost <1-8>) prefix (on|off) (member <1-16> |)	Enables (on) or disables (off) the log information prefix.

4. Set the log information matching conditions

Command	Explanation
Global Mode	
info-center (console | logbuffer | monitor | trapbuffer) match level (emergencies | alerts | critical | errors | warnings | notifications | informational | debugging) (exact |) (keyword WORD |) info-center (logfile <1-4> | loghost <1-8>) match level (emergencies | alerts | critical | errors | warnings | notifications | informational | debugging) (exact |) (keyword WORD |) (member <1-16> |)	Sets the log information matching conditions.
no info-center (console | logbuffer | monitor | trapbuffer) match no info-center (logfile <1-4> | loghost <1-8>) match (member <1-16> |)	Deletes the log information matching conditions.

5. Configure the IP address and facility of the log host

Command	Explanation
Global Mode	

info-center loghost <1-8> config (A.B.C.D | X:X::X:X) facility (local0|local1|local2|local3|local4|local5|local6|local7) (member <1-16> |)	Configures the IP address and facility of the log host.
no info-center loghost <1-8> config (member <1-16> |)	Deletes the IP address and facility of the log host.

6. Configure the log file capacity and storage path

Command	Explanation
Global Mode	
info-center logfile <1-4> config count <1-40960> (flash|usb|mmc) WORD (member <1-16> |)	Configures the maximum number of logs in the log file and the storage path.
no info-center logfile <1-4> config (member <1-16> |)	Deletes the log file capacity and storage path configuration.

7. Configure the recording of user operation commands

Command	Explanation
Global Mode	
info-center (logbuffer | logfile <1-4> | loghost <1-8>) record-cmd	Enables recording of user operation commands.
no info-center (logbuffer | logfile <1-4> | loghost <1-8>) record-cmd	Disables recording of user operation commands.

8. Delete logs recorded in the logbuffer or trapbuffer

Command	Explanation
Global Mode	
info-center clear trapbuffer	Deletes the logs recorded in the trapbuffer.
info-center clear logbuffer (member <1-16> |)	Deletes the logs recorded in the logbuffer.

9. View the storage areas that support saving files

Command	Explanation
Global Mode	
info-center list all disk	Displays the storage areas that support saving files.

10. Perform one-key collection

Command	Explanation
Global Mode	
info-center save all (((flash|usb|mmc) WORD)|)	Performs one-key collection of information center data.

9.5.3 Configuration Examples

9.5.3.1 Console Output Log

Example 1: Enable the console to output logs with a prefix at the debugging level or higher.

The configuration steps are as follows:

```
Switch(config)#info-center console output-enable
Switch(config)#info-center console prefix on
Switch(config)#info-center console match level debugging
```

9.5.3.2 Monitor Output Log

Example 2: Enable the monitor to output logs without a prefix at the warnings level only.

The configuration steps are as follows:

```
Switch(config)#info-center monitor output-enable
Switch(config)#info-center monitor prefix off
Switch(config)#info-center monitor match level warnings exact
```

9.5.3.3 Logbuffer Output Log

Example 3: Enable the logbuffer to output logs with a prefix at the debugging level or higher that match the regular expression DEVSM.

The configuration steps are as follows:

```
Switch(config)#info-center logbuffer output-enable
Switch(config)#info-center logbuffer prefix on
Switch(config)#info-center logbuffer match level debugging keyword DEVSM
```

9.5.3.4 Trapbuffer Output Log

Example 4: Enable the trapbuffer to output logs with a prefix at the warnings level only that match the regular expression DEVSM.

The configuration steps are as follows:

```
Switch(config)#info-center trapbuffer output-enable
Switch(config)#info-center trapbuffer prefix on
Switch(config)#info-center trapbuffer match level warnings exact keyword
DEVSM
```

9.5.3.5 Loghost Output Log

Example 5: Enable log host 1 to output logs with a prefix at the debugging level or higher. The IP address of the log host is 192.168.1.11 and the facility is local0.

The configuration steps in VSF mode are as follows:

```
Switch(config)#info-center loghost 1 output-enable member 1
Switch(config)#info-center loghost 1 prefix on member 1
Switch(config)#info-center loghost 1 match level debugging member 1
Switch(config)#info-center loghost 1 config 192.168.1.11 facility local0
member 1
```

The configuration steps in stand-alone mode are as follows:

```
Switch(config)#info-center loghost 1 output-enable
Switch(config)#info-center loghost 1 prefix on
Switch(config)#info-center loghost 1 match level debugging
Switch(config)#info-center loghost 1 config 192.168.1.11 facility local0
```

9.5.3.6 Logfile Output Log

Example 6: Enable logfile 1 to output logs with a prefix at the debugging level or higher, with up to 2000 logs. The storage disk is flash and the file name is file1.log.

The configuration steps in VSF mode are as follows:

```
Switch(config)#info-center logfile 1 output-enable member 1
Switch(config)#info-center logfile 1 prefix on member 1
Switch(config)#info-center logfile 1 match level debugging member 1
Switch(config)#info-center logfile 1 config count 2000 flash file1.log member
1
```

9.5.3.7 Record User Operation Commands

Example 7: Enable the logbuffer to record user operation commands.

The configuration steps are as follows:

```
Switch(config)#info-center logbuffer output-enable
Switch(config)#info-center logbuffer record-cmd
```

9.5.3.8 Delete Logs Recorded in the Logbuffer or Trapbuffer

Example 8: Delete the logs recorded in the logbuffer or trapbuffer.

The configuration steps are as follows:

```
Switch(config)#info-center clear logbuffer
Switch(config)#info-center clear trapbuffer
```

9.5.3.9 View the Storage Areas That Support Saving Files

Example 9: View the storage areas that support saving files.

The configuration steps are as follows:

```
Switch(config)#info-center list all disk
flash:
usb:
mmc:
```

9.5.3.10 Perform One-Key Collection

Example 10: Perform one-key collection.

The configuration steps are as follows:

```
Switch(config)#info-center list all disk
*****Now saving Master card(member 1)*****
Now saving infocenter all configuration, please wait..
Now saving infocenter logbuffer content, please wait..
Now saving infocenter trapbuffer content, please wait..
*****Master card(member 1) saving finished!*****
*****Now saving Slave card(member 2)*****
Now saving infocenter logbuffer content, please wait..
*****Slave card(member 2) saving finished!*****
```

9.5.4 Notes

- A configured matching level applies only to subsequent logs; previously saved log information remains unchanged.
- Disk capacity is limited. When the maximum number of logs on the disk is reached, new logs replace the oldest ones.
- The information center is enabled globally by default. Use the show info-center config command to display all current information center configurations.

9.6 Mirror

9.6.1 Introduction to Mirror

Mirror functions include port mirroring and flow mirroring.

Port mirroring copies the data frames sent or received on one port to another port. The port whose traffic is copied is the mirror source port, and the port that receives the copies is the mirror destination port. A protocol analyzer (such as Sniffer) or an RMON monitor is connected to the mirror destination port to monitor and manage the network and diagnose network problems.

Flow mirroring copies the data frames received on a port that match a specified rule to another port. Flow mirroring takes effect only when the specified rule is permit.

A chassis switch supports up to 4 mirror destination ports, and each line card allows one source or destination port per mirror session. A box switch supports multiple mirror sessions. The number of mirror source ports is not limited; one or more ports can be configured. Multiple source ports can be in the same VLAN or in different VLANs, and the source and destination ports can be in different VLANs.

9.6.2 Mirror Configuration Task List

1. Specify the mirror destination port
2. Specify the mirror source port
3. Specify the flow mirror source

1. Specify the mirror destination port

Command	Explanation
Global Mode	
monitor session <*session*> destination interface <*interface-number*> **no monitor session <*session*> destination interface <*interface-number*>**	Specifies the mirror destination port. The no command deletes the mirror destination port.

2. Specify the mirror source port

Command	Explanation
Global Mode	
monitor session <*session*> source {interface <*interface-list*>} {rx| tx| both} **no monitor session <*session*> source {interface <*interface-list*>}**	Specifies the mirror source port. The no command deletes the mirror source port.

3. Specify the flow mirror source

Specifies the flow mirror source port and applies the rule. The no command deletes the flow mirror source port.

9.6.3 Mirror Examples

☞ **Example:**

Requirement: On interface 1/0/1, monitor the data frames sent by interface 1/0/9, the data frames received on interface 1/0/7, and the data frames received on interface 1/0/15 that match rule 120 (source IP address 1.2.3.4 and destination IP address 5.6.7.8).

```
Configuration guidelines:
1. Configure interface 1/0/1 to be a mirror destination interface.
2. Configure the interface 1/0/7 ingress and interface 1/0/9 egress to be
mirrored source.
3. Configure access list 120.
4. Configure access 120 to binding interface 15 ingress.
```

The configuration procedure is as follows:

```
Switch(config)#monitor session 4 destination interface ethernet 1/0/1
Switch(config)#monitor session 4 source interface ethernet 1/0/7 rx
Switch(config)#monitor session 4 source interface ethernet 1/0/9 tx
Switch(config)#access-list 120 permit tcp 1.2.3.4 0.0.0.255 5.6.7.8 0.0.0.255
Switch(config)#monitor session 4 source interface ethernet 1/0/15 access-list
120 rx
```

9.6.4 Mirror Troubleshooting

If problems occur when configuring port mirroring, check the following possible causes first:

- ☞ Check whether the mirror destination port is a member of a trunk (port-channel) group. If it is, modify the trunk group.
- ☞ If the throughput of the mirror destination port is lower than the total throughput of the mirror source ports, the destination port cannot copy all source port traffic. Reduce the number of source ports, mirror only one direction, or choose a port with higher throughput as the destination port. Do not add the mirror destination port to an isolate VLAN; otherwise, mirroring between VLANs is affected.

9.7 RSPAN

9.7.1 Introduction to RSPAN

Port mirroring copies the data frames sent or received on one port to another port. The port whose traffic is copied is the mirror source port, and the port that receives the copies is the mirror destination port. Port mirroring makes it easier for network administrators to monitor and manage the network and to diagnose problems. However, it applies only when the mirror source port and the mirror destination port are on the same switch.

RSPAN (Remote Switched Port Analyzer) is remote port mirroring. It removes the restriction that the source port and the destination port must be on the same switch, allowing them to be on different devices in the network and making it easier for network administrators to manage remote switches. Traffic cannot be forwarded in the remote mirror VLAN.

There are three types of switches in an RSPAN deployment:

1. Source switch: The switch to which the monitored port belongs. The source switch copies the mirrored traffic to the remote VLAN, and the mirrored traffic is then sent through Layer 2 forwarding to an intermediate switch or the destination switch.

2. Intermediate switch: A switch between the source switch and the destination switch. The intermediate switch forwards mirrored traffic to the next intermediate switch or to the destination switch. If the source and destination switches are directly connected, there is no intermediate switch.

3. Destination switch: The switch to which the remote mirroring destination port belongs. It forwards the mirrored traffic received from the remote VLAN to the monitoring device through the destination port.

When configuring RSPAN on the source switch, you can use either reflector port mode or destination mirror port mode. The destination switch redirects all data frames in the RSPAN VLAN to the RSPAN destination port. RSPAN supports normal mode and advanced mode. Normal mode is the default and suits typical users; advanced mode suits advanced users.

1. Advanced mode: Data frames in the RSPAN VLAN are redirected to the RSPAN destination port. The intermediate and destination devices must support flow redirection.

2. Normal mode: The RSPAN destination port is configured in the RSPAN VLAN, so packets in the RSPAN VLAN are broadcast to the destination port. In this mode, the destination port must be in the RSPAN VLAN, and broadcast storm control must not be configured on the source port. Configure trunk ports carefully so that RSPAN packets are not forwarded to external networks. Normal mode is easy to configure and uses fewer system resources.

Note: Normal mode is the default. In normal mode, packets with reserved MAC addresses cannot be broadcast.

The number of mirror source ports is not limited; there can be one or more. Multiple source ports do not have to be in the same VLAN, and the destination port and the source ports can be in different VLANs.

To configure RSPAN, first configure a dedicated RSPAN VLAN to carry the RSPAN packets. The default VLAN, dynamic VLANs, private VLANs, multicast VLANs, and VLANs with a Layer 3 interface enabled cannot be configured as the RSPAN VLAN. The reflector port must belong to the RSPAN VLAN. The destination port must be connected to the monitoring device and configured as an access port or a trunk port. An RSPAN reflector port is dedicated to mirroring: when a port is configured as a reflector port, it drops all existing connections to the remote peer, disables loopback-related configurations, and stops forwarding packets. Make sure that the remote VLAN provides connectivity between the source and destination switches.

Note:

1. Do not configure Layer 3 interfaces for the RSPAN VLAN on the source, intermediate, or destination switches; otherwise, the mirrored packets may be discarded.

2. On the source and intermediate switches, the native VLAN of a trunk port cannot be the RSPAN VLAN; otherwise, the RSPAN tag is removed before the packets reach the destination switch.

3. In advanced RSPAN mode, do not add the source port (in access or trunk mode) to the RSPAN VLAN.

4. When configuring remote mirroring, make sure the network bandwidth is sufficient to carry both the normal traffic and the mirrored traffic.

Keywords:

RSPAN: Remote Switched Port Analyzer.

RSPAN VLAN: Dedicated VLAN for RSPAN.

RSPAN Tag: The VLAN tag attached to the RSPAN packets at the MTP.

Reflector Port: The local mirror port between the RSPAN source and destination ports, which is not directly connected to the intermediate switch.

9.7.2 RSPAN Configuration Task List

1. Configure the RSPAN VLAN
2. Configure the mirror source port
3. Configure the mirror destination port
4. Configure the reflector port
5. Configure the remote VLAN of the mirror session

1. Configure the RSPAN VLAN

Command	Explanation
VLAN Configuration Mode	
remote-span **no remote-span**	Configures the specified VLAN as the RSPAN VLAN. The no command removes the RSPAN VLAN configuration.

2. Configure the mirror source port

Command	Explanation
Global Mode	
monitor session <*session*> source {interface <*interface-list*>} {rx| tx| both} **no monitor session <*session*> source {interface <*interface-list*>}**	Configures the mirror source port. The no command deletes the mirror source port.

3. Configure the mirror destination port

Command	Explanation
Global Mode	

monitor session <*session*> destination interface <*interface-number*> **no monitor session <*session*> destination interface <*interface-number*>**	Configures the mirror destination port. The no command deletes the mirror destination port.

4. Configure the reflector port

Command	Explanation
Global Mode	
monitor session <session> reflector-port <interface-number> **no monitor session <session> reflector-port**	Configures the interface as the reflector port. The no command deletes the reflector port.

5. Configure the remote VLAN of the mirror session

Command	Explanation
Global Mode	
monitor session <*session*> remote vlan <*vid*> **no monitor session <session> remote vlan**	Configures the remote VLAN of the mirror session. The no command deletes the remote VLAN of the mirror session.

9.7.3 Typical Examples of RSPAN

Before RSPAN, network administrators had to connect their PCs directly to the switches to check network statistics.

With RSPAN, network administrators can configure and monitor switches remotely, which is more efficient. The figure below shows a sample RSPAN application.

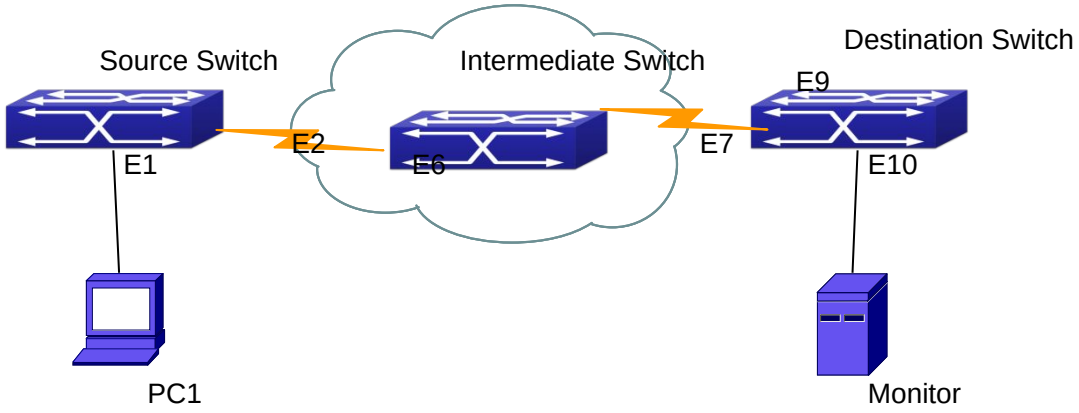


Figure 9-3 RSPAN Application Sample

RSPAN supports two configuration solutions: without a reflector port and with a reflector port. In the first solution, only one fixed port can connect to the intermediate switch, but no reflector port needs to be configured, which maximizes the use of switch ports. In the second solution, the port connected to the intermediate switch is not fixed, and packets are broadcast in the RSPAN VLAN through the loopback, which is more flexible.

The normal mode configuration is shown below:

Solution 1:

Source switch:

Interface ethernet 1/0/1 is the mirror source port.

Interface ethernet 1/0/2 is the destination port, which is connected to the intermediate switch.

RSPAN VLAN is 5.

```
Switch(config)#vlan 5
Switch(Config-Vlan5)#remote-span
Switch(Config-Vlan5)#exit
Switch(config)#interface ethernet 1/0/2
Switch(Config-If-Ethernet1/0/2)#switchport mode trunk
Switch(Config-If-Ethernet1/0/2)#exit
Switch(config)#monitor session 1 source interface ethernet1/0/1 rx
Switch(config)#monitor session 1 destination interface ethernet1/0/2
Switch(config)#monitor session 1 remote vlan 5
```

Intermediate switch:

Interface ethernet1/0/6 is the source port, which is connected to the source switch.

Interface ethernet1/0/7 is the destination port, which is connected to the destination switch. The native VLAN of this port cannot be the RSPAN VLAN; otherwise, the mirrored data may not reach the destination switch.

RSPAN VLAN is 5.

```
Switch(config)#vlan 5
Switch(Config-Vlan5)#remote-span
Switch(Config-Vlan5)#exit
Switch(config)#interface ethernet 1/0/6-7
Switch(Config-If-Port-Range)#switchport mode trunk
Switch(Config-If-Port-Range)#exit
```

Destination switch:

Interface ethernet1/0/9 is the source port, which is connected to the source switch.

Interface ethernet1/0/10 is the destination port, which is connected to the monitoring device. This port must be configured as an access port and belong to the RSPAN VLAN.

RSPAN VLAN is 5.

```
Switch(config)#vlan 5
Switch(Config-Vlan5)#remote-span
Switch(Config-Vlan5)#exit
Switch(config)#interface ethernet 1/0/9
Switch(Config-If-Ethernet1/0/9)#switchport mode trunk
Switch(Config-If-Ethernet1/0/9)#exit
Switch(config)#interface ethernet 1/0/10
Switch(Config-If-Ethernet1/0/10)#switchport access vlan 5
Switch(Config-If-Ethernet1/0/10)#exit
```

Solution 2:

Source switch:

Interface ethernet 1/0/1 is the source port.

Interface ethernet 1/0/2 is a trunk port connected to the intermediate switch. Its native VLAN must not be the RSPAN VLAN.

Interface Ethernet 1/0/3 is the reflector port. The reflector port belongs to the RSPAN VLAN and is an access port or trunk port of the RSPAN VLAN.

RSPAN VLAN is 5.

```
Switch(config)#vlan 5
Switch(Config-Vlan5)#remote-span
Switch(Config-Vlan5)#exit
Switch(config)#interface ethernet1/0/2
Switch(Config-If-Ethernet1/0/2)#switchport mode trunk
Switch(Config-If-Ethernet1/0/2)#exit
Switch(config)#interface ethernet 1/0/3
Switch(Config-If-Ethernet1/0/3)#switchport mode trunk
```

```
Switch(Config-If-Ethernet1/0/3)#exit
Switch(config)#monitor session 1 source interface ethernet1/0/1 rx
Switch(config)#monitor session 1 reflector-port ethernet1/0/3
Switch(config)#monitor session 1 remote vlan 5
```

Intermediate switch:

Interface ethernet1/0/6 is the source port, which is connected to the source switch.

Interface ethernet1/0/7 is the destination port, which is connected to the destination switch. The native VLAN of this port must not be the RSPAN VLAN; otherwise, the mirrored data may not reach the destination switch.

RSPAN VLAN is 5.

```
Switch(config)#vlan 5
Switch(Config-Vlan5)#remote-span
Switch(Config-Vlan5)#exit
Switch(config)#interface ethernet 1/0/6-7
Switch(Config-If-Port-Range)#switchport mode trunk
Switch(Config-If-Port-Range)#exit
```

Destination switch:

Interface ethernet1/0/9 is the source port, which is connected to the source switch.

Interface ethernet1/0/10 is the destination port, which is connected to the monitoring device. This port must be configured as an access port and belong to the RSPAN VLAN.

RSPAN VLAN is 5.

```
Switch(config)#vlan 5
Switch(Config-Vlan5)#remote-span
Switch(Config-Vlan5)#exit
Switch(config)#interface ethernet 1/0/9
Switch(Config-If-Ethernet1/0/9)#switchport mode trunk
Switch(Config-If-Ethernet1/0/9)#exit
Switch(config)#interface ethernet 1/0/10
Switch(Config-If-Ethernet1/0/10)#switchport access vlan 5
Switch(Config-If-Ethernet1/0/10)#exit
```

9.7.4 RSPAN Troubleshooting

RSPAN may not work for the following reasons:

- ☞ The destination mirror port is a member of a port-channel group. If so, change the port-channel group configuration.
- ☞ The throughput of the destination port is lower than the total throughput of the source mirror ports. In this case, the destination port cannot capture all packets from the source ports. Reduce the number of source ports, mirror traffic in one direction only, or choose a port with higher capacity as the destination port.
- ☞ The native VLAN of the trunk ports between the source switch and the intermediate switch is configured as the RSPAN VLAN. If so, change the native VLAN of the trunk ports.
- ☞ After RSPAN is configured, a VLAN tag is added to egress-mirrored packets, which can cause error frames on the reflector port. In this case, modify the default MTU value of the switch.

9.8 ERSPAN

9.8.1 Introduction to ERSPAN

ERSPAN (Encapsulated Remote Switched Port Analyzer) removes the restriction that the source port and the destination port must be on the same switch, allowing them to be on different devices in the network and making it easier for network administrators to manage remote switches. Compared with

traditional RSPAN, ERSPAN is simpler to configure and transmits the monitored traffic through a specified tunnel.

Note:

1. ERSPAN currently supports only port mirroring as the monitor source; CPU mirroring and flow mirroring are not supported.
2. A tunnel must exist between the source and destination switches of the ERSPAN connection.
3. When configuring remote mirroring, make sure the network bandwidth is sufficient to carry both the normal traffic and the mirrored traffic.

Keywords:

ERSPAN: Encapsulated Remote Switched Port Analyzer.

9.8.2 ERSPAN Configuration Task List

1. Specify the mirror source port

Command	Explanation
Global Mode	
monitor session <*session*> source {interface <*interface-list*>} {rx| tx| both} **no monitor session <*session*> source {interface <*interface-list*>}**	Specifies the mirror source port. The no command deletes the mirror source port.

2. Specify the mirror destination tunnel

Command	Explanation
Global Mode	
monitor session <*session*> destination tunnel <*tunnel-number*> reflector-port <interface-number> **no monitor session <*session*> destination tunnel <*tunnel-number*> reflector-port <interface-number>**	Specifies the mirror destination tunnel. The no command deletes the mirror destination tunnel.

9.8.3 Typical Examples of ERSPAN

Before ERSPAN, network administrators had to connect their PCs directly to the switches to check the network status.

With ERSPAN, network administrators can configure and monitor switches remotely, which is more efficient.

In a Layer 3 network, Device A connects to the marketing department through Ethernet 1/0/1 and to Ethernet1/0/2 of Device B through Ethernet 1/0/2. Device C connects to the server through Ethernet 1/0/2 and to Ethernet 1/0/1 of Device B through Ethernet 1/0/1. With remote port mirroring configured, the server can monitor the bidirectional traffic of the marketing department across a GRE tunnel. The figure below shows a sample ERSPAN application.

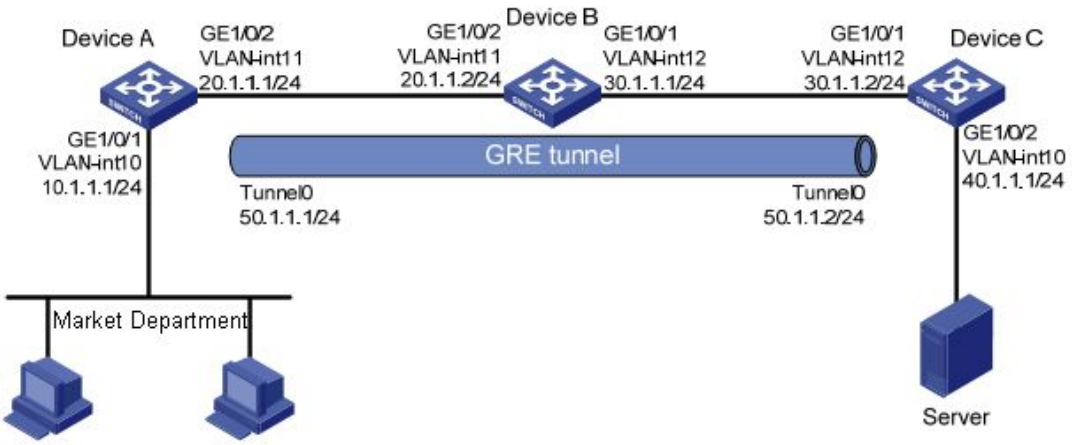


Figure 9-4 ERSPAN application diagram

Before configuring Layer 3 remote port mirroring, make sure a GRE tunnel connecting the source and destination devices has been created and is forwarding traffic normally.

Layer 3 remote port mirroring is configured on the source and destination devices separately. Source and destination ports are configured on both devices, with the following differences:

1) On Device A, configure the port to be monitored as the source port and the tunnel interface as the destination port.

2) On Device C, configure the physical port corresponding to the tunnel interface as the source port and the port connected to the monitoring device as the destination port.

(1) Configure IP addresses

Configure the IP address and subnet mask of each interface. The procedure is omitted here.

(2) Configure Device A (the source device)

Create interface Tunnel1, and configure an IP address and mask for it.

```
SwitchA(config)#interface tunnel 1
SwitchA(config-if-tunnel1)# tunnel mode gre ip
SwitchA (config-if-tunnel1)#ip address 50.1.1.1 255.255.255.0
```

Configure Tunnel1 to operate in GRE tunnel mode, and configure its source and destination IP addresses.

```
SwitchA(config-if-tunnel1)# tunnel source 10.1.1.1
SwitchA(config-if-tunnel1)# tunnel destination 40.1.1.1
SwitchA(config-if-tunnel1)# exit
```

Configure OSPF protocol.

```
SwitchA (config)#router ospf
SwitchA (config-router)#network 0.0.0.0/0 area 0
SwitchA (config-router)#exit
```

Configure Ethernet 1/0/1 as the source port and Tunnel1 as the destination port of local mirroring group 1, and Ethernet1/0/3 as the reflector port.

```
SwitchA(config)#monitor session 4 destination tunnel 1 reflector-port
ethernet 1/0/3
SwitchA(config)#monitor session 4 source interface ethernet 1/0/1 both
```

(3) Configure Device B (the intermediate device)

Configure OSPF protocol.

```
SwitchB (config)#router ospf
SwitchB (config-router)#network 0.0.0.0/0 area 0
SwitchB (config-router)#exit
```

(4) Configure Device C (the destination device)

Create interface Tunnel1, and configure an IP address and mask for it.

```
SwitchC(config)#interface tunnel 1
SwitchC (config-if-tunnel1)# tunnel mode gre ip
SwitchC (config-if-tunnel1)#ip address 50.1.1.2 255.255.255.0
```

Configure Tunnel1 to operate in GRE tunnel mode, and configure its source and destination IP addresses.

```
SwitchC (config-if-tunnel1)# tunnel source 40.1.1.1
SwitchC (config-if-tunnel1)# tunnel destination 10.1.1.1
SwitchC (config-if-tunnel1)# exit
```

Configure OSPF protocol.

```
SwitchC (config)#router ospf
SwitchC (config-router)#network 0.0.0.0/0 area 0
SwitchC (config-router)#exit
```

Configure Ethernet 1/0/1 as a source port and Ethernet 1/0/2 as the destination port of local mirroring group 1.

```
SwitchC (config)#monitor session 1 destination interface ethernet 1/0/2
SwitchC (config)#monitor session 1 source interface ethernet 1/0/1 rx
```

9.8.4 ERSPAN Troubleshooting

If problems occur when configuring ERSPAN, check whether they are caused by the following:

- Check the GRE tunnel configuration to make sure traffic is transmitted normally.
- If the throughput of the mirror destination port is lower than the total throughput of the mirror source ports, the destination port cannot copy all source port traffic. Reduce the number of source ports, mirror only one direction, or choose a port with higher throughput as the destination port.

9.9 sFlow

9.9.1 Introduction to sFlow

sFlow (RFC 3176), developed by InMon Corporation, is a standards-based protocol for monitoring network traffic. The monitored switch or router samples traffic and collects statistics, and sends the data to the analyzer, which analyzes it according to user requirements to monitor the network.

An sFlow monitoring system consists of an sFlow agent, a central data collector, and an sFlow analyzer. The sFlow agent collects data from the switch through sampling. The sFlow collector formats the sampled data and statistics and forwards them to the sFlow analyzer, which analyzes the sampled data and takes action based on the results. The switch acts as both the agent and the central data collector in the sFlow system.

Data sampling and statistics are supported on physical ports.

Sampled data includes IPv4 and IPv6 packets; other extension types are not currently supported. For non-IPv4 and non-IPv6 packets, the generic HEADER mode defined in RFC 3176 is used, and the packet header is copied based on the protocol type of the packet.

The latest sFlow version released by InMon is version 5, whereas RFC 3176 describes version 4, so conflicts may exist in some areas, such as structure and packet format. Because version 5 is not yet an official standard, the switch continues to follow RFC 3176 for compatibility with current applications.

9.9.2 sFlow Configuration Task List

1. Configure the sFlow collector address

Command	Explanation
Global Mode and Port Mode	
sflow destination <*collector-address*> [<*collector-port*>] **no sflow destination**	Configures the IP address and port number of the host on which the sFlow analysis software is installed. If an address is configured on a port, the port configuration applies; otherwise, the global configuration applies. The "**no sflow destination**" command restores the default port number and deletes the IP address.

2. Configure the sFlow agent address

Command	Explanation
Global Mode	
sflow agent-address <*collector-address*> **no sflow agent-address**	Configures the source IP address used by the sFlow agent. The no command deletes this address.

3. Configure the sFlow agent priority

Command	Explanation
Global Mode	
sflow priority <*priority-vlaue*> **no sflow priority**	Configures the priority with which sFlow receives packets from the hardware. The "**no sflow priority**" command restores the default priority.

4. Configure the packet header length copied by sFlow

Command	Explanation
Port Mode	
sflow header-len <*length-vlaue*> **no sflow header-len**	Configures the length of the packet header copied during sFlow data sampling. The no command restores the default value.

5. Configure the maximum data length of sFlow packets

Command	Explanation
Port Mode	
sflow data-len <*length-vlaue*> **no sflow data-len**	Configures the maximum length of the data in an sFlow packet. The no command restores the default value.

6. Configure the sampling rate

Command	Explanation
Port Mode	
sflow rate {input <*input-rate*> | output <*output-rate* >} **no sflow rate [input | output]**	Configures the sampling rate used by sFlow for hardware sampling. The no command deletes the rate value.

7. Configure the sFlow statistics sampling interval

Command	Explanation
Port Mode	
sflow counter-interval <*interval-vlaue*> **no sflow counter-interval**	Configures the maximum interval for sFlow statistics sampling. The no command deletes the interval.

8. Configure the analyzer used by sFlow

Command	Explanation
Global Mode	
sflow analyzer sflowtrend **no sflow analyzer sflowtrend**	Configures the analyzer used by sFlow. The no command deletes the analyzer.

9.9.3 sFlow Examples

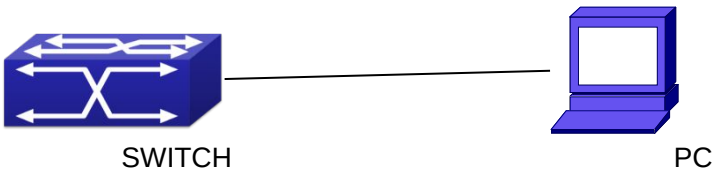


Figure 9-5 sFlow configuration topology

As shown in the figure, sFlow sampling is enabled on ports 1/0/1 and 1/0/2 of the switch. Assume the sFlow analysis software is installed on a PC with the address 192.168.1.200. The address of the Layer 3 interface on SwitchA connected to the PC is 192.168.1.100, and a loopback interface with the address 10.1.144.2 is configured on SwitchA. The sFlow configuration is as follows:

The configuration procedure is as follows:

```
Switch#config
Switch (config)#sflow ageng-address 10.1.144.2
Switch (config)#sflow destination 192.168.1.200
Switch (config)#sflow priority 1
Switch (config)# interface ethernet1/0/1
Switch (Config-If-Ethernet1/0/1)#sflow rate input 10000
Switch (Config-If-Ethernet1/0/1)#sflow rate output 10000
Switch (Config-If-Ethernet1/0/1)#sflow counter-interval 20
Switch (Config-If-Ethernet1/0/1)#exit
Switch (config)# interface ethernet1/0/2
Switch (Config-If-Ethernet1/0/2)#sflow rate input 20000
Switch (Config-If-Ethernet1/0/2)#sflow rate output 20000
Switch (Config-If-Ethernet1/0/2)#sflow counter-interval 40
```

9.9.4 sFlow Troubleshooting

When sFlow is configured and used, the sFlow server may fail to work properly because of a physical connection failure, an incorrect configuration, and so on. Make sure that:

- ☞ The physical connection is correct.
- ☞ The address of the sFlow analyzer configured in Global Mode or Port Mode is reachable.
- ☞ If traffic sampling is required, the sampling rate of the interface is configured.
- ☞ If statistics sampling is required, the statistics sampling interval of the interface is configured.
- ☞ If the problem persists, contact EnGenius technical support.

9.10 NQA

9.10.1 Introduction to NQA

1. Overview

Devices can monitor interface status, link status, network reachability, and network performance in various ways, such as NQA (Network Quality Analyzer), BFD (Bidirectional Forwarding Detection),

and interface management. The modules responsible for monitoring interface status, network reachability, and so on are called monitoring modules.

Generally, application modules, such as VRRP, static routing, RIP, OSPF, BGP, and GRE tunnels, can be associated directly with monitoring modules, which report the monitoring results directly to the application module for processing. If an application module can be associated with multiple monitoring modules, it must process the results from each monitoring module separately, because different monitoring modules report results in different forms. This increases the processing burden on application modules and slows service recovery.

The Track function adds a Track module between the application modules and the monitoring modules. The Track module hides the differences between monitoring modules, provides a unified interface to the application modules, and simplifies their processing.

2. Working Principle of Track Linkage

The Track linkage function consists of three parts: application modules, the Track module, and monitoring modules. Linkage items are established between these modules so that a monitoring module can trigger an application module, through the Track module, to perform certain operations. The monitoring module detects the link status, network performance, and so on, and notifies the application module of the results through the Track module. When the application module detects a change in network status, it promptly takes action to avoid communication interruptions or degraded service quality.

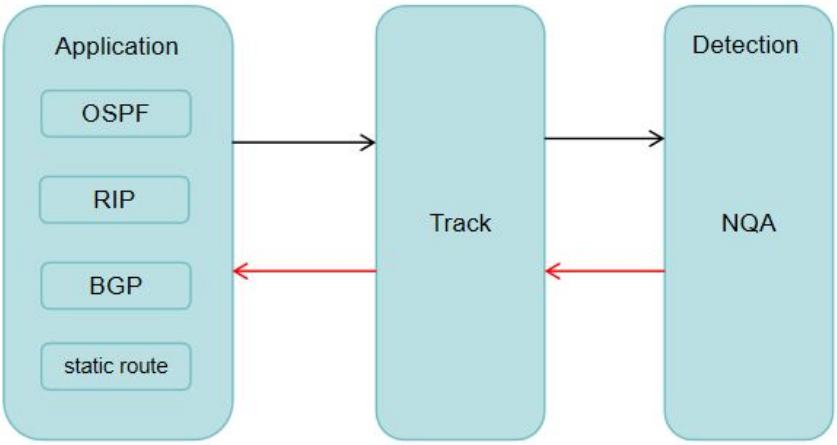


Figure 1-1 Track Linkage Function

The Track linkage function works in two parts:

Linkage between the Track module and monitoring modules

Linkage between the Track module and application modules

Linkage among the Track module, application modules, and monitoring modules

The linkage between the Track module and monitoring modules is configured through the CLI. The monitoring module that supports linkage with the Track module is:

NQA (Network Quality Analyzer)

The linkage between the Track module and application modules is configured through the CLI. The application modules that support linkage with the Track module are:

Static routing;

RIP (Routing Information Protocol);

OSPF (Open Shortest Path First);

BGP (Border Gateway Protocol).

3. Requirements Overview

The following functions support linkage with Track:

The NQA detection function includes:

※ ICMP-echo probe;

※ ICMP-jitter probe;

※ UDP-echo probe;

※ UDP-jitter probe;

※ TCP probe;

※ DHCP probe.

Association of NQA with Track items is supported.

Association of multiple monitoring methods with a single Track item is supported.

Association of the same monitoring method with multiple Track items is supported.

A Track item cannot be associated with multiple application instances; it can be associated with only one application instance.

Linkage between Track and static routing is supported.

Linkage between Track and RIP is supported.

Linkage between Track and OSPF is supported.

Linkage between Track and BGP is supported.

9.10.2 NQA Configuration Task List

Track configuration tasks are divided into monitoring module configuration tasks, Track association configuration tasks, and application module configuration tasks.

Monitoring module configuration tasks:

Enable the NQA client function

Create an NQA probe entry

Start the NQA entry probe

Configure the NQA entry description

Configure the NQA entry frequency

Configure the probe count within the NQA entry frequency

Configure the NQA entry probe timeout

Configure the NQA entry packet data size

Configure the NQA entry packet TTL

Configure the NQA entry packet ToS

Configure the NQA entry failure percentage

Configure the maximum number of NQA entry result records

Configure the NQA entry probe threshold

Enable the NQA server function

Configure the NQA server TCP listening port number

16. Configure the NQA server UDP listening port number

Track association configuration tasks:

1. Configure Track association with an NQA entry

2. Configure the Boolean relationship of Track association with multiple NQA entries

Application module configuration tasks:

Configure Track association with static routing

Configure Track association with RIP

Configure Track association with OSPF

Configure Track association with BGP

1. Monitoring module configuration tasks

1). Enable the NQA client function

Command	Explanation
Global Mode	
nqa agent **no nqa agent**	Enables the NQA client function. The no command disables the NQA client function.

2). Create an NQA probe entry

Command	Explanation
Global Mode	
nqa entry <1-64> type icmp-echo dest-ip *A.B.C.D* (src-ip *A.B.C.D* |) (nexthop-ip *A.B.C.D* |) (vrf *WORD* |) **no nqa entry <1-64>**	Sets the NQA entry probe type to icmp-echo. The no command deletes the NQA entry.
nqa entry <1-64> type icmp-jitter dest-ip *A.B.C.D* (src-ip *A.B.C.D* |) (nexthop-ip *A.B.C.D* |) (vrf *WORD* |) **no nqa entry <1-64>**	Sets the NQA entry probe type to icmp-jitter. The no command deletes the NQA entry.
nqa entry <1-64> type udp-echo dest-ip *A.B.C.D* dest-port <1-65535> (src-ip *A.B.C.D* |) (nexthop-ip *A.B.C.D* |) (vrf *WORD* |) **no nqa entry <1-64>**	Sets the NQA entry probe type to udp-echo. The no command deletes the NQA entry.
nqa entry <1-64> type udp-jitter dest-ip *A.B.C.D* dest-port <1-65535> (src-ip *A.B.C.D* |) (nexthop-ip *A.B.C.D* |) (vrf *WORD* |) **no nqa entry <1-64>**	Sets the NQA entry probe type to udp-jitter. The no command deletes the NQA entry.
nqa entry <1-64> type tcp dest-ip *A.B.C.D* dest-port <1-65535> (src-ip *A.B.C.D* |) (nexthop-ip *A.B.C.D* |) (vrf *WORD* |) **no nqa entry <1-64>**	Sets the NQA entry probe type to tcp. The no command deletes the NQA entry.
nqa entry <1-64> type dhcp interface vlan <1-4094> **no nqa entry <1-64>**	Sets the NQA entry probe type to dhcp. The no command deletes the NQA entry.

3). Start the NQA entry probe

Command	Explanation
Global Mode	
nqa entry <1-64> start (|forever|time-range *WORD*) **no nqa entry <1-64> start**	Starts the NQA entry probe. The no command restores the default start setting.

4). Configure the NQA entry description

Command	Explanation

Global Mode	
nqa entry <1-64> description *TEXT* **no nqa entry <1-64> description**	Configures the NQA entry description. The no command deletes the NQA entry description.

5). Configure the NQA entry frequency

Command	Explanation
Global Mode	
nqa entry <1-64> frequency <3-60> **no nqa entry <1-64> frequency**	Configures the NQA entry frequency. The no command restores the default frequency.

6). Configure the probe count within the NQA entry frequency

Command	Explanation
Global Mode	
nqa entry <1-64> probe-count <1-5> **no nqa entry <1-64> probe-count**	Configures the probe count within the NQA entry frequency. The no command restores the default probe count.

7). Configure the NQA entry probe timeout

Command	Explanation
Global Mode	
nqa entry <1-64> probe-timeout <1-5> **no nqa entry <1-64> probe-timeout**	Configures the NQA entry probe timeout. The no command restores the default probe timeout.

8). Configure the NQA entry packet data size

Command	Explanation
Global Mode	
nqa entry <1-64> datasize <1-8100> **no nqa entry <1-64> datasize**	Configures the NQA entry packet data size. The no command restores the default data size.

9). Configure the NQA entry packet TTL

Command	Explanation
Global Mode	
nqa entry <1-64> ttl <1-255> **no nqa entry <1-64> ttl**	Configures the NQA entry packet TTL. The no command restores the default TTL.

10). Configure the NQA entry packet ToS

Command	Explanation
Global Mode	
nqa entry <1-64> tos <0-255> **no nqa entry <1-64> tos**	Configures the NQA entry packet ToS. The no command restores the default ToS.

11). Configure the NQA entry failure percentage

Command	Explanation
Global Mode	
nqa entry <1-64> fail-percent <1-100> **no nqa entry <1-64> fail-percent**	Configures the NQA entry failure percentage. The no command restores the default failure percentage.

12). Configure the maximum number of NQA entry result records

Command	Explanation
Global Mode	

nqa entry <1-64> records-result <1-10> **no nqa entry <1-64> records-result**	Configures the maximum number of NQA entry result records. The no command restores the default maximum.

13). Configure the NQA entry probe threshold

Command	Explanation
Global Mode	
nqa entry <1-64> **probe-threshold <50-5000>** **no nqa entry <1-64> probe-threshold**	Configures the NQA entry probe threshold. The no command restores the default probe threshold.

14). Enable the NQA server function

Command	Explanation
Global Mode	
nqa server **no nqa server**	Enables the NQA server function. The no command disables the NQA server function.

15). Configure the NQA server TCP listening port number

Command	Explanation
Global Mode	
nqa server tcp-port <1-65535> **(vrf *WORD* |)** **no nqa server tcp-port (vrf *WORD* |)**	Configures the NQA server TCP listening port number. The no command disables TCP listening on the NQA server.

16). Configure the NQA server UDP listening port number

Command	Explanation
Global Mode	
nqa server udp-port <1-65535> **(vrf *WORD* |)** **no nqa server udp-port (vrf *WORD* |)**	Configures the NQA server UDP listening port number. The no command disables UDP listening on the NQA server.

2. Track association configuration tasks

1). Configure Track association with an NQA entry

Command	Explanation
Global Mode	
track <1-32> nqa <1-64> **no track <1-32> (nqa <1-64>|)**	Associates a Track item with an NQA entry. The no command removes the association with the NQA entry.

2). Configure the Boolean relationship of Track association with multiple NQA entries

Command	Explanation
Global Mode	
track <1-32> list boolean (and|or) **no track <1-32> list boolean**	Configures the Boolean relationship of a Track item associated with multiple NQA entries. The no command removes the Boolean relationship.

3. Application module configuration tasks

1). Configure Track association with static routing

Command	Explanation
Global Mode	

ip route (*A.B.C.D A.B.C.D* | *A.B.C.D/M*) *A.B.C.D* track <1-32> **no ip route (*A.B.C.D A.B.C.D* | *A.B.C.D/M*) (*A.B.C.D*|) (track <1-32>|)**	Configures a static route associated with a Track item. The no command deletes the static route.

2). Configure Track association with RIP

Command	Explanation
Interface Mode	
rip track <1-32> **no rip track <1-32>**	Associates RIP with a Track item. The no command removes the association.

3). Configure Track association with OSPF

Command	Explanation
Interface Mode	
ip ospf track <1-32> **no ip ospf track <1-32>**	Associates OSPF with a Track item. The no command removes the association.

4). Configure Track association with BGP

Command	Explanation
BGP Router Mode	
neighbor {<*ip-address*>|<*TAG*>} track <1-32> **no neighbor {<*ip-address*>|<*TAG*>} track <1-32>**	Associates a BGP neighbor with a Track item. The no command removes the association.

Chapter 10 Network Time Management Configuration

10.1 NTP

10.1.1 Introduction to NTP Function

NTP (Network Time Protocol) synchronizes time among distributed time servers and clients across WANs and LANs with millisecond-level precision. Its events, states, transmission functions, and actions are defined in RFC 1305.

NTP keeps the time of all clock-dependent devices in the network consistent, so that the devices can provide various applications based on a consistent time.

A local system running NTP can synchronize its time with other reference sources, can act as a reference source to synchronize other clocks, and can synchronize with peers by exchanging NTP packets.

10.1.2 NTP Function Configuration Task List

1. Enable the NTP function
2. Configure the NTP server
3. Configure the maximum number of broadcast or multicast servers supported by the NTP client
4. Configure the time zone
5. Configure the NTP access control list
6. Configure NTP authentication
7. Configure an interface as an NTP broadcast/multicast client interface
8. Configure an interface not to receive NTP packets
9. Configure the request packet sending interval of the NTP client
10. Display information
11. Debug

1. Enable the NTP function

Command	Explanation
Global Mode	
ntp enable **ntp disable**	Enables or disables the NTP function.

2. Configure the NTP server

Command	Explanation
Global Mode	
ntp server {<*ip-address*> | <*ipv6-address*>} [version <*version_no*>] [key <*key-id*>] **no ntp server {<*ip-address*> | <*ipv6-address*>}**	Specifies the NTP server used as the time source.

3. Configure the maximum number of broadcast or multicast servers supported by the NTP client

Command	Explanation
Global Mode	

ntp broadcast server count <number> **no ntp broadcast server count**	Sets the maximum number of broadcast or multicast servers supported by the NTP client. The no command restores the default value.

4. Configure the time zone

Command	Explanation
Global Mode	
clock timezone WORD {add | subtract} <0-23> [<0-59>] **no clock timezone WORD**	Configures the time zone. The no command deletes the configured time zone.

5. Configure the NTP access control list

Command	Explanation
Global Mode	
ntp access-group server <*acl*> **no ntp access-group server < *acl*>**	Configures the NTP server access control list.

6. Configure NTP authentication

Command	Explanation
Global Mode	
ntp authenticate **no ntp authenticate**	Enables NTP authentication.
ntp authentication-key <*key-id*> md5 <*value*> **no ntp authentication-key <*key-id*>**	Configures the authentication key for NTP authentication.
ntp trusted-key <*key-id*> **no ntp trusted-key <*key-id*>**	Configures the trusted key.

7. Configure an interface as an NTP broadcast/multicast client interface

Command	Explanation
VLAN Configuration Mode	
ntp broadcast client **no ntp broadcast client**	Configures the interface to receive NTP broadcast packets.
ntp multicast client **no ntp multicast client**	Configures the interface to receive NTP multicast packets.
ntp ipv6 multicast client **no ntp ipv6 multicast client**	Configures the interface to receive IPv6 NTP multicast packets.

8. Configure an interface not to receive NTP packets

Command	Explanation
VLAN Configuration Mode	
ntp disable **no ntp disable**	Disables the NTP function on the interface.

9. Configure the request packet sending interval of the NTP client

Command	Explanation
Global Mode	

ntp syn-interval <1-3600> **no ntp syn-interval**	Configures the request packet sending interval of the NTP client (1s to 3600s). The no command restores the default value of 64s.

10. Display information

Command	Explanation
Admin Mode	
show ntp status	Displays the time synchronization status.
show ntp session [<*ip-address*> | <*ipv6-address*>]	Displays NTP session information.

11. Debug

Command	Explanation
Admin Mode	
debug ntp authentication **no debug ntp authentication**	Enables NTP authentication debugging.
debug ntp packets [send | receive] **no debug ntp packets [send | receive]**	Enables NTP packet debugging.
debug ntp adjust **no debug ntp adjust**	Enables time adjustment debugging.
debug ntp sync **no debug ntp sync**	Enables time synchronization debugging.
debug ntp events **no debug ntp events**	Enables NTP event debugging.

10.1.3 Typical Examples of NTP Function

A client switch needs to synchronize its time with the time servers in the network. There are two time servers: one is the primary and the other is the standby. The connections and configuration are as follows (Switch A and Switch B are switches or routers that support the NTP server function):

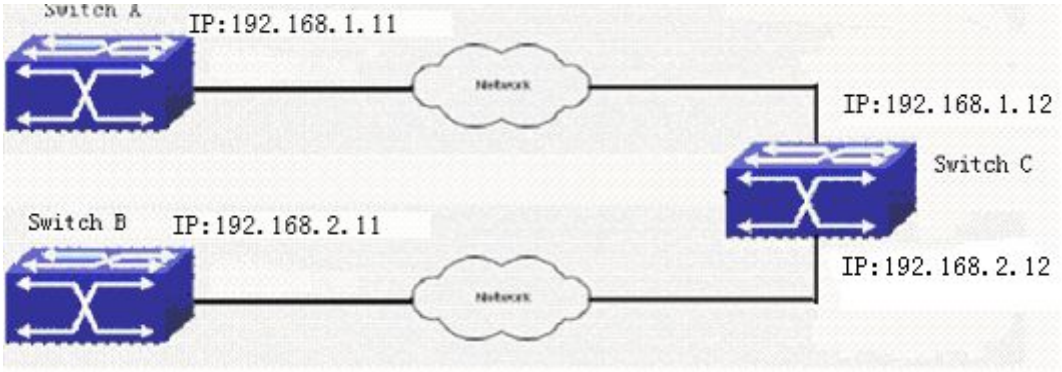


Figure 9-24

The configuration of Switch C is as follows. (The commands on Switch A and Switch B may differ by vendor and are not described here. The switch does not currently support the NTP server function.)

```
Switch C:
Switch(config)#ntp enable
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ip address 192.168.1.12 255.255.255.0
Switch(config)#interface vlan 2
Switch(Config-if-Vlan1)#ip address 192.168.2.12 255.255.255.0
Switch(config)#ntp server 192.168.1.11
Switch(config)#ntp server 192.168.2.11
```

10.1.4 NTP Function Troubleshooting

If an error occurs during configuration, the system displays debugging information.

The NTP function is disabled by default. Use the show command to display the current configuration. If the configuration is correct, use the relevant debug commands to display detailed information about the process and check whether the function is configured correctly. You can also use the show command to display NTP operating information. If the problem persists, send the recorded messages to EnGenius technical support.

10.2 SNTP

10.2.1 Introduction to SNTP

The Network Time Protocol (NTP) is widely used to synchronize the clocks of computers connected to the Internet. NTP measures the packet transmission delay in the network and independently estimates the clock offset of the computer, achieving high clock accuracy for networked computers. In most cases, NTP provides an accuracy of 1 to 50 ms, depending on the characteristics of the synchronization source and the network path.

Simple Network Time Protocol (SNTP) is a simplified version and subset of NTP that omits the complex NTP algorithms. SNTP is used by hosts that do not require full NTP functions. A common practice is to synchronize the clocks of several hosts in a LAN with other NTP hosts over the Internet, and to use those hosts to provide time synchronization to other clients in the LAN. The figure below shows an NTP/SNTP network topology, in which SNTP mainly runs between second-level servers and various terminals, because these scenarios do not require very high time accuracy and the accuracy of SNTP (1 to 50 ms) is usually sufficient.

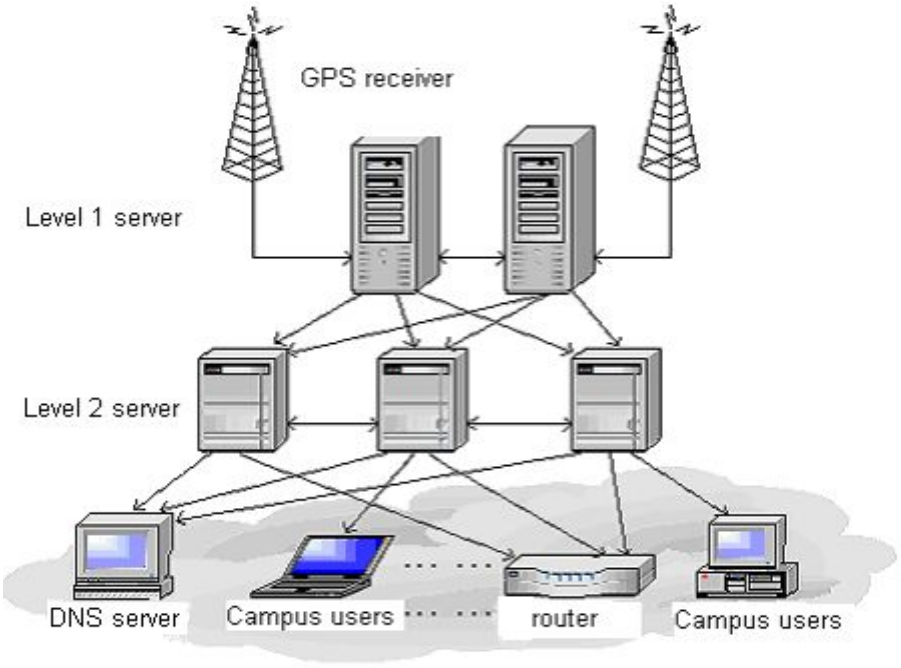


Figure 10-2 Working Scenario

The switch implements SNTPv4 and supports SNTP client unicast as described in RFC 2030. SNTP client multicast and unicast are not supported, nor is the SNTP server function.

10.2.2 Typical Examples of SNTP Configuration

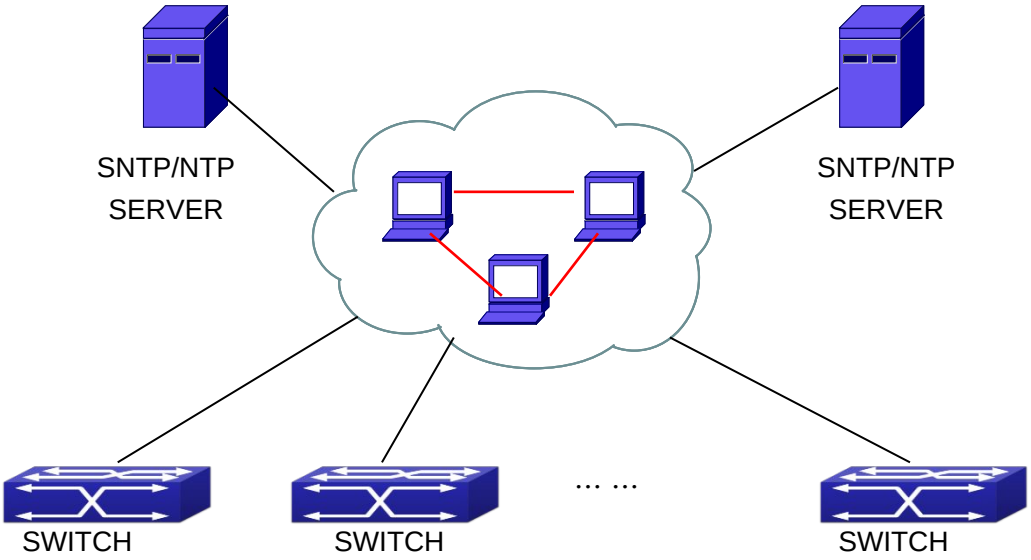


Figure 10-3 Typical SNTP Configuration

All switches in the autonomous zone must synchronize their time, which is provided by two redundant SNTP/NTP servers. For time synchronization to work, the network must be configured properly, with a reachable route between each switch and the two SNTP/NTP servers.

Example: Assume that the IP addresses of the SNTP/NTP servers are 10.1.1.1 and 20.1.1.1 and that the SNTP/NTP server function (such as NTP master) is enabled on them. The configuration on each switch is as follows:

```
Switch#config
Switch(config)#sntp server 10.1.1.1
```

10.3 DNSv4/v6

10.3.1 Introduction to DNS

DNS (Domain Name System) is a distributed database used by TCP/IP applications to translate domain names into IPv4/IPv6 addresses. With DNS, you can use easy-to-remember, meaningful domain names in applications, and the DNS server translates them into the correct IPv4/IPv6 addresses.

There are two types of DNS service, static and dynamic, which complement each other. When a name query is received, the static DNS database is checked first, and then the dynamic DNS database. Putting frequently used addresses in the static DNS database reduces the lookup time in the dynamic DNS database and improves efficiency. Static domain name resolution sets up mappings between domain names and IPv4/IPv6 addresses, so that the IPv4/IPv6 address of a domain name can be found in the static DNS database when you use an application. Dynamic domain name resolution is implemented by querying the DNS server. When you use an application with a domain name, the user program sends a name query to the resolver of the DNS client. The resolver first looks up the local domain name cache for a match. If a match is found, it returns the corresponding IPv4/IPv6 address to the switch. If no match is found, it sends a query to a higher-level DNS server. This process continues until a result, success or failure, is returned.

The Domain Name System (DNS) is a hierarchical naming system for computers, services, and other resources on the Internet. It associates various information with the domain names assigned to these participants. Most importantly, it translates human-readable domain names into the numerical (binary) identifiers of networking equipment, so that these devices can be located and addressed worldwide. DNS is often described as the "phone book" of the Internet because it translates human-friendly host names into IP addresses.

DNS makes it possible to assign domain names to groups of Internet users in a meaningful way, independent of each user's physical location. As a result, World Wide Web (WWW) hyperlinks and Internet contact information can remain consistent even if Internet routing changes or the participant uses a mobile device. Internet domain names are easier to remember than IP addresses such as 208.77.188.166 (IPv4) or 2001:db8:1f70::999:de8:7648:6e8 (IPv6). People benefit from this when they use meaningful URLs and email addresses without needing to know how the machine actually locates them.

DNS distributes the responsibility for assigning domain names and mapping them to IP networks by designating authoritative name servers for each domain to track their own changes, which avoids the need for a central register that must be continually consulted and updated.

DNS also stores other types of information, such as the list of mail servers that accept email for a given Internet domain. By providing a worldwide, distributed, keyword-based redirection service, DNS is an essential component of the Internet.

10.3.2 DNSv4/v6 Configuration Task List

1. Enable or disable the DNS function
2. Configure or delete a DNS server
3. Configure or delete a domain name suffix
4. Delete the domain entry of a specified address from the dynamic cache

5. Perform DNS dynamic domain name resolution
6. Enable or disable the DNS server function
7. Configure the maximum number of client entries in the switch queue
8. Configure the timeout for caching client information on the switch
9. Monitor and diagnose the DNS function

1. Enable or disable the DNS function

Command	Explanation
Global Mode	
ip domain-lookup **no ip domain-lookup**	Enables the DNS dynamic lookup function. The no command disables it.

2. Configure or delete a DNS server

Command	Explanation
Global Mode	
dns-server {<*ip-address*> | <*ipv6-address*>} [priority <*value*>] **no dns-server {<*ip-address*> | <*ipv6-address*>}**	Configures a DNS server. The no command deletes the DNS server.

3. Configure or delete a domain name suffix

Command	Explanation
Global Mode	
ip domain-list <*WORD*> **no ip domain-list <*WORD*>**	Configures a domain name suffix. The no command deletes it.

4. Delete the domain entry of a specified address from the dynamic cache

Command	Explanation
Admin Mode	
clear dynamic-host {<*ip-address*> | <*ipv6-address*> | all}	Deletes the domain entry of the specified address from the dynamic cache.

5. Perform DNS dynamic domain name resolution

Command	Explanation
Global Mode	
dns lookup {ipv4 | ipv6} <*hostname*>	Performs DNS dynamic domain name resolution.

6. Enable or disable the DNS server function

Command	Explanation
Global Mode	
ip dns server **no ip dns server**	Enables the DNS server function. The no command disables it.

7. Configure the maximum number of client entries in the switch queue

Command	Explanation
Global Mode	
ip dns server queue maximum <1-5000> **no ip dns server queue maximum**	Configures the maximum number of client entries in the switch queue.

8. Configure the timeout for caching client information on the switch

Command	Explanation
Global Mode	

ip dns server queue timeout <1-100> **no ip dns server queue timeout**	Configures the timeout for caching client information on the switch.

9. Monitor and diagnose the DNS function

Command	Explanation
Admin Mode and Configuration Mode	
show dns name-server	Displays the configured DNS server information.
show dns domain-list	Displays the configured DNS domain name suffix information.
show dns hosts	Displays the dynamic domain name information resolved by the switch.
show dns config	Displays the global DNS configuration on the switch.
show dns client	Displays the DNS client information maintained by the switch.
debug dns {all | packet [send | recv] | events | relay} **no debug dns {all | packet [send | recv] | events | relay}**	Enables DNS debugging. The no command disables it.

10.3.3 Typical Examples of DNS

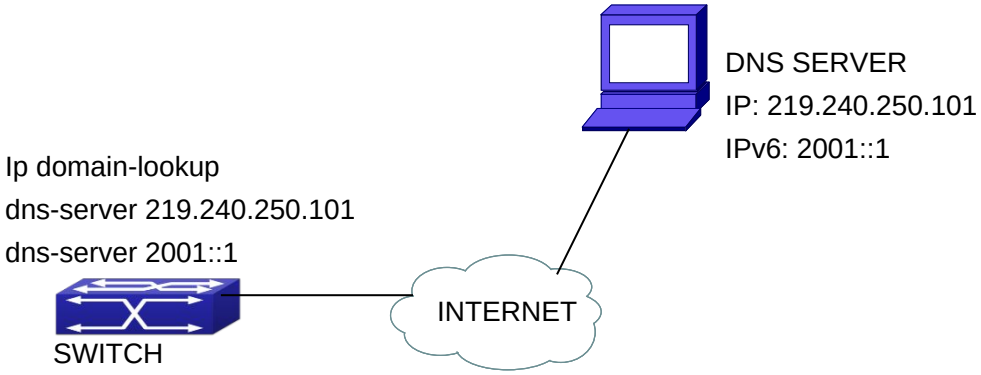


Figure 10-4 DNS CLIENT typical environment

As shown in the figure, the switch is connected to the DNS server through the network. To access the Google website, the switch does not need to know its IPv4/IPv6 address; it only needs the domain name www.google.com. The DNS server resolves the domain name into an IPv4/IPv6 address and sends it to the switch, which can then access the website. To configure the switch as a DNS client, enable DNS dynamic domain name resolution on the switch and configure the DNS server address. The switch can then obtain the IPv4/IPv6 address through dynamic domain name resolution when using tools such as ping.

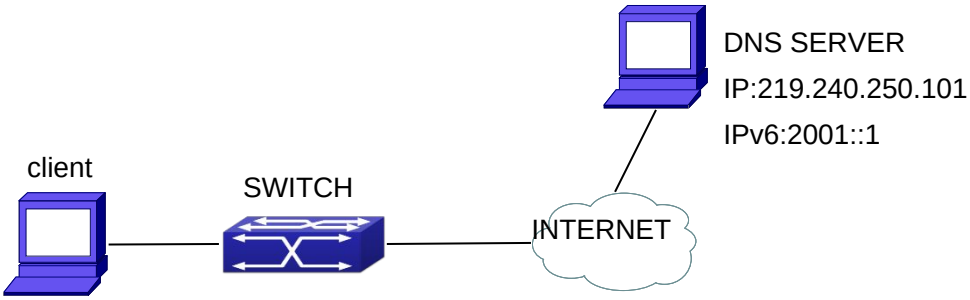


Figure 10-5 DNS SERVER typical environment

The figure above shows a DNS server application. In some cases, a client PC does not know the real DNS server and points to the switch instead. The switch acts as a DNS server after two steps: enable the DNS server function globally, and configure the IP address of the real DNS server. With the DNS server function enabled, the switch looks up its local cache when it receives a DNS request from a client PC. If the requested domain is in the cache, the switch answers the request directly. Otherwise, the switch relays the request to the real DNS server, passes the reply to the client, and records the domain and its IP address for faster lookup later.

```
Switch configuration for DNS CLIENT:
Switch(config)# ip domain-lookup
Switch(config)# dns-server 219.240.250.101
Switch(config)# dns-server 2001::1
Switch#ping host www.google.com
Switch#traceroute host www.google.com
Switch#telnet host www.google.com
Switch configuration for DNS SERVER:
Switch(config)# ip domain-lookup
Switch(config)# dns-server 219.240.250.101
Switch(config)# dns-server 2001::1
Switch(config)# ip dns server
```

10.3.4 DNS Troubleshooting

When DNS is configured and used, it may fail because of a physical connection failure, an incorrect configuration, and so on. Make sure that:

- The physical connection to the DNS server is in good condition.
- All interfaces and link protocols are in the UP state (use the “**show interface**” command).
- The DNS dynamic lookup function is enabled (use the “ip domain-lookup” command) before the DNS client function is used. To use the DNS server function, enable it (use the “ip dns server” command).
- The DNS server address is configured (use the “**dns-server**” command) and the switch can ping the DNS server.
- If the problem persists, run debug dns all and other debugging commands, record the debug messages for 3 minutes, and send them to EnGenius technical support.

10.4 Summer Time

10.4.1 Introduction to Summer Time

Summer time, also called daylight saving time (DST), is a time system adopted to save energy. In summer, clocks are advanced by 1 hour so that people start the day earlier and use less lighting, saving electricity. The rules for summer time differ by country. Currently, about 110 countries use summer time.

Summer time is usually 1 hour ahead of standard time. For example, while summer time is in effect, 10:00 am standard time becomes 11:00 am summer time.

10.4.2 Summer Time Configuration Task Sequence

1. Configure an absolute or recurring time range for summer time

Command	Explanation
Global Mode	
clock summer-time <*word*> absolute <*HH:MM*> <*YYYY.MM.DD*> <*HH:MM*> <*YYYY.MM.DD*> [<*offset*>] **no clock summer-time**	Sets an absolute time range for summer time, with start and end times in a specified year.
clock summer-time <*word*> recurring <*HH:MM*> <*MM.DD*> <*HH:MM*> <*MM.DD*> [<*offset*>] **no clock summer-time**	Sets a recurring time range for summer time. Summer time begins at the start time and ends at the end time every year.
clock summer-time <*word*> recurring <*HH:MM*> <*week*> <*day*> <*month*> <*HH:MM*> <*week*> <*day*> <*month*> [<*offset*>] **no clock summer-time**	Sets a recurring time range for summer time. Summer time begins at the start time and ends at the end time every year.

10.4.3 Examples of Summer Time

Example 1:

Requirement: Summer time runs from 23:00 on April 1, 2012 to 00:00 on October 1, 2012, with a clock offset of 1 hour, and is named 2012.

The configuration procedure is as follows:

```
Switch(config)# clock summer-time 2012 absolute 23:00 2012.4.1 00:00
2012.10.1
```

Example 2:

Requirement: Summer time runs from 23:00 on the first Saturday of April to 00:00 on the last Sunday of October every year, with a clock offset of 2 hours, and is named time_travel.

The configuration procedure is as follows:

```
Switch(config)#clock summer-time time_travel recurring 23:00 first sat apr
00:00 last sun oct 120
```

10.4.4 Summer Time Troubleshooting

If a problem occurs when using summer time, check whether it is caused by the following:

- ☞ Check whether the command is entered in Global Mode.
- ☞ Check whether the system clock is correct.

Chapter 11 Virtualization Configuration

11.1 VSF

11.1.1 Overview

11.1.1.1 Introduction to VSF

VSF uses VSF ports to connect several devices into one virtual logical device. Through this virtual device, you can manage all the interconnected physical devices.

Traditional campus and data center networks use a multi-layer topology, as shown in Figure 11-1. This type of network has the following drawbacks:

(1) The network and servers are complex, which reduces operating efficiency and increases cost.

(2) Without stateful failover, a switch failure prolongs application recovery and service downtime.

(3) Low resource utilization reduces the return on investment (ROI) and increases capital expenditure.

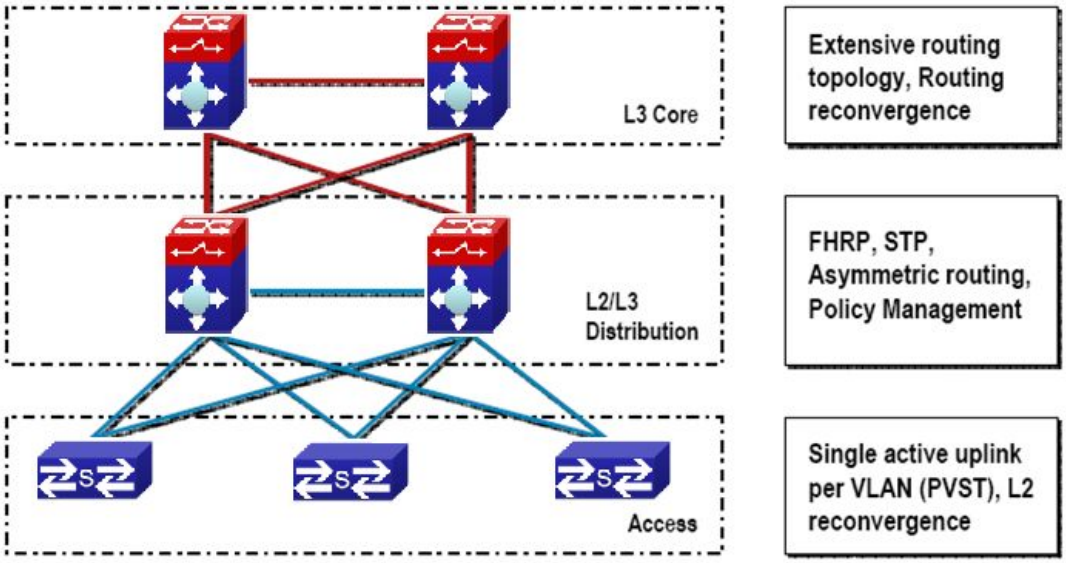


Figure 11-25 Traditional Enterprise Network

VSF addresses these problems by combining multiple VSF-capable devices into a single virtual switch. In VSF, the management engine and the switching fabric data plane of both switches can be active at the same time. VSF members are connected through the VSL. The VSL between member switches uses 10-Gigabit Ethernet connections (up to 4 links, for redundancy), as shown in Figure 11-2.

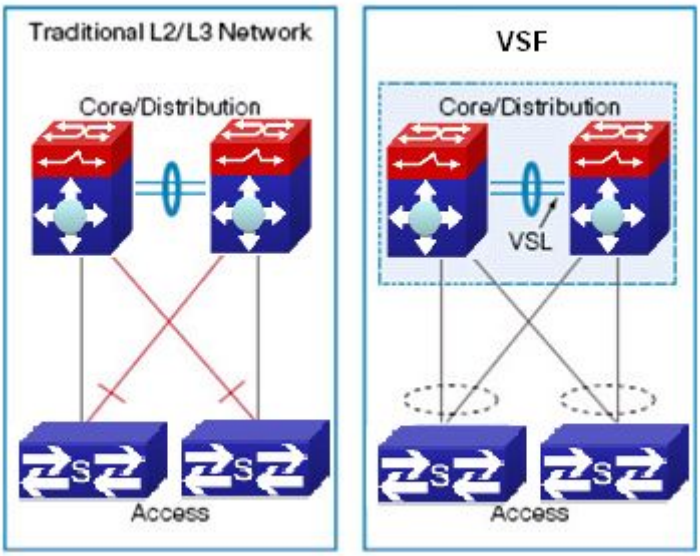


Figure 11-26 VSF

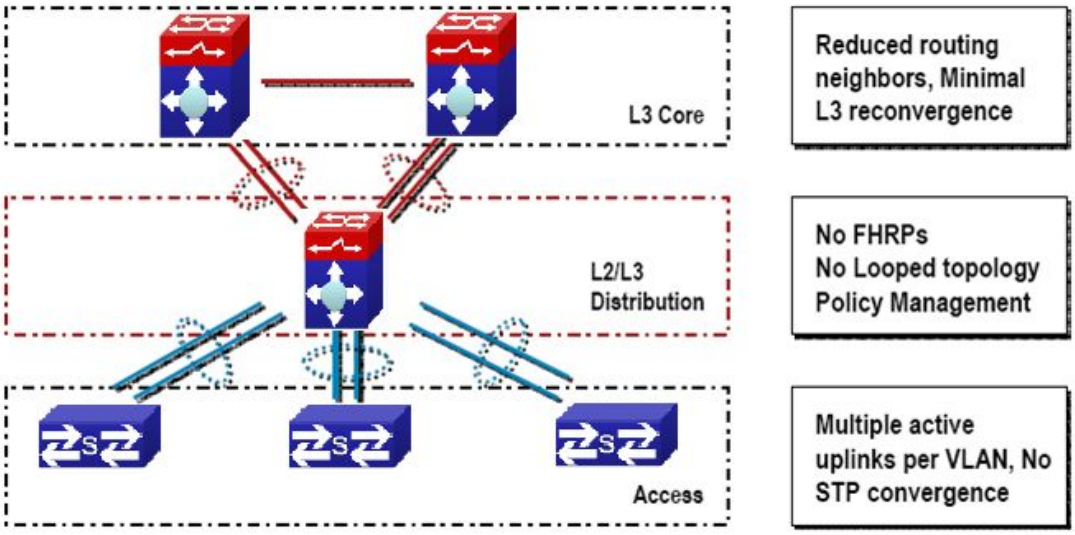


Figure 11-27 Enterprise that adopting virtual technology

Compared with the traditional L2/L3 network design, VSF provides several significant advantages, which can be summarized as follows:

1. VSF improves operating efficiency.

There is a single point of management, including the configuration file and the gateway IP address (HSRP/VRRP/GLBP are not needed).

One logical device composed of multiple physical devices creates a simple loop-free topology without the Spanning Tree Protocol (STP).

The underlying physical switches are connected through standard 10-Gigabit Ethernet ports, which provides flexible deployment options.

2. VSF minimizes communication interruptions.

Stateful failover does not interrupt applications that rely on network state information. With VSF, if a member of the virtual switch fails, the virtual switch recovers quickly without L2/L3 reconvergence. Port groups must use the active mode.

3. VSF expands system bandwidth.

All Layer 2 bandwidth in the VSF switch can be active, and load can be balanced across multiple VSF members as bandwidth is expanded.

11.1.1.2 Basic Concept

(1) Roles

Each device in the VSF is called a member device. Member devices have the following roles according to their function:

Master: Manages the entire VSF.

Standby Master: The standby member of the VSF, which serves as the backup of the Master. When the Master fails, the system automatically switches over to the Standby Master, which takes over the Master's work.

Slave: A member device other than the Master and the Standby Master.

The Master, Standby Master, and Slaves are determined by election. A VSF can have only one Master and one Standby Master; all other members are Slaves.

(2) VSF Port

A VSF port is a logical port dedicated to VSF. There are two VSF ports, vsf-port1 and vsf-port2, which take effect only after they are bound to VSF physical ports.

(3) VSF Physical Port

A VSF physical port is a port used for the physical VSF connection. It can be a dedicated VSF port, an Ethernet port, or an optical port, depending on the device model; select ports according to the actual situation. Ethernet ports and optical ports normally forward service packets. After they are bound to a VSF port, they become VSF physical ports and carry packets between members, including VSF negotiation packets and service packets that need to cross members.

(4) Merge

Merge is the process in which two separately running VSFs become one VSF through physical connection and configuration.

Figure 11-4 Merging of VSFs

(5) Split

Split is the process in which a formed VSF is divided into two VSFs because a VSF link failure physically disconnects adjacent devices.

Figure 11-5 Splitting of a VSF

(6) Member Priority

Member priority is an attribute of a member device used in role election to determine the role of each member. The higher the priority, the more likely the device is to become the Master. The default

priority is 1. To make a specific device the Master, manually increase its priority before forming the VSF.

11.1.1.3 Glossary

VSF: Virtual Switching Framework.

VSF Port: A VSF port is a mode of a port-channel.

VSF AM: The active master card of the VSF.

VSF SM: The standby master card of the VSF.

VSF Slave: A slave card of the VSF that runs the VSF protocol; during switchover, it refers to a Chassis AM that is not elected as the VSF AM or VSF SM.

VSF Member: A member device of the VSF.

VSF Master Member: The master member device of the VSF.

VSF Standby Master Member: The standby master member device of the VSF.

VSF Slave Member: A slave member device of the VSF.

HA: High Availability.

AM: Active Master, the active controller. It is the switch that works in Master mode and is in the active state. A box-type VSF group can have only one AM at any time.

SM: Standby Master, the switch in a box-type VSF group that works in Master mode and is in the standby state.

SSO: Stateful Switchover. Protocol-layer data is synchronized, so that after a switchover the backup master can quickly take over the corresponding protocol functions without losing their data; however, routing protocol data is not synchronized.

NSF: Non-Stop Forwarding. Because the data plane is managed independently of the control plane, the data plane continues to forward data normally when the control plane fails.

NSF/SSO: Nonstop Forwarding with Stateful Switchover. Data forwarding is not affected by an SSO master-backup switchover.

Syn process: The HA synchronization process, responsible for collecting and distributing the protocol functions of the active and backup controllers and for data synchronization communication.

GR: Graceful Restart. To ensure uninterrupted forwarding, routing protocols must be extended to support GR.

MAD: Multi-Active Detection.

11.1.1.4 VSF Typical Application

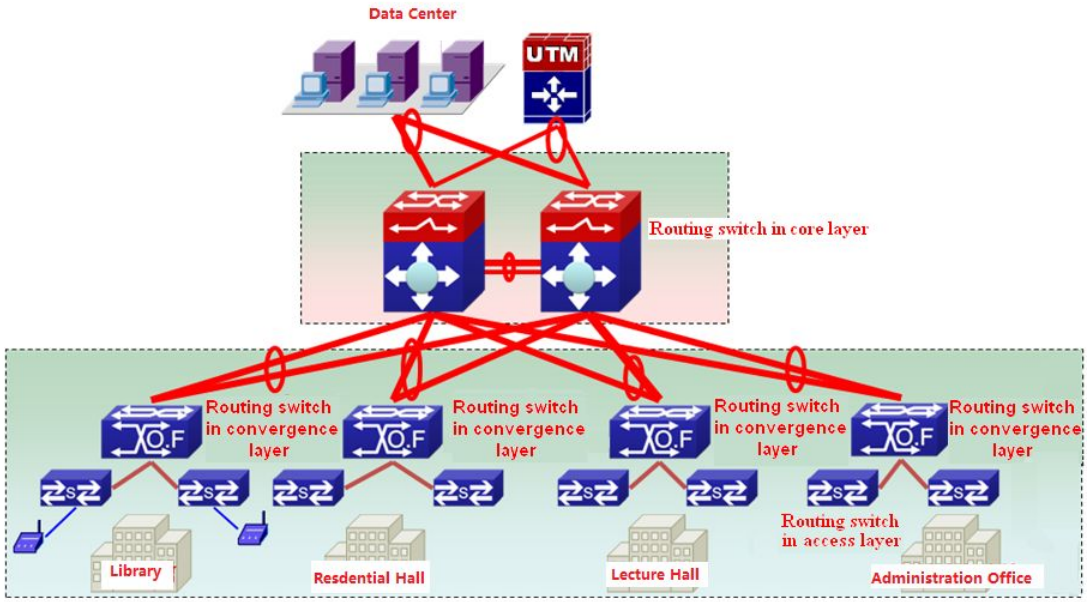


Figure 11-6 Application of VSF in a Campus Data Center

Figure 1-6 shows VSF applied in a campus network. With VSF, several devices are grouped into a single logical device. This simplification removes the need for MSTP and VRRP and simplifies network configuration. In addition, link aggregation across devices is used (port groups must use the active mode). When some members fail, recovery does not depend on MSTP, VRRP, or other protocol convergence, which increases reliability.

11.1.1.5 LACP MAD

LACP MAD is a detection method based on LACP dynamic aggregation. Each member device of the VSF must have at least one port connected to the intermediate device.

Note: The intermediate device must support the LACP extension function.

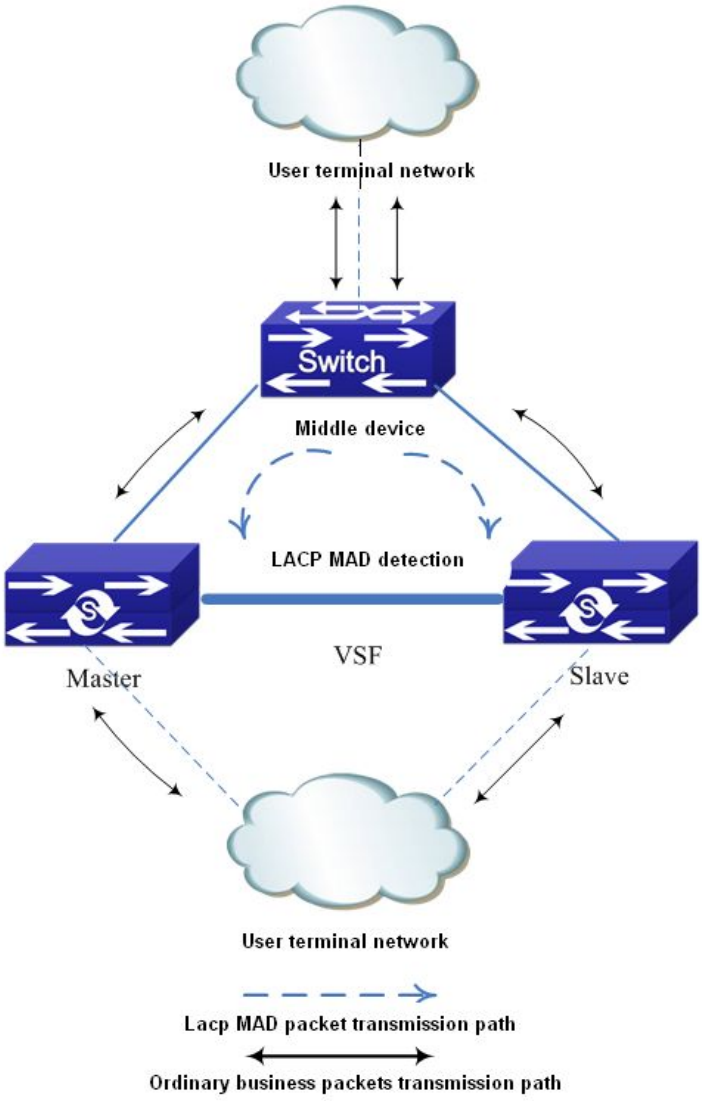


Figure 11-28 LACP MAD detection

LACP MAD detection is implemented through extended LACP packet content. A new TLV (Type Length Value) is defined in the extended field of LACP packets to exchange the ActiveID of the VSF. The ActiveID is unique within a VSF system and is expressed as the member number of the master device in the VSF. After LACP MAD detection is enabled, member devices exchange ActiveID information with other member devices through LACP packets.

When the VSF is running normally, the ActiveID values in the LACP packets sent by all member devices are the same, and there is no Active conflict.

When the VSF splits into two VSFs, the ActiveID values in the LACP packets sent by member devices of different VSFs differ, and a multi-Active conflict is detected.

11.1.1.6 BFD MAD

The BFD MAD detection topology is simpler than that of LACP MAD. However, once a VLAN is selected for BFD MAD detection, that VLAN and its ports are dedicated to BFD MAD and cannot be configured with other functions.

Connection method: Select one port on member 1 and one port on member 2, and connect them with a cable.

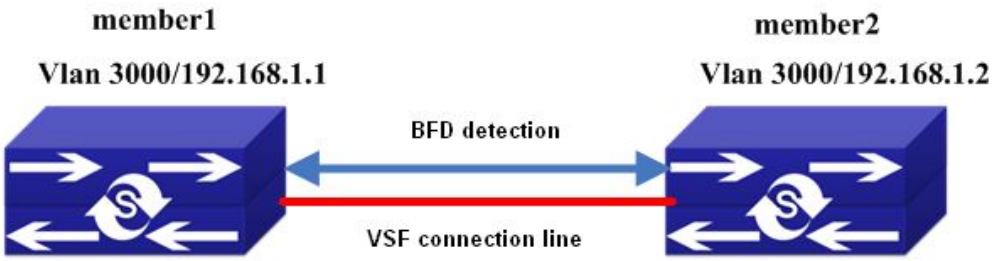


Figure 11-29 BFD MAD detection

BFD MAD detection is implemented through the BFD protocol. For BFD MAD detection to work, enable BFD MAD detection on a Layer 3 interface and configure MAD IP addresses on that interface. Unlike an ordinary IP address, a MAD IP address is bound to a member device. A MAD IP address must be configured for each member device in the VSF, and all of them must be in the same network segment.

When the VSF is running normally, only the MAD IP address configured on the master is effective; the MAD IP addresses associated with the other member devices are not effective, and the BFD session is down.

When the VSF splits into two VSFs, the MAD IP addresses associated with the members of the new VSF become effective. A BFD session is established between the two VSFs over the BFD MAD detection link to detect the multi-Active conflict.

11.1.2 VSF Configuration

11.1.2.1 VSF Configuration Tasks

VSF configuration task list:

Configure the VSF member number (required)

Configure the priority of VSF members (optional)

Configure the VSF domain (optional)

Configure the logical VSF port

Bind the physical port to the logical port

Switch the device from stand-alone mode to VSF mode

Configure automatic VSF merging (optional)

Configure VSF member descriptions (optional)

Configure the down-report delay of the VSF link (optional)

Configure the MAC address retention time of the VSF group after a VSF split (optional)

Switch the device from VSF mode to stand-alone mode

Detect VSF link status changes quickly

Configure the VSF member number (required)

Command	Explanation
Global Mode	
vsf member <*member-id*> **no vsf member <*member-id*>**	Configures the VSF member number. The no command deletes it.

Configure the priority and domain of VSF members (optional)

Command	Explanation
Global Mode	

vsf priority <*priority*> **no vsf priority**	Configures the priority of the VSF member. The no command deletes it.
vsf domain <*domain-id*> **no vsf domain**	Configures the VSF domain. The no command restores the default value of 1.

Configure the logical VSF port and bind the physical port to the logical port

Command	Explanation
Global Mode	
vsf port-group <*port-number*> **no vsf port-group <*port-number*>**	Configures the logical VSF port. The no command deletes it.
VSF Port Configuration Mode	
vsf port-group interface Ethernet <*interface-list*> **no vsf port-group interface Ethernet <*interface-list*>**	Binds the physical port to the logical VSF port. The no command deletes the binding.

Switch the operating mode of the device

Command	Explanation
Global Mode	
switch convert mode (stand-alone | vsf)	Switches the device from stand-alone mode to VSF mode, or from VSF mode to stand-alone mode.

Other configurations

Command	Explanation
Global Mode	
vsf auto-merge enable **no vsf auto-merge enable**	Enables automatic VSF merging. The no command disables this function.
vsf member <*member-id*> description <*text*> **no vsf member <*member-id*> description**	Configures a description for the VSF member. This information is written only to the configuration file of the VSF master. The no command deletes the description.
vsf link delay<*interval*> **no vsf link delay**	Configures the down-report delay of the VSF link, which prevents split and merge events caused by short-term link changes. The **no** command restores the default delay.
vsf mac-address persistent <*timer* | *always*> **no vsf mac-address persistent**	Configures the MAC address retention time of the VSF group after a VSF split. The **no** command deletes the MAC address retention time.
vsf non-wait port-inactive **no vsf non-wait port-inactive**	Detects VSF link status changes quickly to discover a VSF split. The **no** command restores the default detection method.

11.1.2.2 LACP MAD Configuration

LACP MAD configuration task list:

1. Create the aggregation group
2. Add ports to the aggregation group, with at least one port in active mode
3. Configure fast detection (optional)
4. Enable LACP MAD

1. **Create the aggregation group**

Command	Explanation

Global Mode	
port-group <*port-group id*> **no Port-group <*port-group id*>**	Creates the port group. The no command deletes it.

2. **Add ports to the aggregation group**

Command	Explanation
Global Mode	
interface ethernet <*port-num*>	Enters Port Mode.
Port Mode	
port-group <*port-group id*> mode <*active* | *passive*> **no port-group <*port-group id*>**	Adds the port to the aggregation group. The no command removes it from the group.

3. **Configure fast detection**

Command	Explanation
Port Mode	
lacp timeout <*long* | *short*> **no lacp timeout**	Configures fast detection. The no command deletes the configuration.

4. **Enable LACP MAD**

Command	Explanation
Port-channel Mode	
vsf mad lacp <*enable* | *disable*>	Enables or disables LACP MAD on the port group.

11.1.2.3 BFD MAD Configuration

BFD MAD configuration task list:

1. Create the VLAN used for BFD MAD
2. Add the ports used for BFD MAD to the VLAN
3. Configure the IP addresses of the BFD MAD Layer 3 interface
4. Enable the BFD MAD function

1. **Create the VLAN used for BFD MAD**

Command	Explanation
Global Mode	
vlan <*vlan-id*> **no vlan <*vlan-id*>**	Creates a VLAN. The no command deletes it.

2. **Add the ports used for BFD MAD to the VLAN**

Command	Explanation
VLAN Configuration Mode	
switchport interface ethernet <*port-num*> **no switchport interface ethernet <*port-num*>**	Adds the port to the VLAN. The no command removes it from the VLAN.

3. **Configure the IP addresses of the BFD MAD Layer 3 interface**

Command	Explanation
Global Mode	
Interface vlan <*vlan-id*>	Enters VLAN Interface Mode.
VLAN Interface Mode	

vsf mad ip address <*ip-addr*> <*ip-mask*> member <*member-id*> **no vsf mad ip address <*ip-addr*> <*ip-mask*> member <*member-id*>**	Configures the IP address used for BFD MAD on the Layer 3 interface. The no command deletes it.

4. **Enable the BFD MAD function**

Command	Explanation
VLAN Interface Mode	
vsf mad bfd <*enable* | *disable*>	Enables or disables BFD MAD.

11.1.3 Typical VSF Example

Case 1:

In stand-alone mode, configure two switches to form a VSF. The member numbers of the two devices are 1 and 2. To make member 2 the VSF master, set the priority of member 2 to 32. Create two VSF port groups between the devices, and bind one 10-Gigabit port to each VSF port group.

```
switch1 VSF configuration as following：
switch1#config
switch1(config)#vsf member 1
switch1(config)#vsf port-group 1
switch1(config-vsf-port1)#vsf port-group interface ethernet 1/1
switch1(config)#vsf port-group 2
switch1(config-vsf-port1)#vsf port-group interface ethernet 1/2
switch1(config)#exit
switch1(config)#switch convert mode vsf
switch2 VSF configuration as following：
switch2#config
switch1(config)#vsf member 2
switch1(config)#vsf priority 32
switch1(config)#vsf port-group 1
switch1(config-vsf-port1)#vsf port-group interface ethernet 2/1
switch1(config)#vsf port-group 2
switch1(config-vsf-port1)#vsf port-group interface ethernet 2/2
switch1(config)#exit
switch1(config)#switch convert mode vsf
```

Case 2:

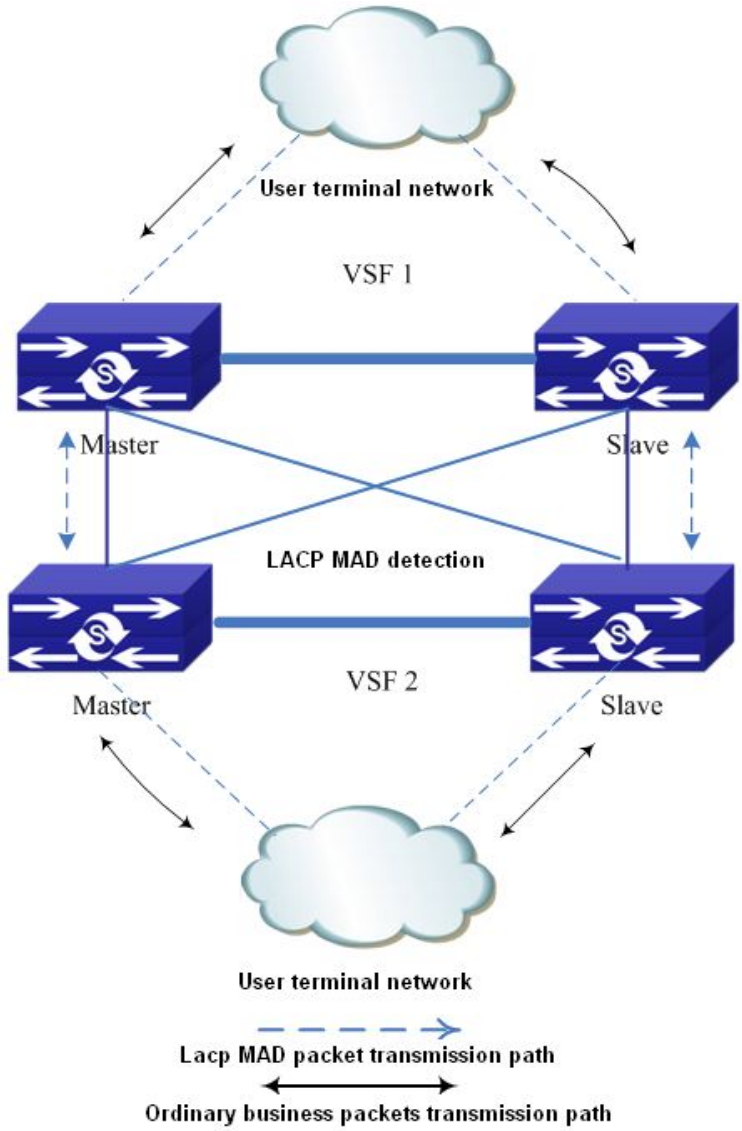


Figure 11-9 LACP MAD Detection Topology

As shown in the figure above, LACP MAD detection is used between two VSFs. VSF1 and VSF2 are each a detected device and an intermediate device for the other. The configuration is the same as above. Recommendation: Use cross-connections between the devices, so that VSF1 can still act as the intermediate device to detect VSF2 after VSF1 splits.

Assume that VSF1 and VSF2 use the LACP ports ethernet1/1/1, ethernet 1/1/2, ethernet 2/1/1, and ethernet 2/1/2.

VSF1 configuration:

```
Switch(config)#vsf domain 1
```

Configure the VSF domain number. It can be set to another value, but it must differ from that of VSF2.

```
Switch(config)#port-group 1
Switch(config)#interface ethernet 1/1/1
Switch(config-if-ethernet1/1/1)#port-group 1 mode active
Switch(config)#interface ethernet 1/1/2
Switch(config-if-ethernet1/1/2)#port-group 1 mode active
Switch(config)#interface ethernet 2/1/1
Switch(config-if-ethernet2/1/1)#port-group 1 mode active
Switch(config)#interface ethernet 2/1/2
Switch(config-if-ethernet2/1/2)#port-group 1 mode active
```

```
Switch(config-if-ethernet2/1/2)#interface port-channel 1
Switch(config-if-port-channel1)#vsf mad lacp enable
```

VSF2 configuration:

```
Switch(config)#vsf domain 2
```

Configure the VSF domain number. It can be set to another value, but it must differ from that of VSF1.

The other configurations are the same as on VSF1.

Case 3:

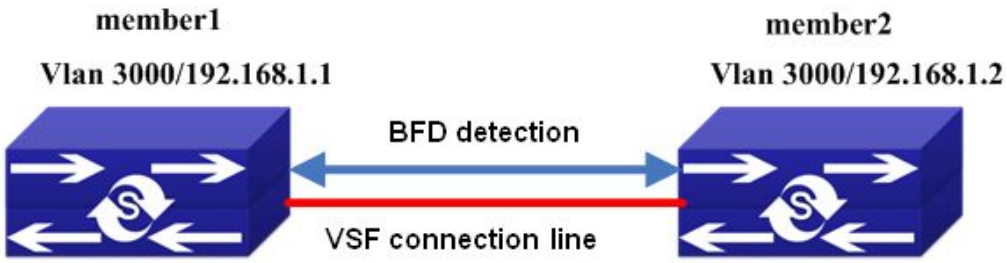


Figure 11-300

As shown in the figure above, assume that the ports used for BFD MAD detection are ethernet1/1/1 and ethernet2/1/1 and the VLAN is 3000. The MAD IP addresses in this example are in the network segment of 192.168.1.1.

BFD MAD configuration of the switch:

```
Switch(config)#vlan 3000
Switch(config-vlan3000)#interface ethernet 1/1/1
Switch(config-if-ethernet1/1/1)#switchport access vlan 3000
Switch(config-if-ethernet1/1/1)#interface ethernet 2/1/1
Switch(config-if-ethernet2/1/1)#switchport access vlan 3000
Switch(config-if-ethernet2/1/1)#interface vlan 3000
Switch(config-if-vlan3000)#vsf mad bfd enable
Switch(config-if-vlan3000)#vsf mad ip add 192.168.1.1 255.255.255.0 member 2
Switch(config-if-vlan3000)#vsf mad ip add 192.168.1.2 255.255.255.0 member 1
```

Note: Because the whole VSF is now a single device, the BFD MAD configuration is performed on one device.

11.1.4 VSF Troubleshooting

If a command does not work during VSF configuration and use, check the following:

- ☞ Check whether the device is in the correct operating mode. Some commands can be configured only in VSF mode, while others can be configured in both VSF mode and stand-alone mode.

If VSF fails or behaves abnormally, check the following:

- ☞ First, check whether the physical connections are correct. Currently, VSF can be bound only to 10-Gigabit ports or logical ports. Check whether the bound ports are 10-Gigabit ports.
- ☞ Check whether the VSF member IDs conflict. If the member IDs conflict, the devices cannot form a VSF.
- ☞ Check whether the VSF domain numbers are the same. The devices can form a VSF only if their VSF domain numbers are identical.
- ☞ Check whether the logical port has other configurations, such as duplex, bandwidth limits, security authentication, or ACLs.

When configuring in VSF mode, note the following:

- In VSF mode, VSF-related configurations can be made on each member separately. However, configurations made on members cannot be saved independently; the write command can be executed only on the VSF master. The VSF-related configurations of all members are saved in the vsf.cfg file of the master.
- In VSF mode, some settings, such as the VSF domain number, member priority, and member ID, can still be configured or modified. After configuration, show run displays the latest values, but they take effect only after the configuration is saved and the device is restarted.
- In VSF mode, port groups must use the active mode to ensure fast fault convergence.

If problems occur during LACP MAD configuration and use, check the following:

- When creating aggregation groups, the aggregation groups on the detected device and the intermediate device must be identical.
- For LACP MAD, the ports of the detected device and the intermediate device cannot be added to the aggregation group in on mode; at least one side must use active mode.
- When the detected device and the intermediate device are both VSFs, the domain IDs (VSF domain numbers) of the two VSFs must be different.

If problems occur during BFD MAD configuration and use, check the following:

- The BFD MAD function cannot be enabled on interface vlan 1.
- The BFD MAD function and loop-prevention functions (such as STP and MSTP) are mutually exclusive. If a loop-prevention function is configured on the VSF, disable it on the ports used for BFD MAD detection; otherwise, detection may fail.
- If trunk ports (including port-channel ports) are configured on the VSF, check whether the VLAN used for BFD MAD detection is allowed on these trunks (trunk ports belong to all VLANs by default). If it is, filter the BFD MAD VLAN on these ports; otherwise, a loop may occur.
- If hybrid ports are configured on the VSF, make sure they do not belong to the VLAN used for BFD MAD detection; otherwise, a loop may occur.
- The VLAN interface and all ports in the VLAN used for BFD MAD detection cannot be configured with anything other than BFD MAD detection.
- If BFD MAD is no longer used, clear its configuration and remove the cable used for BFD MAD.
- If BFD MAD configuration is complete and you need to modify other ports, VLANs, or global commands, make sure the changes comply with the items above.
- Recommendation: Select a VLAN that is not used for any service as the dedicated VLAN for BFD MAD detection.

Chapter 12 IPv6 Configuration

12.1 DHCPv6

12.1.1 Introduction to DHCPv6

DHCPv6 [RFC3315] is the IPv6 version of the Dynamic Host Configuration Protocol (DHCP). It assigns IPv6 addresses and other network configuration parameters, such as DNS server addresses and domain names, to DHCPv6 clients. DHCPv6 is a stateful address autoconfiguration protocol for IPv6. In stateful address configuration, the DHCPv6 server assigns a complete IPv6 address to the client and provides the DNS server address, domain name, and other configuration information. DHCPv6 packets can be forwarded through relay agents, and the DHCPv6 server can record the binding between each IPv6 address and its client, which enhances network management. The DHCPv6 server can also provide stateless DHCPv6 service, which assigns only the DNS server address, domain name, and other configuration information without assigning an IPv6 address; this compensates for the limitations of stateless IPv6 address autoconfiguration. DHCPv6 also provides the prefix delegation extension, with which an upstream router automatically assigns address prefixes to downstream routers. This enables hierarchical automatic IPv6 address assignment and simplifies IPv6 network deployment for ISPs.

The DHCPv6 protocol has three entities: the client, the relay, and the server. DHCPv6 runs over UDP. The DHCPv6 client sends request messages to the DHCPv6 server or relay with destination port 547, and the DHCPv6 server and relay send reply messages with destination port 546. The DHCPv6 client sends Solicit and Request messages to the multicast address ff02::1:2, which is reserved for DHCPv6 relays and servers.

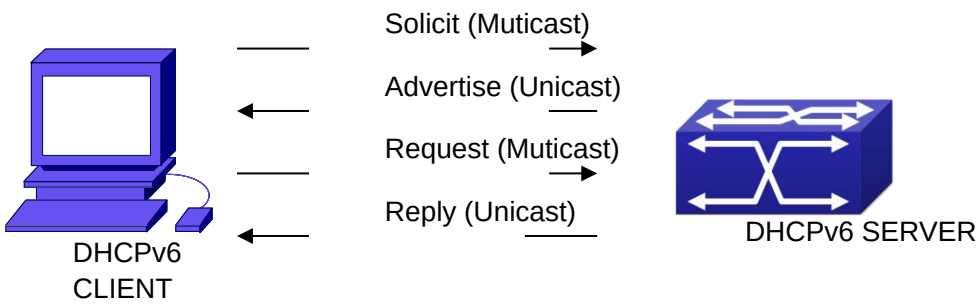


Figure 12-1 DHCPv6 Negotiation

When a DHCPv6 client requests an IPv6 address and other configuration from a DHCPv6 server, the client must first locate a DHCPv6 server and then request the configuration from it.

1. To locate a server, the DHCPv6 client multicasts a SOLICIT message to all DHCPv6 relay agents and servers using the multicast address FF02::1:2.
2. Every DHCPv6 server that receives the request replies to the client with an ADVERTISE message, which includes the server identifier (DUID) and the server priority.
3. The client may receive multiple ADVERTISE messages. The client selects one server and replies to it with a REQUEST message to request the address advertised in the ADVERTISE message.
4. The selected DHCPv6 server then confirms the IPv6 address and other configuration to the client with a REPLY message.

These four steps complete the dynamic host configuration process. However, if the DHCPv6 server and the DHCPv6 client are not on the same network, the server does not receive the multicast packets sent by the client and therefore sends no DHCPv6 packets to the client. In this case, a

DHCPv6 relay is required to forward the DHCPv6 packets so that the client and the server can complete the exchange.

The switch implements the DHCPv6 server, the DHCPv6 relay, and the DHCPv6 prefix delegation client. When the DHCPv6 relay receives a message from a DHCPv6 client, it encapsulates the request in a Relay-Forward message and sends it to the next DHCPv6 relay or to the DHCPv6 server. Messages from the server are encapsulated in Relay-Reply messages and sent to the DHCPv6 relay, which removes the encapsulation and delivers the message to the DHCPv6 client or to the next DHCPv6 relay in the network.

For DHCPv6 prefix delegation, the DHCPv6 server is configured on the PE router and the DHCPv6 client is configured on the CPE router. The CPE router sends an address prefix allocation request to the PE router and obtains a preconfigured address prefix, instead of having the prefix set manually. The negotiation between the prefix delegation client and server is similar to that used to obtain a DHCPv6 address. The CPE router then divides the allocated prefix, whose length must be less than 64, into subnets with a /64 prefix length. These prefixes are advertised in router advertisement (RA) messages to the hosts directly connected to the client.

12.1.2 DHCPv6 Server Configuration

The DHCPv6 server configuration task list is as follows:

1. Enable or disable the DHCPv6 service
2. Configure a DHCPv6 address pool

(1) Create or delete a DHCPv6 address pool

(2) Configure the parameters of the DHCPv6 address pool

3. Enable the DHCPv6 server function on an interface

1. Enable or disable the DHCPv6 service

Command	Explanation
Global Mode	
service dhcpv6 **no service dhcpv6**	Enables the DHCPv6 service. The no command disables the DHCPv6 service.

2. Configure a DHCPv6 address pool

(1) Create or delete a DHCPv6 address pool

Command	Explanation
Global Mode	
ipv6 dhcp pool <*poolname*> **no ipv6 dhcp pool <*poolname*>**	Creates a DHCPv6 address pool and enters DHCPv6 Address Pool Mode. The no command deletes the address pool.

(2) Configure the parameters of the DHCPv6 address pool

Command	Explanation
DHCPv6 Address Pool Mode	
network-address <*ipv6-pool-start-address*> {<*ipv6-pool-end-address*> | <*prefix-length*>} [eui-64] **no network-address**	Configures the range of IPv6 addresses that the address pool can assign. The no command deletes the configured range.
dns-server <*ipv6-address*> **no dns-server <*ipv6-address*>**	Configures the DNS server address for DHCPv6 clients. The no command deletes the DNS server address.
domain-name <*domain-name*> **no domain-name <*domain-name*>**	Configures the domain name for DHCPv6 clients. The no command deletes the domain name.

excluded-address <*ipv6-address*> **no excluded-address <*ipv6-address*>**	Excludes an IPv6 address in the address pool from dynamic assignment. The no command removes the exclusion.
lifetime {<*valid-time*> | infinity} {<*preferred-time*> | infinity} **no lifetime**	Configures the valid lifetime and preferred lifetime of the addresses in the DHCPv6 address pool. The no command restores the default lifetimes.

3. Enable the DHCPv6 server function on an interface

Command	Explanation
Interface Mode	
ipv6 dhcp server <*poolname*> [preference <*value*>] [rapid-commit] [allow-hint] **no ipv6 dhcp server <*poolname*>**	Enables the DHCPv6 server function on the specified interface and binds the DHCPv6 address pool to be used. The no command disables the DHCPv6 server function on the interface.

12.1.3 DHCPv6 Relay Configuration

The DHCPv6 relay configuration task list is as follows:

1．Enable or disable the DHCPv6 service

2．Configure the DHCPv6 relay on an interface

1. Enable the DHCPv6 service

Command	Explanation
Global Mode	
service dhcpv6 **no service dhcpv6**	Enables the DHCPv6 service. The no command disables the DHCPv6 service.

2. Configure the DHCPv6 relay on an interface

Command	Explanation
Interface Mode	
ipv6 dhcp relay destination {[<*ipv6-address*>] [interface { <*interface-name*> | vlan <*1-4096*>}]} **no ipv6 dhcp relay destination {[<*ipv6-address*>] [interface { <*interface-name*> | vlan <*1-4096*>}]}**	Specifies the destination address to which the DHCPv6 relay forwards packets. The no command deletes the configuration.

12.1.4 DHCPv6 Prefix Delegation Server Configuration

The DHCPv6 prefix delegation server configuration task list is as follows:

1. Enable or disable the DHCPv6 service
2. Configure a prefix delegation pool
3. Configure a DHCPv6 address pool

(1) Create or delete a DHCPv6 address pool

(2) Configure the prefix delegation pool used by the DHCPv6 address pool

(3) Configure a static prefix delegation binding

(4) Configure other parameters of the DHCPv6 address pool

4. Enable the DHCPv6 prefix delegation server function on an interface

1. Enable or disable the DHCPv6 service

Command	Explanation

Global Mode	
service dhcpv6 **no service dhcpv6**	Enables the DHCPv6 service. The no command disables the DHCPv6 service.

2. Configure a prefix delegation pool

Command	Explanation
Global Mode	
ipv6 local pool <*poolname*> <*prefix*|*prefix-length*> <*assigned-length*> **no ipv6 local pool <*poolname*>**	Configures a prefix delegation pool. The no command deletes the prefix delegation pool.

3. Configure a DHCPv6 address pool

(1) Create or delete a DHCPv6 address pool

Command	Explanation
Global Mode	
ipv6 dhcp pool <*poolname*> **no ipv6 dhcp pool <*poolname*>**	Creates a DHCPv6 address pool and enters DHCPv6 Address Pool Mode. The no command deletes the address pool.

(2) Configure the prefix delegation pool used by the DHCPv6 address pool

Command	Explanation
DHCPv6 Address Pool Mode	
prefix-delegation pool <*poolname*> ***[*lifetime {<*valid-time*> | *infinity*}** **{<*preferred-time*> | infinity}]** **no prefix-delegation pool <*poolname*>**	Specifies the prefix delegation pool used by the DHCPv6 address pool, from which available prefixes are assigned to clients. The no command removes the prefix delegation pool from the address pool.

(3) Configure a static prefix delegation binding

Command	Explanation
DHCPv6 Address Pool Mode	
prefix-delegation <*ipv6-prefix/prefix-length*> <*client-DUID*> [iaid <*iaid*>*] **[lifetime {<valid-time> | infinity}** **{<*preferred-time*> | infinity}]** **no prefix-delegation** **<*ipv6-prefix/prefix-length*> <*client-DUID*> [iaid <*iaid*>]**	Statically binds an IPv6 prefix to a client (and optionally to a specified IAID). The no command deletes the static binding.

(4) Configure other parameters of the DHCPv6 address pool

Command	Explanation
DHCPv6 Address Pool Mode	
dns-server <*ipv6-address*> **no dns-server <*ipv6-address*>**	Configures the DNS server address for DHCPv6 clients. The no command deletes the DNS server address.
domain-name <*domain-name*> **no domain-name <*domain-name*>**	Configures the domain name for DHCPv6 clients. The no command deletes the domain name.

4. Enable the DHCPv6 prefix delegation server function on an interface

Command	Explanation
Interface Mode	

ipv6 dhcp server <*poolname*> [preference <*value*>] [rapid-commit] [allow-hint] **no ipv6 dhcp server <*poolname*>**	Enables the DHCPv6 server function on the specified interface and binds the DHCPv6 address pool to be used. The no command disables the DHCPv6 server function on the interface.

12.1.5 DHCPv6 Prefix Delegation Client Configuration

The DHCPv6 prefix delegation client configuration task list is as follows:

1. Enable or disable the DHCPv6 service
2. Enable the DHCPv6 prefix delegation client function on an interface

1. Enable or disable the DHCPv6 service

Command	Explanation
Global Mode	
service dhcpv6 **no service dhcpv6**	Enables the DHCPv6 service. The no command disables the DHCPv6 service.

2. Enable the DHCPv6 prefix delegation client function on an interface

Command	Explanation
Interface Mode	
ipv6 dhcp client pd <*prefix-name*> [rapid-commit] **no ipv6 dhcp client pd**	Enables the prefix delegation request function of the DHCPv6 client on the specified interface and associates the obtained prefix with the specified general prefix name. The no command disables the function.

12.1.6 DHCPv6 Configuration Examples

Example 1:

When IPv6 networks are deployed, the switch can be configured as a DHCPv6 server to manage the allocation of IPv6 addresses. Both stateful and stateless DHCPv6 are supported.

Topology:

The access layer uses Switch1 to connect users in dormitory buildings. Switch2 is configured as a DHCPv6 relay at the primary aggregation layer. Switch3 is configured as a DHCPv6 server at the secondary aggregation layer and connects to the backbone network or a higher aggregation layer. The PCs must run an operating system with a DHCPv6 client, such as Windows Vista.

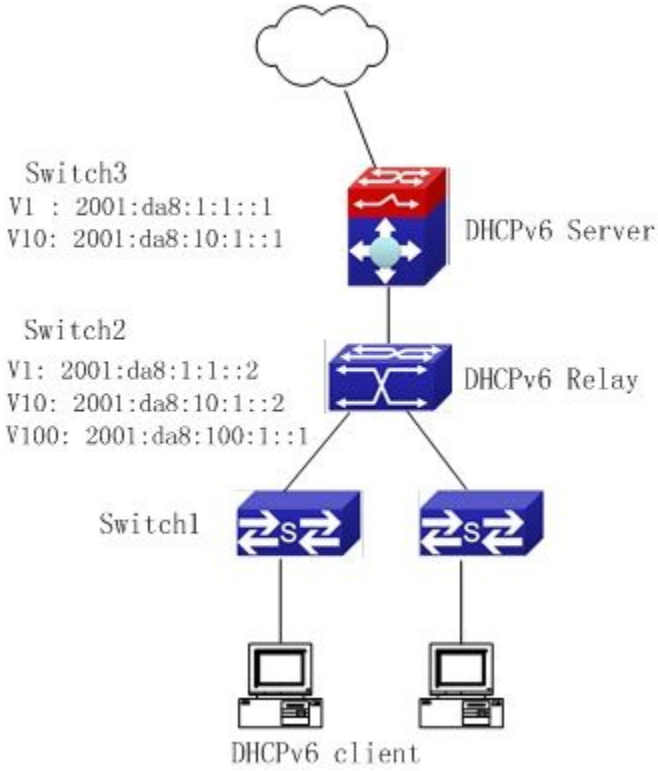


Figure 14-2

Usage Guidelines:

```
Switch3 configuration：
Switch3>enable
Switch3#config
Switch3(config)#service dhcpv6
Switch3(config)#ipv6 dhcp pool EastDormPool
Switch3(dhcpv6-EastDormPool-config)#network-address 2001:da8:100:1::1
2001:da8:100:1::100
Switch3(dhcpv6-EastDormPool-config)#excluded-address 2001:da8:100:1::1
Switch3(dhcpv6-EastDormPool-config)#dns-server 2001:da8::20
Switch3(dhcpv6-EastDormPool-config)#dns-server 2001:da8::21
Switch3(dhcpv6-EastDormPool-config)#domain-name dhcpv6.com
Switch3(dhcpv6-EastDormPool-config)#lifetime 1000 600
Switch3(dhcpv6-EastDormPool-config)#exit
Switch3(config)#interface vlan 1
Switch3(Config-if-Vlan1)#ipv6 address 2001:da8:1:1::1/64
Switch3(Config-if-Vlan1)#exit
Switch3(config)#interface vlan 10
Switch3(Config-if-Vlan10)#ipv6 address 2001:da8:10:1::1/64
Switch3(Config-if-Vlan10)#ipv6 dhcp server EastDormPool preference 80
Switch3(Config-if-Vlan10)#exit
Switch3(config)#
Switch2 configuration：
Switch2>enable
Switch2#config
Switch2(config)#service dhcpv6
Switch2(config)#interface vlan 1
Switch2(Config-if-Vlan1)#ipv6 address 2001:da8:1:1::2/64
Switch2(Config-if-Vlan1)#exit
Switch2(config)#interface vlan 10
Switch2(Config-if-Vlan10)#ipv6 address 2001:da8:10:1::2/64
Switch2(Config-if-Vlan10)#exit
Switch2(config)#interface vlan 100
Switch2(Config-if-Vlan100)#ipv6 address 2001:da8:100:1::1/64
```

```
Switch2(Config-if-Vlan100)#no ipv6 nd suppress-ra
Switch2(Config-if-Vlan100)#ipv6 nd managed-config-flag
Switch2(Config-if-Vlan100)#ipv6 nd other-config-flag
Switch2(Config-if-Vlan100)#ipv6 dhcp relay destination 2001:da8:10:1::1
Switch2(Config-if-Vlan100)#exit
Switch2(config)#
```

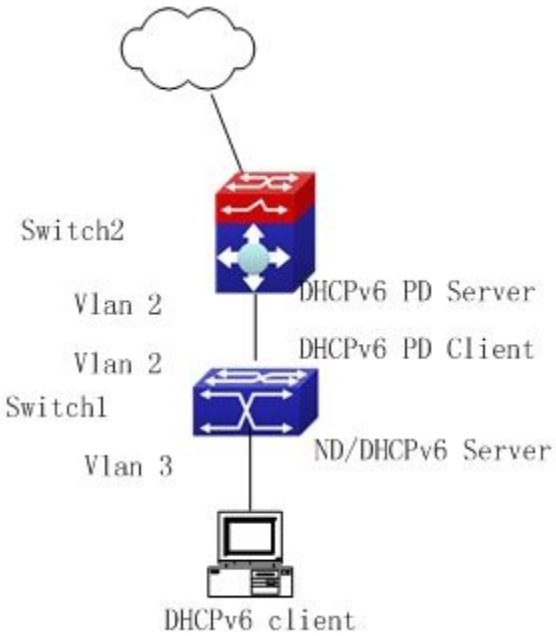
Example 2:

When a network operator deploys IPv6 networks, automatic network configuration can be achieved through IPv6 prefix delegation instead of configuring each switch manually:

1. Configure the switching or routing device connected to the client switch as the DHCPv6 prefix delegation server, which maintains a local database of the bindings between the allocated prefixes and the DUIDs of the client switches.
2. Configure the switch as the prefix delegation client so that it obtains an IPv6 address prefix from the prefix delegation server, through a process similar to DHCPv6 address allocation.
3. The edge device that receives the address prefix sends router advertisement (RA) messages containing the prefix through the interfaces connected to the hosts, and the hosts obtain valid IPv6 addresses through stateless autoconfiguration. At the same time, a stateless DHCPv6 server is configured on those interfaces to provide the DHCPv6 clients with information such as DNS servers and domain names.

Network Topology:

The edge switch is Switch1. Its interface connected to the core switch, Switch2, is configured as the prefix delegation client. The interfaces connected to the hosts are configured as stateless DHCPv6 servers that provide the hosts with stateless information, such as DNS servers and domain names, and router advertisements for stateless address allocation are enabled on these interfaces. On Switch2, the prefix delegation server is configured and router advertisements for stateful address allocation are enabled. The hosts must run an operating system with a DHCPv6 client, such as Windows Vista.



Usage Guidelines:

```
Switch2 configuration
Switch2>enable
Switch2#config
Switch2(config)#interface vlan 2
Switch2(Config-if-Vlan2)#ipv6 address 2001:da8:1100::1/64
Switch2(Config-if-Vlan2)#exit
```

```
Switch2(config)#service dhcpv6
Switch2(config)#ipv6 local pool client-prefix-pool 2001:da8:1800::/40 48
Switch2(config)#ipv6 dhcp pool dhcp-pool
Switch2(dhcpv6-dhcp-pool-config)#prefix-delegation pool client-prefix-pool
1800 600
Switch2(dhcpv6-dhcp-pool-config)#exit
Switch2(config)#interface vlan 2
Switch2(Config-if-Vlan2)#ipv6 dhcp server dhcp-pool
Switch2(Config-if-Vlan2)#exit
Switch1 configuration
Switch1>enable
Switch1#config
Switch1(config)#service dhcpv6
Switch1(config)#interface vlan 2
Switch1(Config-if-Vlan2)#ipv6 dhcp client pd prefix-from-provider
Switch1(Config-if-Vlan2)#exit
Switch1(config)#interface vlan 3
Switch1(Config-if-Vlan3)#ipv6 address prefix-from-provider 0:0:0:1::1/64
Switch1(Config-if-Vlan3)#exit
Switch1(config)#ipv6 dhcp pool foo
Switch1(dhcpv6-foo-config)#dns-server 2001:4::1
Switch1(dhcpv6-foo-config)#domain-name www.ipv6.org
Switch1(dhcpv6-foo-config)#exit
Switch1(config)#interface vlan 3
Switch1(Config-if-Vlan3)#ipv6 dhcp server foo
Switch1(Config-if-Vlan3)#ipv6 nd other-config-flag
Switch1(Config-if-Vlan3)#no ipv6 nd suppress-ra
Switch1(Config-if-Vlan3)#exit
```

12.1.7 DHCPv6 Troubleshooting

If DHCPv6 clients cannot obtain IPv6 addresses and other network parameters after the client hardware and cables have been verified, perform the following checks:

- ☞ Verify that the DHCPv6 server is running; if it is not, start the DHCPv6 server function.
- ☞ If the DHCPv6 clients and server are not on the same physical network, verify that the router forwarding DHCPv6 packets provides the DHCPv6 relay function. If the intermediate router does not support DHCPv6 relay, replace the router or upgrade its software to a version that supports DHCPv6 relay.
- ☞ If hosts connected to a DHCPv6-enabled switch cannot obtain IPv6 addresses, first check whether the ports connected to the hosts are directly connected to the port connected to the DHCPv6 server. If they are directly connected, check whether the IPv6 address range of the VLAN that the port belongs to is in the same subnet as the address pool configured on the DHCPv6 server. If they are not directly connected and a Layer 3 DHCPv6 relay is configured between the hosts and the DHCPv6 server, first check whether a valid IPv6 address has been configured on the switch interface connected to the hosts. If not, configure a valid IPv6 address. If one is configured, check whether it is in the same subnet as an address pool on the DHCPv6 server. If it is not, add a matching address pool.

12.2 DHCPv6 Option 37, 38

12.2.1 Introduction to DHCPv6 Option 37, 38

DHCPv6 (Dynamic Host Configuration Protocol for IPv6) is designed for the IPv6 addressing scheme and assigns IPv6 prefixes, IPv6 addresses, and other configuration parameters to hosts.

When a DHCPv6 client requests an address and configuration parameters from a DHCPv6 server on a different link, it communicates with the server through a DHCPv6 relay agent. The relay agent

encapsulates the DHCPv6 messages it receives in Relay-Forward messages and forwards them to the server. The server sends Relay-Reply messages back to the relay agent, which restores the DHCPv6 messages and sends them to the DHCPv6 client, completing the communication between the client and the server.

Using a DHCPv6 relay agent raises several problems, for example: how to assign IP addresses in a fixed range to specific users; how to prevent an illegal DHCPv6 client from launching an IP address exhaustion attack by forging the MAC address field of DHCPv6 packets; and how to prevent an illegal DHCPv6 client from launching a denial-of-service attack by using the MAC addresses of other legal clients. To solve these problems, the IETF defined RFC 4649 and RFC 4580, that is, DHCPv6 option 37 and option 38.

DHCPv6 option 37 and option 38 are similar to DHCP option 82. When a DHCPv6 client sends request packets to a DHCPv6 server through a DHCPv6 relay agent that supports option 37 and option 38, the relay agent adds these options to the request packets. In the server's response packets, option 37 and option 38 are meaningless and are removed. Therefore, option 37 and option 38 are transparent to the client.

Using option 37 and option 38, the DHCPv6 server can authenticate the identity of the DHCPv6 client and the DHCPv6 relay device, assign and manage client addresses flexibly through configured assignment policies, and effectively prevent DHCPv6 attacks based on the client information included, such as IP address exhaustion attacks triggered by forged MAC address fields in DHCPv6 packets. Because the server can identify multiple request packets from the same access port, it can limit the number of assigned addresses through a policy to prevent address exhaustion. RFC 4649 and RFC 4580 do not specify how the DHCPv6 server uses option 37 and option 38, so they can be used flexibly as needed.

12.2.2 DHCPv6 Option 37, 38 Configuration Task List

1. Configure the basic DHCPv6 snooping option functions
2. Configure the basic DHCPv6 relay option functions
3. Configure the basic DHCPv6 server option functions

1. Configure the basic DHCPv6 snooping option functions

Command	Description
Global Mode	
ipv6 dhcp snooping remote-id option **no ipv6 dhcp snooping remote-id option**	Enables DHCPv6 snooping support for option 37. The no command disables it.
ipv6 dhcp snooping subscriber-id option **no ipv6 dhcp snooping subscriber-id option**	Enables DHCPv6 snooping support for option 38. The no command disables it.
ipv6 dhcp snooping remote-id policy {drop | keep | replace} **no ipv6 dhcp snooping remote-id policy**	Configures the forwarding policy of the system for received DHCPv6 packets that carry option 37. The policy can be: **drop,** the system discards packets that carry option 37; **keep,** the system keeps option 37 unchanged and forwards the packet to the server; **replace,** the system replaces option 37 of the packet with its own option 37 before forwarding the packet to the server. The no command

	restores the forwarding policy for DHCPv6 packets with option 37 to replace.
ipv6 dhcp snooping subscriber-id policy {drop | keep | replace} **no ipv6 dhcp snooping subscriber-id policy**	Configures the forwarding policy of the system for received DHCPv6 packets that carry option 38. The policy can be: **drop,** the system discards packets that carry option 38; **keep,** the system keeps option 38 unchanged and forwards the packet to the server; **replace,** the system replaces option 38 of the packet with its own option 38 before forwarding the packet to the server. The no command restores the forwarding policy for DHCPv6 packets with option 38 to replace.
ipv6 dhcp snooping subscriber-id select (sp | sv | pv | spv) delimiter WORD (delimiter WORD |) **no ipv6 dhcp snooping subscriber-id select delimiter**	Configures the user-selected options used to generate the subscriber-id. The no command restores the default configuration, that is, the enterprise number together with the VLAN MAC address.
ipv6 dhcp snooping subscriber-id select (sp|sv|pv|spv) delimiter WORD (delimiter WORD |) **no ipv6 dhcp snooping subscriber-id select delimiter**	Configures the user-selected options used to generate the subscriber-id. The no command restores the default configuration, that is, the VLAN name together with the port name.
Port Mode	
ipv6 dhcp snooping remote-id <remote-id> **no ipv6 dhcp snooping remote-id**	Configures the content of option 37 added to received DHCPv6 request packets. **<remote-id>** is the user-defined remote-id content of option 37, a string of fewer than 128 characters. The no command restores the remote-id in option 37 to the enterprise number together with the VLAN MAC address.
ipv6 dhcp snooping subscriber-id <subscriber-id> **no ipv6 dhcp snooping subscriber-id**	Configures the content of option 38 added to received DHCPv6 request packets. **<subscriber-id>** is the user-defined subscriber-id content of option 38, a string of fewer than 128 characters. The no command restores the subscriber-id in option 38 to the VLAN name together with the port name, such as "Vlan2+Ethernet1/0/2".

2. Configure the basic DHCPv6 relay option functions

Command	Description
Global Mode	
ipv6 dhcp relay remote-id option **no ipv6 dhcp relay remote-id option**	Enables DHCPv6 relay support for option 37 on the switch. The no command disables it.
ipv6 dhcp relay subscriber-id option **no ipv6 dhcp relay subscriber-id option**	Enables DHCPv6 relay support for option 38 on the switch. The no command disables it.
ipv6 dhcp relay subscriber-id select (sp | sv | pv | spv) delimiter WORD (delimiter WORD |) **no ipv6 dhcp relay subscriber-id select delimiter**	Configures the user-selected options used to generate the subscriber-id. The no command restores the default configuration, that is, the VLAN name together with the port name.
Interface Mode (Layer 3 interface)	
ipv6 dhcp relay remote-id <remote-id> **no ipv6 dhcp relay remote-id**	Configures the content of option 37 added to received DHCPv6 request packets. **<remote-id>** is the user-defined remote-id content of option 37, a string of fewer than 128 characters. The no command restores the remote-id in option 37 to the enterprise number together with the VLAN MAC address.
ipv6 dhcp relay subscriber-id <subscriber-id> **no ipv6 dhcp relay subscriber-id**	Configures the content of option 38 added to received DHCPv6 request packets. **<subscriber-id>** is the user-defined subscriber-id content of option 38, a string of fewer than 128 characters. The no command restores the subscriber-id in option 38 to the VLAN name together with the port name, such as "Vlan2+Ethernet1/0/2".

3. Configure the basic DHCPv6 server option functions

Command	Description
Global Mode	
ipv6 dhcp server remote-id option **no ipv6 dhcp server remote-id option**	Enables the DHCPv6 server to recognize option 37. The no command disables it.
ipv6 dhcp server subscriber-id option **no ipv6 dhcp server subscriber-id option**	Enables the DHCPv6 server to recognize option 38. The no command disables it.
ipv6 dhcp use class **no ipv6 dhcp use class**	Enables the DHCPv6 server to use DHCPv6 classes during address assignment. The no command disables this function without removing the configured DHCPv6 class

	information.
ipv6 dhcp class <class-name> **no ipv6 dhcp class <class-name>**	Defines a DHCPv6 class and enters DHCPv6 Class Mode. The no command removes the DHCPv6 class.
Interface Mode	
ipv6 dhcp server select relay-forw **no ipv6 dhcp server select relay-forw**	Configures the DHCPv6 server, when a packet contains multiple option 37 or option 38 instances, to select option 37 and option 38 of the innermost relay-forw message. The no command restores the default, which selects option 37 and option 38 of the original packet.
DHCPv6 Class Mode	
{remote-id [*] <remote-id> [*] | subscriber-id [*] <subscriber-id> [*]} **no {remote-id [*] <remote-id> [*] | subscriber-id [*] <subscriber-id> [*]}**	Configures the option 37 and option 38 values that match the class in DHCPv6 Class Mode. The no command deletes the matching values.
DHCPv6 Address Pool Mode	
class <class-name> **no class <class-name>**	Associates a class with the address pool in DHCPv6 Address Pool Mode and enters the class configuration mode of the address pool. The no command removes the association.
address range <start-ip> <end-ip> **no address range <start-ip> <end-ip>**	Sets the address range for a DHCPv6 class in DHCPv6 Address Pool Mode. The no command removes the address range. The prefix/plen format is not supported.

12.2.3 DHCPv6 Option 37, 38 Examples

12.2.3.1 DHCPv6 Snooping Option 37, 38 Example

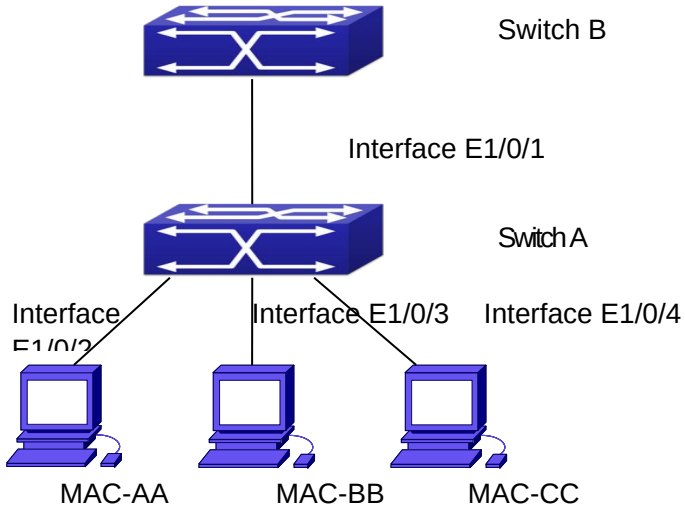


Figure 12-4 DHCPv6 Snooping Option Schematic

As shown in the figure above, Mac-AA, Mac-BB, and Mac-CC are normal users connected to untrusted interfaces 1/0/2, 1/0/3, and 1/0/4 respectively, and they obtain the IP addresses 2010:2, 2010:3, and 2010:4 through the DHCPv6 client. The DHCPv6 server is connected to trusted interface 1/0/1. Three address assignment policies (classes) are configured: CLASS1 matches option 38, CLASS2 matches option 37, and CLASS3 matches both option 37 and option 38. In the address pool EastDormPool, requests that match CLASS1, CLASS2, and CLASS3 are assigned addresses from 2001:da8:100:1::2 to 2001:da8:100:1::30, from 2001:da8:100:1::31 to 2001:da8:100:1::60, and from 2001:da8:100:1::61 to 2001:da8:100:1::100 respectively. DHCPv6 snooping is enabled and option 37 and option 38 are configured on Switch A.

```
Switch A configuration:
SwitchA(config)#ipv6 dhcp snooping remote-id option
SwitchA(config)#ipv6 dhcp snooping subscriber-id option
SwitchA(config)#int e 1/0/1
SwitchA(config-if-ethernet1/0/1)#ipv6 dhcp snooping trust
SwitchA(config-if-ethernet1/0/1)#exit
SwitchA(config)#interface vlan 1
SwitchA(config-if-vlan1)#ipv6 address 2001:da8:100:1::1
SwitchA(config-if-vlan1)#exit
SwitchA(config)#interface ethernet 1/0/1-4
SwitchA(config-if-port-range)#switchport access vlan 1
SwitchA(config-if-port-range)#exit
SwitchA(config)#
Switch B configuration:
SwitchB(config)#service dhcpv6
SwitchB(config)#ipv6 dhcp server remote-id option
SwitchB(config)#ipv6 dhcp server subscriber-id option
SwitchB(config)#ipv6 dhcp pool EastDormPool
SwitchB(dhcpv6-eastdormpool-config)#network-address 2001:da8:100:1::2
2001:da8:100:1::1000
SwitchB(dhcpv6-eastdormpool-config)#dns-server 2001::1
SwitchB(dhcpv6-eastdormpool-config)#domain-name dhcpv6.com
SwitchB(dhcpv6-eastdormpool-config)# excluded-address 2001:da8:100:1::2
SwitchB(dhcpv6-eastdormpool-config)#exit
```

```
SwitchB(config)#
SwitchB(config)#ipv6 dhcp class CLASS1
SwitchB(dhcpv6-class-class1-config)#remote-id 00-03-0f-00-00-01 subscriber-id
vlan1+Ethernet1/0/1
SwitchB(dhcpv6-class-class1-config)#exit
SwitchB(config)#ipv6 dhcp class CLASS2
SwitchB(dhcpv6-class-class2-config)#remote-id 00-03-0f-00-00-01 subscriber-id
vlan1+Ethernet1/0/2
SwitchB(dhcpv6-class-class2-config)#exit
SwitchB(config)#ipv6 dhcp class CLASS3
SwitchB(dhcpv6-class-class3-config)#remote-id 00-03-0f-00-00-01 subscriber-id
vlan1+Ethernet1/0/3
SwitchB(dhcpv6-class-class3-config)#exit
SwitchB(config)#ipv6 dhcp pool EastDormPool
SwitchB(dhcpv6-eastdormpool-config)#class CLASS1
SwitchB(dhcpv6-pool-eastdormpool-class-class1-config)#address range
2001:da8:100:1::3 2001:da8:100:1::30
SwitchB(dhcpv6-pool-eastdormpool-class-class1-config)#exit
SwitchB(dhcpv6-eastdormpool-config)#class CLASS2
SwitchB(dhcpv6-pool-eastdormpool-class-class2-config)#address range
2001:da8:100:1::31 2001:da8:100:1::60
SwitchB(dhcpv6-eastdormpool-config)#class CLASS3
SwitchB(dhcpv6-pool-eastdormpool-class-class3-config)#address range
2001:da8:100:1::61 2001:da8:100:1::100
SwitchB(dhcpv6-pool-eastdormpool-class-class3-config)#exit
SwitchB(dhcpv6-eastdormpool-config)#exit
SwitchB(config)#interface vlan 1
SwitchB(config-if-vlan1)#ipv6 address 2001:da8:100:1::2/64
SwitchB(config-if-vlan1)#ipv6 dhcp server EastDormPool
SwitchB(config-if-vlan1)#exit
SwitchB(config)#
```

12.2.3.2 DHCPv6 Relay Option 37, 38 Example

Example 1:

When an IPv6 campus network is deployed, the DHCPv6 server function of a routing device can be used to allocate IPv6 addresses if no dedicated server is used for unified IPv6 address allocation and management. The DHCPv6 server supports both stateful and stateless DHCPv6.

Network topology:

At the access layer, the Layer 2 access device Switch1 connects users in the dormitories. At the first-level aggregation layer, the aggregation device Switch2 acts as the DHCPv6 relay agent. At the second-level aggregation layer, the aggregation device Switch3 acts as the DHCPv6 server and connects to the backbone network or to devices at a higher aggregation layer. On the user side, the PCs generally run Windows Vista, which includes a DHCPv6 client.

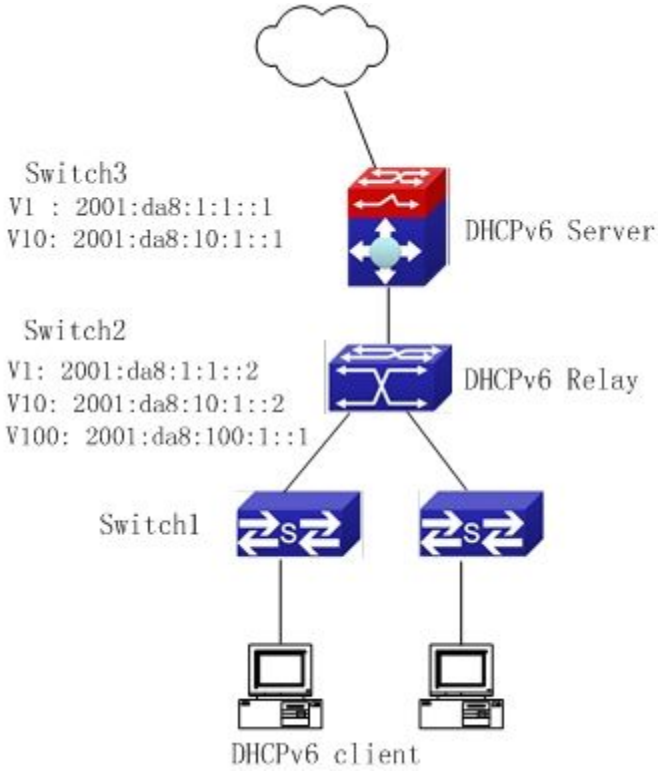


Figure 12-5 DHCPv6 Relay Option Schematic

```
Switch2 configuration:
S2(config)#service dhcpv6
S2(config)#ipv6 dhcp relay remote-id option
S2(config)#ipv6 dhcp relay subscriber-id option
S2(config)#vlan 10
S2(config-vlan10)#int vlan 10
S2(config-if-vlan10)#ipv6 address 2001:da8:1:::2/64
S2(config-if-vlan10)#ipv6 dhcp relay destination 2001:da8:10:1::1
S2(config-if-vlan10)#exit
S2(config)#
```

12.2.4 DHCPv6 Option 37, 38 Troubleshooting

- Request packets sent by a DHCPv6 client are multicast packets that are received only within the client's VLAN. For the DHCPv6 server to receive packets from the client, the DHCPv6 client and server must be in the same VLAN; otherwise, a DHCPv6 relay is required.
- For DHCPv6 request packets that carry option 37 and option 38, snooping can perform one of the following operations: replace the original option 37 and option 38 with its own; discard the packets; or forward the packets unchanged without adding or removing options. If a client obtains a wrong address or no address based on option 37 and option 38, check the snooping option 37/38 policy configured on the second device.
- By default, the DHCPv6 server obtains option 37 and option 38 from the packet sent by the client; if the packet does not contain them, the server obtains option 37 and option 38 from the packet sent by the relay.
- The DHCPv6 server checks only the option 37 and option 38 added by the first DHCPv6 relay; that is, in relayed packets, only option 37 and option 38 of the innermost relay-forw message are valid.

12.3 Prevent ND Spoofing

12.3.1 Overview

ND (Neighbor Discovery) is the neighbor discovery protocol of IPv6. Its operating principle is similar to that of ARP, so ND spoofing and attacks are prevented in the same way as ARP spoofing.

12.3.2 Prevent ND Spoofing Configuration

To prevent ND spoofing, perform the following steps:

4. Disable the ND automatic update function
5. Disable the ND automatic learning function
6. Convert dynamic ND entries to static ND entries

1. Disable the ND automatic update function

Command	Explanation
Global Mode and Port Mode	
ipv6 nd-security updateprotect **no ipv6 nd-security updateprotect**	Disables the ND automatic update function. The no command enables the ND automatic update function.

2. Disable the ND automatic learning function

Command	Explanation
Global Mode and Interface Mode	
ipv6 nd-security learnprotect **no ipv6 nd-security learnprotect**	Disables the ND automatic learning function. The no command enables the ND automatic learning function.

3. Convert dynamic ND entries to static ND entries

Command	Explanation
Global Mode and Port Mode	
ipv6 nd-security convert	Converts dynamic ND entries to static ND entries.

12.3.3 Prevent ND Spoofing Example

For an example, see the ARP binding example in the ARP chapter.

12.4 RIPng

12.4.1 Introduction to RIPng

RIP, on which RIPng is based, was first introduced in ARPANET and is designed for small, simple networks. RIPng is a distance vector routing protocol based on the Bellman-Ford algorithm. Network devices running a distance vector routing protocol periodically send two kinds of information to their neighbors:

• The number of hops to the destination network, that is, the metric or the number of networks to pass.

• The next hop, that is, the direction (vector) to use to reach the destination network.

Distance vector Layer 3 switches periodically send their entire routing tables to neighboring Layer 3 switches. Each Layer 3 switch builds its own routing table based on the information received from its

neighbors and then sends this information to its own neighbors. Because the routing table is built on second-hand information, routes of more than 15 hops are considered unreachable.

RIPng runs over UDP. Hosts running RIPng send and receive packets on UDP port 521. All Layer 3 switches running RIPng send their routing tables to all neighboring Layer 3 switches every 30 seconds. If no update is received from a neighbor within 180 seconds, the neighbor is considered failed and the networks connected through it are considered unreachable. However, the routes through that Layer 3 switch are kept in the routing table for another 120 seconds before they are deleted.

Because Layer 3 switches running RIPng build their routing tables from second-hand information, counting to infinity may occur. In a RIPng network, when a RIPng route becomes unreachable, the neighboring RIPng Layer 3 switch does not send a route update immediately; instead, it waits until the update interval expires (every 30 seconds) and then sends update packets containing that route. If, before it sends the update, its neighbors send packets that still contain the route through the failed neighbor, "counting to infinity" occurs: the route to the unreachable Layer 3 switch is selected again with a progressively increasing metric. This greatly affects route selection and route convergence time.

To avoid counting to infinity, RIPng provides mechanisms such as split horizon and triggered updates to prevent routing loops. Split horizon prevents routes learned from a gateway from being sent back to that gateway. There are two split horizon methods: simple split horizon and split horizon with poison reverse. Simple split horizon removes the routes learned from a neighbor gateway from the updates sent to that gateway; poison reverse not only removes these routes but also sets their metrics to infinity. With triggered updates, whenever a gateway changes a route metric, it immediately advertises update packets instead of waiting for the 30-second timer.

RIPng currently has only one version, version 1, which is defined in RFC 2080. RIPng sends update packets as multicast packets (multicast address FF02::9).

Each Layer 3 switch running RIPng maintains a route database containing all route entries for reachable destinations, and builds its routing table from this database. When a RIPng Layer 3 switch sends route updates to its neighbors, the packets contain the complete routing table. Therefore, in a large network, each Layer 3 switch must transmit and process a large amount of routing data, which degrades network performance.

In addition, RIPng allows IPv6 routes discovered by other routing protocols to be imported into its routing table.

RIPng operates as follows:

1. When RIPng is enabled, the switch sends request packets to the neighboring Layer 3 switches by multicast; on receiving the request, the neighbors reply with packets containing their local routing information.
2. The Layer 3 switch updates its local routing table when it receives the reply packets and sends triggered update packets to its neighbors to advertise the route changes. On receiving a triggered update, each neighboring Layer 3 switch sends triggered update packets to its own neighbors. After a series of triggered updates, all Layer 3 switches obtain and maintain the latest routing information.

In addition, RIPng Layer 3 switches advertise their local routing tables to their neighbors every 30 seconds. On receiving the packets, the neighbors update their local routing tables, select the best routes, and advertise the updates to their own neighbors, so that the updated routes become valid throughout the network. RIPng also uses a timeout mechanism for outdated routes: if a switch does not receive periodic updates from a neighbor within a certain interval (the invalid timer), it considers the routes from that neighbor invalid; after holding the routes for a further interval (the garbage-collection timer), it deletes them.

As IPv6 networks develop, some network segments may still not support IPv6, so IPv6 traffic must be carried through tunnels. Therefore, RIPng can be configured on tunnels and can pass through networks that do not support IPv6 by using unicast packets encapsulated in IPv4.

12.4.2 RIPng Configuration Task List

RIPng configuration task list:

1. Enable the RIPng protocol (required)
   （1） Enable or disable the RIPng protocol
   （2） Configure the interfaces running RIPng
2. Configure RIPng protocol parameters (optional)
   （1） Configure the RIPng sending mechanism
      1) Configure the destination address for specified RIPng packets
   （2） Configure RIPng routing parameters
      1) Configure route redistribution (the default route metric and the routes of other protocols to be imported into RIPng)
      2) Configure the route offset
      3) Configure and apply route filters
      4) Configure split horizon
3. Configure other RIPng parameters
   (2) Configure the RIPng update, timeout, and hold-down timers
4. Delete specified routes from the RIPng routing table
5. Configure RIPng route aggregation
   （1） Configure IPv6 aggregate routes in Router Mode
   （2） Configure IPv6 aggregate routes in Interface Mode
   （3） Display IPv6 aggregate route information
6. Configure redistribution of OSPFv3 routes into RIPng
   （1） Enable redistribution of OSPFv3 routes into RIPng
   （2） Display and debug the configuration of OSPFv3 route redistribution into RIPng

1. Enable the RIPng protocol

The basic RIPng configuration on the switch is simple. Normally, you only need to enable RIPng and configure the interfaces that run RIPng; the switch then sends and receives RIPng packets with the default RIPng settings.

Command	Explanation
Global Mode	
[no] router IPv6 rip	Enables the RIPng protocol. The **no router IPv6 rip** command disables the RIPng protocol.
Interface Mode	
[no] IPv6 router rip	Enables RIPng on the interface. The **no IPv6 router rip** command disables RIPng on the interface.

2. Configure RIPng protocol parameters

(1) Configure the RIPng sending mechanism

1) Configure unicast transmission of RIPng packets

Command	Explanation

Router Mode	
[no] neighbor <*IPv6-address*> <*ifname*>	Specifies the IPv6 link-local address and interface of a neighboring router to which RIPng packets are sent by unicast. The **no neighbor <*IPv6-address*> <*ifname*>** command deletes the specified neighbor.
[no] passive-interface <*ifname*>	Blocks RIPng multicast on the specified interface so that RIPng packets are sent only to Layer 3 switches configured as neighbors. The **no passive-interface <*ifname*>** command cancels this function.

(2) Configure RIPng routing parameters

1) Configure route redistribution (the default route metric and the routes of other protocols to be imported into RIPng)

Command	Explanation
Router Mode	
default-metric <*value*> no default-metric	Configures the default metric of redistributed routes. The **no default-metric** command restores the default value 1.
[no]redistribute {kernel |connected| static| ospf| isis| bgp} [metric<*value*>] [route-map<*word*>]	Redistributes routes from other routing protocols into RIPng. The **no redistribute {kernel | connected| static| ospf| isis| bgp} [metric<*value*>] [route-map<*word*>]** command cancels the redistribution of routes from the corresponding protocol.
[no]default-information originate	Generates a default route in RIPng. The **no default-information originate** command cancels this function.

2) Configure the route offset

Command	Explanation
Router Mode	
[no] offset-list <access-list-number |access-list-name> {in|out} <number > [<ifname>]	Adds an offset to the route metric when the interface sends or receives RIPng packets. The **no offset-list <*access-list-number* |*access-list-name*> {in|out} <*number* > [<*ifname*>]** command removes the offset list.

3) Configure and apply route filters and route aggregation

Command	Explanation
Router Mode	
[no] distribute-list {<*access-list-number* |*access-list-name*> | prefix<*prefix-list-name*>} {in|out} [<*ifname*>]	Filters routes when the interface sends or receives RIPng packets. The **no distribute-list {< *access-list-number* |*access-list-name* > | prefix<*prefix-list-name*>} {in|out} [<*ifname*>]** command cancels the route filter.
[no]aggregate-address <*IPv6-address*>	Configures route aggregation. The **no aggregate-address <*IPv6-address*** command cancels the route aggregation.

4) Configure split horizon

Command	Explanation
Interface Mode	
IPv6 rip split-horizon [poisoned]	Enables split horizon when the interface sends packets; **poisoned** enables poison reverse.

no IPv6 rip split-horizon	Disables split horizon.

3. Configure other RIPng protocol parameters

(1) Configure the RIPng update, timeout, and hold-down timers

Command	Explanation
Router Mode	
timers basic <*update*> <*invalid*> <*garbage*> **no timers basic**	Configures the RIPng update, timeout, and garbage-collection timers. The **no timers basic** command restores the default configuration.

4. Delete specified routes from the RIPng routing table

Command	Explanation
Admin Mode	
clear IPv6 rip route {<*IPv6-address*>|kernel|static|connected|rip|ospf|isis|bgp|all}	Deletes the specified routes from the RIPng routing table.

5. Configure RIPng route aggregation

(1) Configure IPv6 aggregate routes globally

Command	Explanation
Router Mode	
ipv6 rip aggregate-address X:X::X:X/M **no ipv6 rip aggregate-address X:X::X:X/M**	Configures an IPv6 aggregate route globally. The no command deletes the aggregate route.

(2) Configure IPv6 aggregate routes on an interface

Command	Explanation
Interface Mode	
ipv6 rip aggregate-address X:X::X:X/M **no ipv6 rip aggregate-address X:X::X:X/M**	Configures an IPv6 aggregate route on the interface. The no command deletes the aggregate route.

(3) Display IPv6 aggregate route information

Command	Explanation
Admin Mode and Configuration Mode	
show ipv6 rip aggregate	Displays IPv6 aggregate route information, such as the aggregation interface, metric, number of aggregated routes, and number of aggregations.

6. Configure redistribution of OSPFv3 routes into RIPng

(1) Enable redistribution of OSPFv3 routes into RIPng

Command	Explanation
RIPng Router Mode	
redistribute ospf [<*process-tag*>] [metric<*value*>] [route-map<*word*>] **no redistribute ospf [<*process-tag*>]**	Enables redistribution of OSPFv3 routes into RIPng. The no command disables the redistribution.

(2) Display and debug the configuration of OSPFv3 route redistribution into RIPng

Command	Explanation
Admin Mode	
show ipv6 rip redistribute	Displays RIPng routes redistributed from other routing protocols.
Admin Mode	
debug ipv6 rip redistribute message send **no debug ipv6 rip redistribute message send** **debug ipv6 rip redistribute route receive** **no debug ipv6 rip redistribute route receive**	Enables or disables debugging of messages sent by RIPng for the redistribution of OSPFv3 routes. Enables or disables debugging of route messages received from the NSM.

12.4.3 RIPng Configuration Examples

12.4.3.1 Typical RIPng Examples

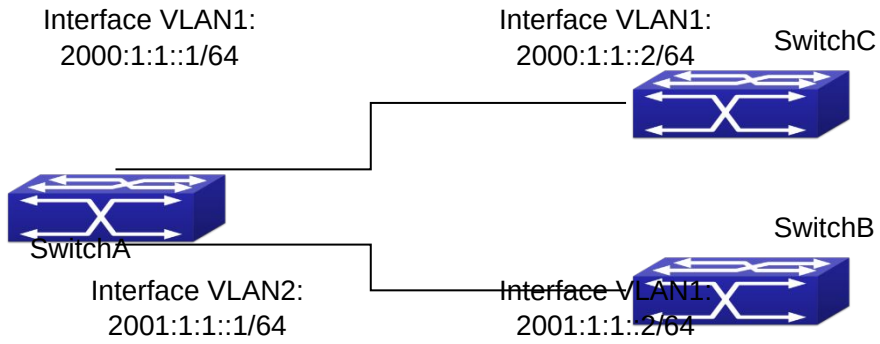


Figure 12-6 RIPng Example

As shown in the figure above, the network consists of three Layer 3 switches. SwitchA and SwitchB connect to SwitchC through interfaces vlan1 and vlan2. All three switches run RIPng. Assume that SwitchA (VLAN1: 2001:1:1::1/64 and VLAN2: 2001:1:1::1/64) exchanges update information only with SwitchB (VLAN1: 2001:1:1::2/64) and does not exchange update information with SwitchC (VLAN1: 2001:1:1::2/64).

The configurations of SwitchA, SwitchB, and SwitchC are shown below:

Layer 3 SwitchA

Enable the RIPng protocol

```
SwitchA(config)#router IPv6 rip
SwitchA(config-router)#exit
```

Configure the IPv6 address of vlan1 and enable RIPng on vlan1

```
SwitchA#config
SwitchA(config)# interface Vlan1
SwitchA(config-if-Vlan1)# IPv6 address 2000:1:1::1/64
SwitchA(config-if-Vlan1)#IPv6 router rip
SwitchA(config-if-Vlan1)#exit
```

Configure the IPv6 address of vlan2 and enable RIPng on vlan2

```
SwitchA(config)# interface Vlan2
SwitchA(config-if-Vlan2)#IPv6 address 2001:1:1::1/64
SwitchA(config-if-Vlan2)#IPv6 router rip
SwitchA(config-if-Vlan2)#exit
```

Configure interface vlan1 not to send RIPng messages to SwitchC

```
SwitchA(config)#
SwitchA(config-router)#passive-interface Vlan1
SwitchA(config-router)#exit
```

Layer 3 SwitchB

Enable the RIPng protocol

```
SwitchB (config)#router IPv6 rip
SwitchB (config-router-rip)#exit
```

Configure the IPv6 address of interface vlan1 and enable RIPng on it

```
SwitchB#config
SwitchB(config)# interface Vlan1
SwitchB(config-if)# IPv6 address 2001:1:1::2/64
SwitchB(config-if)#IPv6 router rip
SwitchB(config-if)exit
```

Layer 3 SwitchC

Enable the RIPng protocol

```
SwitchC(config)#router IPv6 rip
SwitchC(config-router-rip)#exit
```

Configure the IPv6 address of interface vlan1 and enable RIPng on it

```
SwitchC#config
SwitchC(config)# interface Vlan1
SwitchC(config-if)# IPv6 address 2000:1:1::2/64
SwitchC(config-if)#IPv6 router rip
SwitchC(config-if)exit
```

12.4.3.2 Typical RIPng Route Aggregation Examples

The application topology is as follows:

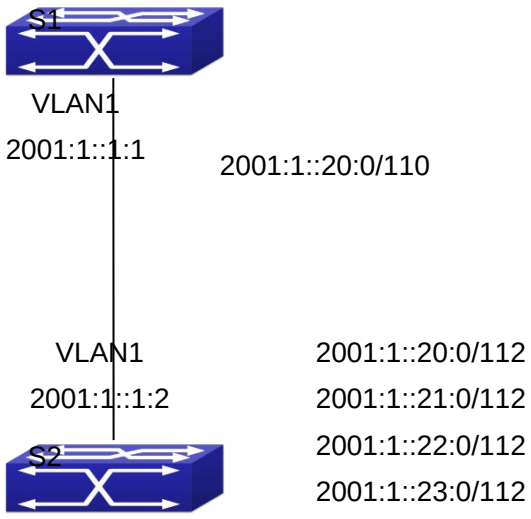


Figure 12-7 Typical application of RIPng route aggregation

In the topology above, S2 connects to S1 through interface vlan1 and has four other subnet routes: 2001:1::20:0/112, 2001:1::21:0/112, 2001:1::22:0/112, and 2001:1::23:0/112. S2 supports route aggregation. After the aggregate route 2001:1::20:0/110 is configured on interface vlan1 of S2, S2 aggregates the four subnet routes into the single route 2001:1::20:0/110 when it sends route updates to S1 through vlan1, and it does not send the individual subnet routes to the neighbor. This reduces the size of the routing table on S1 and saves memory.

S1 configuration:

```
S1(config)#router ipv6 rip
S1(config)# interface Vlan1
S1(config-if-Vlan1)#IPv6 address 2001:1::1:1/112
S1(config-if-Vlan1)#IPv6 router rip
```

S2 configuration:

```
S2(config)#router ipv6 rip
S2(config)#interface vlan 1
S2(config-if-Vlan1)#IPv6 address 2001:1::1:2/112
S2(config-if-Vlan1)#IPv6 router rip
S2(Config-if-Vlan1)#ipv6 rip agg 2001:1::20:0/110
```

12.4.4 RIPng Troubleshooting

RIPng may not work properly because of problems such as faulty physical connections or configuration errors. When configuring and using RIPng, pay attention to the following:

- First, ensure that the physical connections are correct and that IP forwarding is enabled.
- Second, ensure that the interface and the link-layer protocol are up (use the **show interface** command).
- Then enable RIPng (use the **router IPv6 rip** command), enable it on the interfaces (use the **IPv6 router** command), and set the RIPng parameters on the corresponding interfaces.
- Next, note this RIPng characteristic: a Layer 3 switch running RIPng sends route update messages every 30 seconds. If no route update messages are received from a switch within 180 seconds, that switch is considered unreachable, and the route to it remains in the routing table for another 120 seconds before it is deleted. Therefore, a RIPng route that is to be deleted is guaranteed to be removed from the routing table after 300 seconds.
- If the RIPng routing problem persists, use the **debug IPv6 rip** command to record debug messages for three minutes, and send them to EnGenius technical support.

12.5 OSPFv3

12.5.1 Introduction to OSPFv3

OSPFv3 (Open Shortest Path First version 3) is the IPv6 version of the OSPF protocol. It is a link-state interior dynamic routing protocol used within an autonomous system. Layer 3 switches build a link-state database by exchanging link states, and then use the Shortest Path First algorithm to generate a routing table from that database.

An autonomous system (AS) is a self-managed interconnected network. In large networks such as the Internet, the overall network is divided into autonomous systems. A large enterprise network connected to the Internet is an independent AS, because the other hosts on the Internet are not managed by that AS and do not share interior routing information with its Layer 3 switches.

Each link-state Layer 3 switch provides its neighboring Layer 3 switches with the following topology information:

• The network segment (link) connected to the Layer 3 switch

• The state of the connected link

Link-state information is flooded throughout the network so that all Layer 3 switches receive first-hand information. Link-state Layer 3 switches do not broadcast all the information in their routing tables; instead, they send only changed link-state information. Link-state Layer 3 switches establish neighbor relationships by sending Hello packets to their neighbors, and then exchange link-state advertisements (LSAs) with them. Each neighboring Layer 3 switch copies the LSA into its routing

table and passes the information on to the rest of the network. This process is called flooding. In this way, first-hand information is sent throughout the network and provides an accurate map for creating and updating routes. Link-state routing protocols select routes by cost instead of hop count. Cost is assigned automatically or manually. In the link-state algorithm, cost can reflect the number of hops a packet passes, the link bandwidth, and the current link load; the administrator can also add weights to assess the link state more accurately.

1) When a link-state Layer 3 switch joins a link-state network, it sends Hello packets to discover its neighbors and establish neighbor relationships.

2) The neighbors respond with information about their connected links and the related costs.

3) The originating Layer 3 switch uses this information to build its own routing table.

4) Then, as part of regular updates, the Layer 3 switch sends link-state advertisement (LSA) packets to its neighboring Layer 3 switches. The LSAs contain the links and related costs of that Layer 3 switch.

5) Each neighboring Layer 3 switch copies the LSA packet and passes it to the next neighbor (that is, flooding).

6) Because a Layer 3 switch does not recalculate its routing database before it forwards flooded LSAs, convergence time is greatly reduced.

A major advantage of link-state routing protocols is that counting to infinity cannot occur, because of the way they build their routing tables. A second advantage is that a link-state network converges very quickly: when the routing topology changes, updates are flooded throughout the network almost immediately. These advantages free up Layer 3 switch resources, because little processing capacity and bandwidth are consumed by bad route information.

OSPFv3 has the following features: It supports networks of various sizes; an OSPFv3 network can contain several hundred Layer 3 switches. Topology changes are detected quickly and updated LSAs are sent immediately, so routes converge quickly. Routes are calculated from link-state information with the shortest path algorithm, which eliminates routing loops. OSPFv3 divides the autonomous system into areas, which reduces database size, bandwidth usage, and calculation load. (According to their position in the autonomous system, Layer 3 switches are classified as internal area switches, area border switches, AS boundary switches, and backbone switches.) OSPFv3 supports load balancing over multiple equal-cost routes to the same destination. OSPFv3 supports four levels of routes (in order of preference: intra-area routes, inter-area routes, type 1 external routes, and type 2 external routes). OSPFv3 supports IP subnets, redistribution of routes from other routing protocols, and interface-based packet authentication. OSPFv3 sends packets by multicast.

Each OSPFv3 Layer 3 switch maintains a database describing the topology of the entire autonomous system. Each Layer 3 switch collects local status information, such as available interfaces and reachable neighbors, and sends link-state advertisements to exchange link-state information with the other OSPFv3 Layer 3 switches, forming a link-state database that describes the entire autonomous system. From this database, each Layer 3 switch builds a shortest path tree rooted at itself; the tree provides routes to all nodes in the autonomous system. If two or more Layer 3 switches are present on a network (a multi-access network), a designated Layer 3 switch and a backup designated Layer 3 switch are elected. The designated Layer 3 switch advertises the link state of the network, which reduces traffic among the Layer 3 switches on the multi-access network.

OSPFv3 requires the autonomous system to be divided into areas: area 0 (the backbone area) and non-zero areas. Routing information between areas is further abstracted and summarized to reduce the bandwidth required in the network. OSPFv3 uses four kinds of routes, from highest to lowest priority: intra-area routes, inter-area routes, type 1 external routes, and type 2 external routes. Intra-area and inter-area routes describe the internal network structure of the autonomous system, while external routes describe how to select routes to destinations outside the autonomous system. Type 1 external routes correspond to information that OSPFv3 imports from other interior routing protocols; their costs are comparable with OSPFv3 route costs. Type 2 external routes correspond to

information that OSPFv3 imports from exterior routing protocols; their costs are much greater than OSPFv3 route costs, so the OSPFv3 cost is ignored when route costs are calculated.

OSPFv3 areas are centered on the backbone area, identified as area 0. All other areas must connect logically to area 0, and area 0 must be contiguous. For this reason, virtual links are introduced to the backbone area so that physically separated areas still have logical connectivity to it. All Layer 3 switches in the same area must have the same configuration.

In short, LSAs can be transferred only between neighboring Layer 3 switches. OSPFv3 has seven types of LSA: link LSA, intra-area prefix LSA, router LSA, network LSA, inter-area prefix LSA, inter-area router LSA, and AS external LSA. Router LSAs are generated by each Layer 3 switch in an OSPF area and sent to all other neighboring Layer 3 switches in the area. Network LSAs are generated by the designated Layer 3 switch of a multi-access network in the OSPF area and sent to all other neighboring Layer 3 switches in the area. (To reduce traffic among the Layer 3 switches on a multi-access network, a designated Layer 3 switch and a backup designated Layer 3 switch are elected, and the designated Layer 3 switch advertises the network link state.) Inter-area prefix LSAs and inter-area router LSAs are generated by OSPF area border Layer 3 switches and transferred among those switches. AS external LSAs are generated by AS boundary Layer 3 switches and transferred throughout the autonomous system. Link LSAs are generated by each Layer 3 switch on a link and sent to the other Layer 3 switches on that link. Intra-area prefix LSAs are generated by the designated Layer 3 switch of each link in the area and flooded throughout the area.

To reduce the size of the topology database in autonomous systems with many external link-state advertisements, OSPFv3 allows some areas to be configured as stub areas. Router LSAs, network LSAs, inter-area prefix LSAs, link LSAs, and intra-area prefix LSAs can be advertised into a stub area. A stub area must use a default route: the area border Layer 3 switches of the stub area advertise the default route into the area through inter-area prefix LSAs, and these default routes are flooded only within the stub area. Each stub area has its own default route, and routing from the stub area to destinations outside the AS depends only on that default route.

The OSPFv3 route calculation process is outlined below:

1) Each OSPF-enabled Layer 3 switch maintains a link-state (LS) database describing the topology of the entire autonomous system. Each Layer 3 switch generates a link-state advertisement (router LSA) based on its surrounding network topology and sends it to the other Layer 3 switches in link-state update (LSU) packets. Thus, each Layer 3 switch receives LSAs from the other Layer 3 switches, and all the LSAs together form the link-state database.
2) Because an LSA describes the network topology around one Layer 3 switch, the LS database describes the topology of the entire network. Layer 3 switches can easily create a weighted directed graph from the LS database, and all Layer 3 switches in the same autonomous system have the same network topology map.
3) Each Layer 3 switch uses the shortest path first (SPF) algorithm to calculate a shortest path tree rooted at itself. The tree provides routes to all nodes in the autonomous system, and the leaf nodes contain external route information. External routes can be tagged by the Layer 3 switch that advertises them, so that additional information about the autonomous system can be recorded. As a result, the routing table of each Layer 3 switch is different.

OSPFv3 was developed by the IETF. The current OSPFv3 implementation is based on RFC 2328 and RFC 2740.

As IPv6 networks develop, IPv6 traffic sometimes has to cross networks that do not support IPv6, which requires tunneling. Therefore, OSPFv3 on the switch can be configured on tunnels and can cross non-IPv6 networks in unicast packets encapsulated in IPv4.

12.5.2 OSPFv3 Configuration Task List

OSPFv3 configuration task list:

1. Enable OSPFv3 (required)
   （1） Enable or disable OSPFv3 (required)
   （2） Configure the router ID of the Layer 3 switch running OSPFv3 (optional)
   （3） Configure the network scope for running OSPFv3 (optional)
   （4） Enable OSPFv3 on the interface (required)
2. Configure OSPFv3 auxiliary parameters (optional)
   （1） Configure OSPFv3 packet sending mechanism parameters
      1) Set the OSPFv3 interface to receive only
      2) Configure the cost for sending packets from the interface
      3) Configure OSPFv3 packet-sending timer parameters (the Hello interval on broadcast interfaces, the neighbor dead interval, the LSA transmission delay, and the LSA retransmission interval)
   （2） Configure OSPFv3 route redistribution parameters
      1) Configure default parameters (default type, default tag, and default cost)
      2) Configure the routes of other protocols to be imported into OSPFv3
   （3） Configure OSPFv3 to import routes from other OSPFv3 processes
      1) Enable OSPFv3 to import routes from other OSPFv3 processes
      2) Display related information
      3) Debug
   （4） Configure other OSPFv3 protocol parameters
      1) Configure the OSPFv3 routing protocol priority
      2) Configure the OSPFv3 stub area and default route cost
      3) Configure OSPFv3 virtual links
      4) Configure the interface priority for designated Layer 3 switch election
3. Disable OSPFv3

1. Enable OSPFv3 Protocol

The basic OSPFv3 configuration on the Layer 3 switch is simple: normally you only need to enable OSPFv3 and enable it on the interfaces, and the OSPFv3 protocol parameters use their default values. To modify the OSPFv3 protocol parameters, see "2. Configure OSPFv3 parameters".

Command	Explanation
Global Mode	
[no] router IPv6 ospf <*tag*>	Initializes the OSPFv3 routing process and enters OSPFv3 Router Mode to configure the process. The **no router IPv6 ospf <*tag*>** command stops the process. (Required)
OSPFv3 Router Mode	
router-id <*router_id*> **no router-id**	Configures the router ID of the OSPFv3 process. The **no router-id** command resets the ID to 0.0.0.0. (Required)
[no] passive-interface<*ifname*>	Configures the interface to receive but not send OSPFv3 packets. The **no passive-interface<*ifname*>** command cancels the configuration.
Interface Mode	

[no] IPv6 router ospf {area <*area-id*> [instance-id <*instance-id*> | tag <*tag*> [instance-id <*instance-id*>]] | tag <*tag*> area <*area-id*> [instance-id <*instance-id*>]}	Enables OSPFv3 routing on the interface. The **no IPv6 router ospf {area <*area-id*> [instance-id <*instance-id*> | tag <*tag*> [instance-id <*instance-id*>]] | tag <*tag*> area <*area-id*> [instance-id <*instance-id*>]}** command cancels the configuration.

2. Configure OSPFv3 parameters

(1) Configure OSPFv3 packet sending mechanism parameters

1) Set the OSPFv3 interface to receive only

2) Configure the cost for sending packets from the interface

Command	Explanation
Interface Mode	
IPv6 ospf cost <*cost*> [instance-id <*id*>] **no IPv6 ospf cost [instance-id <*id*>]**	Specifies the OSPFv3 cost of the interface. The **no IPv6 OSPF cost [instance-id <*id*>]** command restores the default setting.

3) Configure OSPFv3 packet-sending timer parameters (the Hello interval on broadcast interfaces, the neighbor dead interval, the LSA transmission delay, and the LSA retransmission interval)

Command	Explanation
Interface Mode	
IPv6 ospf hello-interval <*time*> [instance-id <*id*>] **no IPv6 ospf hello-interval [instance-id <*id*>]**	Sets the interval for sending Hello packets. The **no IPv6 ospf hello-interval [instance-id <*id*>]** command restores the default setting.
IPv6 ospf dead-interval <*time*> [instance-id <*id*>] **no IPv6 ospf dead-interval [instance-id <*id*>]**	Sets the interval after which a neighboring Layer 3 switch is considered down. The **no IPv6 ospf dead-interval [instance-id <*id*>]** command restores the default setting.
IPv6 ospf transit-delay <*time*> [instance-id <*id*>] **no IPv6 ospf transit-delay [instance-id <*id*>]**	Sets the delay before link-state advertisements are sent. The **no IPv6 ospf transit-delay [instance-id <*id*>]** command restores the default setting.
IPv6 ospf retransmit <*time*> [instance-id <*id*>] **no IPv6 ospf retransmit [instance-id <*id*>]**	Sets the interval for retransmitting link-state advertisements to neighboring Layer 3 switches. The **no IPv6 ospf retransmit [instance-id <*id*>]** command restores the default setting.

(2) Configure OSPFv3 route redistribution parameters

Configure OSPFv3 route redistribution parameters

Command	Explanation
OSPFv3 Router Mode	
[no]redistribute {kernel |connected| static| rip| isis| bgp} [metric<*value*>] [metric-type {1|2}][route-map<*word*>]	Imports routes discovered by other protocols and static routes into OSPFv3 as external routing information. The **no redistribute {kernel |connected| static| rip| isis| bgp} [metric<*value*>] [metric-type {1|2}][route-map<*word*>]** command cancels the import of external routing information.

(3) Configure OSPFv3 to import routes from other OSPFv3 processes

1) Enable OSPFv3 to import routes from other OSPFv3 processes

Command	Explanation
OSPFv3 Router Mode	

redistribute ospf [<*process-id*>] [metric<*value*>] [metric-type {1|2}] [route-map<*word*>] **no redistribute ospf [<*process-id*>] [metric<*value*>] [metric-type {1|2}] [route-map<*word*>]**	Enables or disables the import of routes from other OSPFv3 processes.

2) Display related information

Command	Explanation
Admin Mode and Configuration Mode	
show ipv6 ospf [<*process-id*>] redistribute	Displays the configuration of the OSPFv3 process for importing external routes.

3) Debug

Command	Explanation
Admin Mode	
debug ipv6 ospf redistribute message send **no debug ipv6 ospf redistribute message send** **debug ipv6 ospf redistribute route receive** **no debug ipv6 ospf redistribute route receive**	Enables or disables debugging of messages sent when the OSPFv3 process redistributes routes to other OSPFv3 processes. Enables or disables debugging of route messages received by the OSPFv3 process from the NSM.

(4) Configure other OSPFv3 protocol parameters

1) Configure the OSPFv3 stub area and default route cost

2) Configure OSPFv3 virtual links

Command	Explanation
OSPFv3 Router Mode	
timers spf <*spf-delay*> <*spf-holdtime*> **no timers spf**	Configures the OSPFv3 SPF timer. The **no timers spf** command restores the default value.
area <*id*> stub [no-summary] **no area <*id*> stub [no-summary]** **area <*id*> default-cost <*cost*>** **no area <*id*> default-cost** **area <*id*> virtual-link A.B.C.D [instance-id <*instance-id*> INTERVAL]** **no area <*id*> virtual-link A.B.C.D [| INTERVAL]**	Configures OSPFv3 area parameters (stub area and virtual link). The no command restores the default values.

4) Configure the interface priority for designated Layer 3 switch (DR) election

Command	Explanation
Interface Mode	
IPv6 ospf priority <*priority*> [instance-id <*id*>] **no IPv6 ospf priority [instance-id <*id*>]**	Sets the interface priority for designated Layer 3 switch election. The **no IPv6 ospf priority [instance-id <*id*>]** command restores the default setting.

3. Disable OSPFv3

Command	Explanation
Global Mode	

no router IPv6 ospf ospf [*<tag>*]	Disables the OSPFv3 routing protocol.

12.5.3 OSPFv3 Examples

Example 1: OSPF autonomous system

This example uses an OSPF autonomous system consisting of five switches.

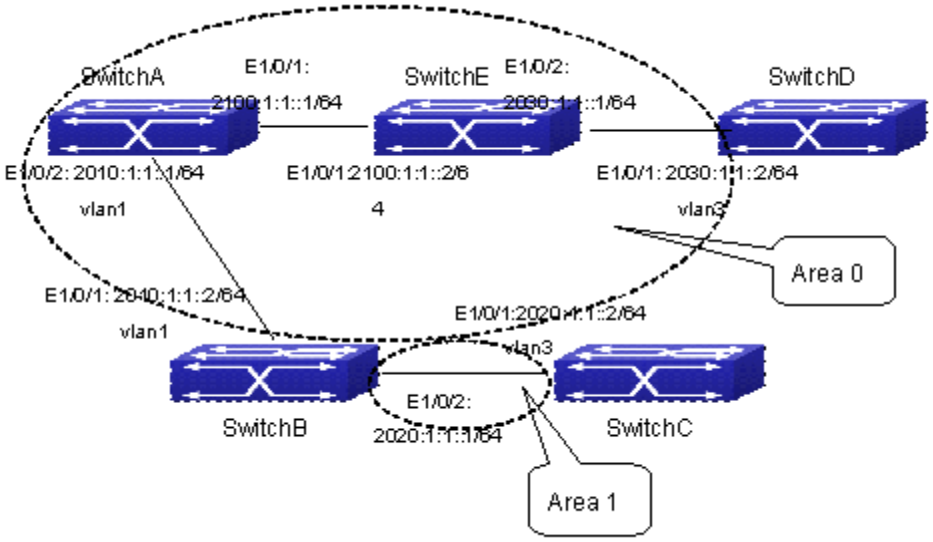


Figure 12-8 Network topology of the OSPF autonomous system

The configurations of Layer 3 switches SwitchA to SwitchE are shown below:

Layer 3 SwitchA:

Enable OSPFv3 and configure the router ID

```
SwitchA(config)#router IPv6 ospf
SwitchA (config-router)#router-id 192.168.2.1
```

Configure the IPv6 address of interface vlan1 and its OSPFv3 area

```
SwitchA#config
SwitchA(config)# interface vlan 1
SwitchA(config-if-vlan1)# IPv6 address 2010:1:1::1/64
SwitchA(config-if-vlan1)# IPv6 router ospf area 0
SwitchA(config-if-vlan1)#exit
```

Configure the IP address of interface vlan2 and its OSPFv3 area

```
SwitchA(config)# interface vlan 2
SwitchA(config-if-vlan2)# IPv6 address 2100:1:1::1/64
SwitchA(config-if-vlan2)# IPv6 router ospf area 0
SwitchA (config-if-vlan2)#exit
SwitchA(config)#exit
SwitchA#
```

Layer 3 SwitchB:

Enable OSPFv3 and configure the router ID

```
SwitchB(config)#router IPv6 ospf
SwitchB (config-router)#router-id 192.168.2.2
```

Configure the IPv6 addresses of interface vlan1 and VLAN2 and their OSPFv3 areas

```
SwitchB#config
SwitchB(config)# interface vlan 1
SwitchB(config-if-vlan1)# IPv6 address 2010:1:1::2/64
SwitchB(config-if-vlan1)# IPv6 router ospf area 0
```

```
SwitchB(config-if-vlan1)#exit
SwitchB(config)# interface vlan 3
SwitchB(config-if-vlan3)# IPv6 address 2020:1:1::1/64
SwitchB(config-if-vlan3)# IPv6 router ospf area 1
SwitchB(config-if-vlan3)#exit
SwitchB(config)#exit
SwitchB#
```

Layer 3 SwitchC:

Enable OSPFv3 and configure the router ID

```
SwitchC(config)#router IPv6 ospf
SwitchC(config-router)#router-id 192.168.2.3
```

Configure the IPv6 address of interface vlan3 and its OSPFv3 area

```
SwitchC#config
SwitchC(config)# interface vlan 3
SwitchC(config-if-vlan3)# IPv6 address 2020:1:1::2/64
SwitchC(config-if-vlan3)# IPv6 router ospf area 1
SwitchC(config-if-vlan3)#exit
SwitchC(config)#exit
SwitchC#
```

Layer 3 SwitchD:

Enable OSPFv3 and configure the router ID

```
SwitchD(config)#router IPv6 ospf
SwitchD(config-router)#router-id 192.168.2.4
```

Configure the IPv6 address of interface vlan3 and its OSPFv3 area

```
SwitchD#config
SwitchD(config)# interface vlan 3
SwitchD(config-if-vlan3)# IPv6 address 2030:1:1::2/64
SwitchD(config-if-vlan3)# IPv6 router ospf area 0
SwitchD(config-if-vlan3)#exit
SwitchD(config)#exit
SwitchD#
```

Layer 3 SwitchE:

Enable OSPFv3 and configure the router ID

```
SwitchE(config)#router IPv6 ospf
SwitchE(config-router)#router-id 192.168.2.5
```

Configure the interface IPv6 address and its OSPFv3 area

```
SwitchE#config
SwitchE(config)# interface vlan 2
SwitchE(config-if-vlan2)# IPv6 address 2100:1:1::2/64
SwitchE(config-if-vlan2)# IPv6 router ospf area 0
SwitchE(config-if-vlan2)#exit
```

Configure the IPv6 address of interface VLAN3 and its area

```
SwitchE(config)# interface vlan 3
SwitchE(config-if-vlan3)# IPv6 address 2030:1:1::1/64
SwitchE(config-if-vlan3)# IPv6 router ospf area 0
SwitchE(config-if-vlan3)#exit
SwitchE(config)#exit
SwitchE#
```

12.5.4 OSPFv3 Troubleshooting

When OSPFv3 is configured and running, problems such as faulty physical connections or configuration errors may prevent OSPFv3 from working. Pay attention to the following:

- ☞ First, ensure that the physical connections are correct.
- ☞ Second, ensure that the interface and the link protocol are up (use the **show interface** command).
- ☞ Configure IPv6 addresses in different network segments on each interface.
- ☞ Enable OSPFv3 (use the **router IPv6 OSPF** command), and configure the OSPFv3 area on the corresponding interfaces.
- ☞ Then, consider the following OSPFv3 characteristics: The OSPFv3 backbone area (area 0) must be contiguous; if it is not, use virtual links to keep it contiguous. Non-zero areas can connect to each other only through area 0, not directly. On an area border Layer 3 switch, some interfaces belong to area 0 and the others belong to non-zero areas. On multi-access networks such as broadcast networks, a DR must be elected. The router ID of an OSPFv3 process must not be 0.0.0.0.

12.6 BGP4+

12.6.1 Introduction to MBGP4+

MBGP4+ is the multiprotocol BGP (Multiprotocol Border Gateway Protocol) extension for IPv6. For an introduction to BGP, see the BGP chapter of this manual. Unlike RIPng and OSPFv3, BGP has no separate protocol for IPv6; instead, the original BGP is extended with address families. The MBGP4+ extensions to BGP are mainly the following:

a. The configured neighbor address can be an IPv6 address.

b. IPv6 unicast address family configuration is added.

12.6.2 MBGP4+ Configuration Task List

MBGP4+ configuration task list:

1. Configure IPv6 neighbors
2. Configure and activate the IPv6 address family
3. Configure redistribution of OSPFv3 routes into MBGP4+
    1) Enable redistribution of OSPFv3 routes into MBGP4+
    2) Display and debug the configuration of OSPFv3 route redistribution into MBGP4+

1. Configure IPv6 neighbors

Command	Explanation
BGP Router Mode	
neighbor <*X:X::X:X*> remote-as <*as-id*>	Configures an IPv6 neighbor.

2. Configure and activate IPv6 address family

Command	Explanation
BGP Router Mode	
address-family IPv6 unicast	Enters the IPv6 unicast address family.
BGP Address Family Mode	
neighbor <*X:X::X:X*> activate **no neighbor <*X:X::X:X*> activate**	Activates or deactivates the address family for the IPv6 neighbor.
exit-address-family	Exits the address family configuration mode.

3. Configure redistribution of OSPFv3 routes into MBGP4+

(1) Enable redistribution of OSPFv3 routes into MBGP4+

Command	Explanation
BGP Router Mode	
redistribute ospf [<*process-tag*>] [route-map<*word*>] **no redistribute ospf [<*process-tag*>]**	Enables or disables redistribution of OSPFv3 routes into MBGP4+.

(2) Display and debug the configuration of OSPFv3 route redistribution into MBGP4+

Command	Explanation
Admin Mode and Configuration Mode	
show ipv6 bgp redistribute	Displays the configuration of MBGP4+ routes redistributed from other routing protocols.
Admin Mode	
debug ipv6 bgp redistribute message send **no debug ipv6 bgp redistribute message send** **debug ipv6 bgp redistribute route receive** **no debug ipv6 bgp redistribute route receive**	Enables or disables debugging of messages sent by MBGP4+ for the redistribution of OSPFv3 routes. Enables or disables debugging of messages received from the NSM.

12.6.3 MBGP4+ Examples

SwitchB, SwitchC, and SwitchD are in AS200, and SwitchA is in AS100. SwitchA and SwitchB are on the same network segment. SwitchB and SwitchD are not physically connected.

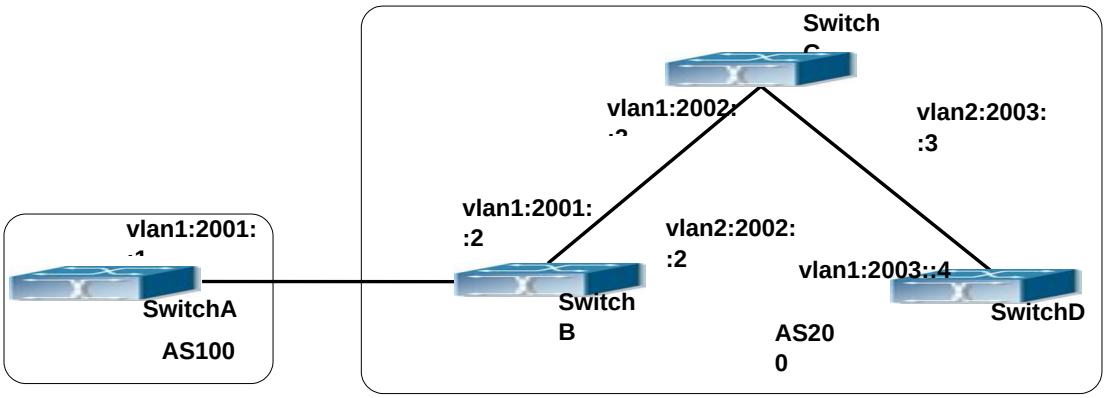


Figure 12-9 BGP network topology

SwitchA configuration:

```
SwitchA(config)#router bgp 100
SwitchA(config-router)#bgp router-id 1.1.1.1
SwitchA(config-router)#neighbor 2001::2 remote-as 200
SwitchA(config-router)#address-family IPv6 unicast
SwitchA(config-router-af)#neighbor 2001::2 activate
SwitchA(config-router-af)#exit-address-family
SwitchA(config-router-bgp)#exit
SwitchA(config)#
```

SwitchB configuration:

```
SwitchB(config)#router bgp 200
SwitchA(config-router)#bgp router-id 2.2.2.2
SwitchB(config-router)#neighbor 2001::1 remote-as 100
SwitchB(config-router)#neighbor 2002::3 remote-as 200
SwitchB(config-router)#neighbor 2003::4 remote-as 200
```

```
SwitchB(config-router)#address-family IPv6 unicast
SwitchB(config-router-af)#neighbor 2001::1 activate
SwitchB(config-router-af)#neighbor 2002::3 activate
SwitchB(config-router-af)#neighbor 2003::4 activate
SwitchB(config-router-af)#exit-address-family
SwitchB(config-router)#exit
SwitchB(config)#
```

SwitchC configuration:

```
SwitchC(config)#router bgp 200
SwitchA(config-router)#bgp router-id 2.2.2.2
SwitchC(config-router)#neighbor 2002::2 remote-as 200
SwitchC(config-router)#neighbor 2003::4 remote-as 200
SwitchC(config-router)#address-family IPv6 unicast
SwitchC(config-router-af)#neighbor 2002::2 activate
SwitchC(config-router-af)#neighbor 2003::4 activate
SwitchC(config-router-af)#exit-address-family
SwitchC(config-router-bgp)#exit
```

SwitchD configuration:

```
SwitchD(config)#router bgp 200
SwitchA(config-router)#bgp router-id 2.2.2.2
SwitchD(config-router)#neighbor 2003::3 remote-as 200
SwitchD(config-router)#neighbor 2002::2 remote-as 200
SwitchD(config-router)#address-family IPv6 unicast
SwitchD(config-router-af)#neighbor 2002::2 activate
SwitchD(config-router-af)#neighbor 2003::3 activate
SwitchD(config-router-af)#exit-address-family
SwitchD(config-router)#exit
```

Here, the connection between SwitchB and SwitchA is EBGP, and the connection between SwitchC and SwitchD is IBGP. A BGP connection can be established between SwitchB and SwitchD without a physical link, provided that each switch has a route to the other. The route can be obtained through static routing or an IGP.

12.6.4 MBGP4+ Troubleshooting

Same as the corresponding section of the BGP chapter.

12.7 IPv6 Black Hole Routing

12.7.1 Introduction to Black Hole Routing

Black hole routing is a special kind of static routing that drops all packets matching the route.

12.7.2 IPv6 Black Hole Routing Configuration Tasks

1. Enable the IPv6 function
2. Configure IPv6 black hole routing

1. Enable the IPv6 function

Command	Explanation
Global Mode	
ipv6 enable	Enables the IPv6 function on the switch.

2. Configure IPv6 black hole routing

Command	Explanation

Global Mode	
ipv6 route <*ipv6-prefix/prefix-length*> null0 [<*precedence*>] **no ipv6 route <*ipv6-prefix/prefix-length*> null0**	Configures a static IPv6 black hole route. The no command removes the specified configuration.

12.7.3 Black Hole Routing Configuration Examples

Example 1: IPv6 black hole routing

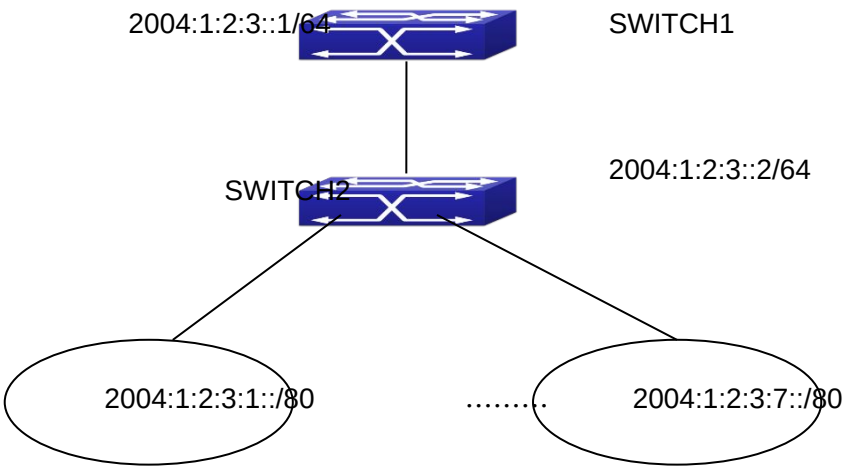


Figure 12-10 IPv6 Black Hole Routing Configuration Example

As shown in the figure, eight interfaces on Switch 2 are configured as Layer 3 VLAN interfaces for access. Their network addresses are 2004:1:2:3:1/80 to 2004:1:2:3:7/80. A default route pointing to Switch 1 is configured on Switch 2, and a route back to Switch 2 for network address 2004:1:2:3::/64 is configured on Switch 1. Normally, this configuration works well. However, suppose one of the Layer 3 interfaces on Switch 2 goes down, for example, the interface for 2004:1:2:3:1/80. When packets for that network arrive at VLAN1 on Switch 2, there is no matching route for them, so the switch forwards them back to Switch 1 according to the default route. When Switch 1 receives these packets, it forwards them back to Switch 2, which creates a routing loop. To solve this problem, configure black hole routing on Switch 2:

```
ipv6 route 2004:1:2:3::/64 null0 50
```

Switch 2 then drops the packets from interface VLAN1 that match the black hole route, and the routing loop is prevented.

The configuration steps are as follows:

```
Switch#config
Switch(config)#ipv6 route 2004:1:2:3::/64 null0 50
```

12.7.4 Black Hole Routing Troubleshooting

Black hole routing may not work because of problems such as an incorrect network address mask or administrative distance. Pay attention to the following:

- IPv6 must be enabled before IPv6 black hole routing can work.
- Use a longer network address mask for black hole routes than for normal routes, so that black hole routes do not interfere with other routes.
- If a black hole route has the same network address mask as another route, set a lower distance for the black hole route.

If the problem persists, run the **show ip route distance**, **show ip route fib**, and **show l3** commands, and send their output to EnGenius technical support.

12.8 IPv6 Multicast Protocol

12.8.1 PIM-DM6

12.8.1.1 Introduction to PIM-DM6

PIM-DM6 (Protocol Independent Multicast, Dense Mode) is the IPv6 version of PIM-DM. It is a dense-mode multicast routing protocol suited to small networks, in which multicast group members are relatively dense. It is identical to the IPv4 version of PIM-DM except that it uses IPv6 addresses. Therefore, this chapter does not distinguish between PIM-DM and PIM-DM6; unless otherwise specified, PIM-DM in this text refers to the IPv6 version.

As IPv6 networks develop, IPv6 multicast traffic sometimes has to cross networks that do not support IPv6 multicast, which requires tunneling. Therefore, PIM-DM6 on the switch can be configured on tunnels and can cross networks without IPv6 multicast support in unicast packets encapsulated in IPv4.

The PIM-DM working process can be summarized as neighbor discovery, flooding-prune, and graft.

1. Neighbor Discovery

When a PIM-DM router starts, it sends Hello messages to discover its neighbors. Network nodes running PIM-DM use Hello messages to contact each other. PIM-DM Hello messages are sent periodically.

2. Flooding-Prune

PIM-DM assumes that all hosts on the network are ready to receive multicast data. When a multicast source S begins sending data to a multicast group G, a router that receives the multicast packet first performs an RPF check based on the unicast routing table. If the check passes, the router creates an (S, G) entry and forwards the multicast packet to all downstream PIM-DM nodes (flooding). If the RPF check fails, that is, the multicast packet arrived on the wrong interface, the packet is discarded. After this procedure, every node in the PIM-DM multicast domain has an (S, G) entry. If a downstream node has no multicast group members, it sends a Prune message to its upstream node, asking it to stop forwarding data for this multicast group. After receiving the Prune message, the upstream node deletes the corresponding interface from the outgoing interface list of the (S, G) multicast forwarding entry. Through this process, a shortest path tree (SPT) rooted at source S is established. The prune process is initiated by downstream routers.

This process is called the flooding-prune process. Each pruned node also has a timeout mechanism: when the prune times out, the router restarts the flooding-prune process. PIM-DM performs flooding-prune periodically.

3. RPF Check

Using the RPF check, PIM-DM builds a multicast forwarding tree rooted at the data source from the existing unicast routing table. When a multicast packet arrives, the router first checks whether it arrived on the correct path. If the arrival interface is the interface toward the multicast source according to unicast routing, the packet is considered to have arrived on the correct path; otherwise, the packet is discarded as redundant. The unicast routing information used for this check can come from any unicast routing protocol, such as RIP or OSPF; PIM-DM does not depend on any specific unicast routing protocol.

4. Assert Mechanism

If two multicast routers, A and B, on the same LAN segment each have a receiving path to multicast source S, both forward the multicast packets from S onto the LAN, and the downstream multicast router C receives two identical copies of each packet. When a router detects this situation, a single

forwarder is selected through the assert mechanism, which uses Assert packets to select the optimal forwarding path. If two or more paths have the same priority and cost, the node with the larger IP address is selected as the upstream neighbor for the (S, G) entry and is responsible for forwarding the (S, G) multicast packets.

5. Graft

When a pruned downstream node needs to resume forwarding, it sends a Graft message to notify the upstream node to resume forwarding multicast data.

12.8.1.2 PIM-DM6 Configuration Task List

1．Enable PIM-DM (Required)

2．Configure static multicast routing entries (Optional)

3．Configure additional PIM-DM parameters (Optional)

(1) Configure parameters for PIM-DM interfaces

1) Configure the interval for PIM-DM hello messages

2) Configure the interval for PIM-DM state-refresh messages

3) Configure the boundary interfaces

4) Configure the management boundary

4．Disable the PIM-DM protocol

1. Enable the PIM-DM protocol

Enabling PIM-DM on the switch takes two steps: first enable PIM multicast routing in Global Mode, and then enable PIM-DM on the specific interfaces.

Command	Explanation
Global Mode	
ipv6 pim multicast-routing	Enables PIM-DM multicast routing globally. To enable PIM-DM on specific interfaces, the following command must also be issued.

Enable PIM-DM on the specific interface:

Command	Explanation
Interface Mode	
ipv6 pim dense-mode	Enables PIM-DM on the specified interface (required).

2. Configure static multicast routing entries

Command	Explanation
Global Mode	
ipv6 mroute *<X:X::X:X>* *<X:X::X:X>* *<ifname>* *<.ifname>* **no ipv6 mroute *<X:X::X:X>* *<X:X::X:X>* [*<ifname>* *<.ifname>*]**	Configures an IPv6 static multicast routing entry. The no command removes the specified routing entry.

3. Configure additional PIM-DM parameters

(1) Configure parameters for PIM-DM interfaces

1) Configure the interval for PIM-DM hello messages

Command	Explanation
Interface Mode	

ipv6 pim hello-interval <*interval*> **no ipv6 pim hello-interval**	Configures the interval for PIM-DM Hello messages. The no command restores the default value.

2) Configure the interval for PIM-DM state-refresh messages

Command	Explanation
Interface Mode	
ipv6 pim state-refresh origination-interval **no ipv6 pim state-refresh origination-interval**	Configures the interval for sending PIM-DM state-refresh packets. The no command restores the default value.

3) Configure the boundary interfaces

Command	Explanation
Interface Mode	
ipv6 pim bsr-border **no ipv6 pim bsr-border**	Configures the interface as a PIM-DM6 boundary. State Refresh messages are neither sent nor received on a boundary interface, and the network connected to the interface is treated as a directly connected network. The no command removes the configuration.

4) Configure the management boundary

Command	Explanation
Interface Mode	
ipv6 pim scope-border <*500-599*>|<*acl_name*> **no ipv6 pim scope-border**	Configures the PIM-DM6 management boundary on the interface and applies an ACL to it. By default, ffx0::/13 is considered the scope of the management group. If an ACL is configured, the scope permitted by the ACL is the scope of the management group. acl_name must be a standard IPv6 ACL name. The no command removes the configuration.

4. Disable the PIM-DM protocol

Command	Explanation
Interface Mode	
no ipv6 pim dense-mode	Disables PIM-DM on the specified interface.
Global Mode	
no ipv6 pim multicast-routing	Disables PIM-DM globally.

12.8.1.3 PIM-DM6 Typical Application

As shown in the following figure, add the ports of Switch A and Switch B to the corresponding VLANs, and enable PIM-DM on each VLAN interface.

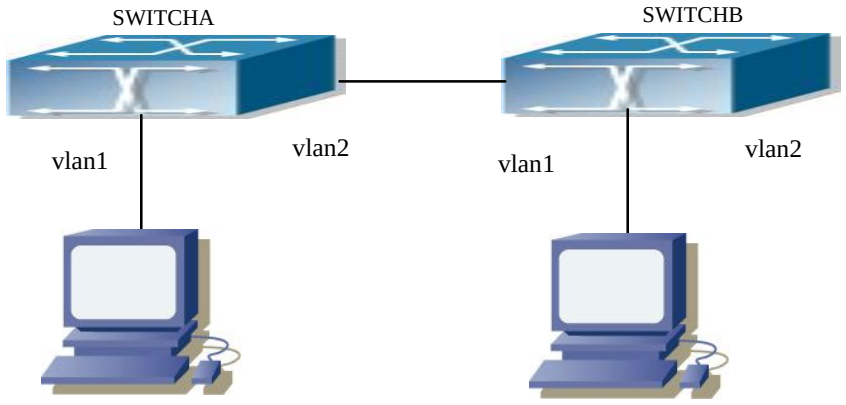


Figure 12-11 Typical PIM-DM environment

The configuration procedure for SwitchA and SwitchB is as follows:

(1) Configure SwitchA:

```
Switch(config)#ipv6 pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ipv6 address 2000:10:1:1::1/64
Switch(Config-if-Vlan1)#ipv6 pim dense-mode
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan2
Switch(Config-if-Vlan2)#ipv6 address 2000:12:1:1:: 1/64
Switch(Config-if-Vlan2)#ipv6 pim dense-mode
```

(2) Configure SwitchB:

```
Switch(config)#ip pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ipv6 address 2000:12:1:1::2/64
Switch(Config-if-Vlan1)#ipv6 pim dense-mode
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)#ipv6 address 2000:20:1:1::1/64
Switch(Config-if-Vlan2)#ipv6 pim dense-mode
```

12.8.1.4 PIM-DM6 Troubleshooting

PIM-DM may not work properly because of problems such as faulty physical connections or incorrect configuration. Pay attention to the following:

- Ensure that the physical connections are correct.
- Ensure that the interface and link protocol are up (use the **show interface** command).
- Ensure that PIM is enabled in Global Mode (use the **ipv6 pim multicast-routing** command).
- Enable PIM-DM on the interface (use the **ipv6 pim dense-mode** command).

The multicast protocol uses unicast routes for RPF checks, so first ensure that the unicast routes are correct. If the PIM-DM problem persists, use debug commands such as **debug ipv6 pim**, record the debug information for three minutes, and send it to EnGenius technical support.

12.8.2 PIM-SM6

12.8.2.1 Introduction to PIM-SM6

PIM-SM6 (Protocol Independent Multicast, Sparse Mode) is the IPv6 version of PIM-SM. It is a sparse-mode multicast routing protocol used mainly in large networks where group members are sparsely and widely distributed. It is identical to the IPv4 version of PIM-SM except that it uses IPv6

addresses. Therefore, this chapter does not distinguish between PIM-SM and PIM-SM6; unless otherwise specified, PIM-SM in this text refers to the IPv6 version. Unlike the flooding-prune approach of dense mode, PIM-SM assumes that no host needs to receive multicast data packets. A PIM-SM router forwards multicast data packets to a host only on explicit request.

By setting up an RP (Rendezvous Point) and a BSR (Bootstrap Router), PIM-SM advertises multicast information to all PIM-SM routers and uses router Join/Prune messages to build an RP-rooted shared tree (RPT). This reduces the network bandwidth used by data packets and control messages and lowers the processing overhead of routers. Multicast data flows along the shared tree to the network segments where the multicast group members are located. When the data traffic reaches a certain level, the multicast data stream can be switched to a source-based shortest path tree (SPT) to reduce network delay. PIM-SM does not depend on any specific unicast routing protocol; it performs RPF checks using the existing unicast routing table.

1. PIM-SM Working Principles

The PIM-SM working process mainly includes neighbor discovery, RPT creation, multicast source registration, and SPT switchover. The neighbor discovery mechanism is the same as in PIM-DM and is not described again here.

(1) Creation of RP Shared Tree (RPT)

When a host joins multicast group G, the leaf router directly connected to the host learns from IGMP messages that there is a receiver for group G. It then determines the rendezvous point (RP) for group G and sends a Join message to the upstream node toward the RP. Every router on the path from the leaf router to the RP creates a (*, G) entry, indicating that traffic from any source to group G matches this entry. When the RP receives traffic sent to group G, the traffic travels along the established path to the leaf router and then reaches the host. In this way, the RPT rooted at the RP is created.

(2) Multicast Source Registration

When multicast source S sends a multicast packet to multicast group G, the PIM-SM multicast router directly connected to S encapsulates the multicast packet in a Register message and unicasts it to the corresponding RP. If there is more than one PIM-SM multicast router on a network segment, the DR (Designated Router) forwards the multicast packet.

(3) SPT Switchover

When a multicast router finds that the rate of multicast packets from the RP with destination address G exceeds a threshold, it sends a Join message to the upstream node toward the source, which switches the traffic from the RPT to the SPT.

2. Preparation Before PIM-SM Configuration

(1) Configure Candidate RPs

A PIM-SM network can have more than one RP (candidate RP), and each C-RP (candidate RP) forwards multicast packets for destination addresses in a certain range. Configuring multiple candidate RPs provides RP load balancing. There is no master/slave relationship among RPs. After receiving the candidate RP information advertised by the BSR, all multicast routers use the same algorithm to determine the RP for a given multicast group.

Note that one RP can serve more than one multicast group, or even all multicast groups. However, each multicast group can correspond to only one RP at any moment; it cannot correspond to multiple RPs at the same time.

(2) Configure the BSR

As the management core of a PIM-SM network, the BSR collects the information sent by candidate RPs and advertises it.

There can be only one BSR in a network, but several candidate BSRs can be configured. With this arrangement, if the BSR fails, another one can take over. The BSR is elected automatically from the C-BSRs.

12.8.2.2 PIM-SM6 Configuration Task List

1．Enable PIM-SM (Required)

2．Configure static multicast routing entries (Optional)

3．Configure additional parameters for PIM-SM (Optional)

(1) Configure parameters for PIM-SM interfaces

1) Configure the interval for PIM-SM hello messages

2) Configure the holdtime for PIM-SM hello messages

3) Configure ACL for PIM-SM6 neighbors

4) Configure the interface as a PIM-SM6 boundary interface

5) Configure the interface as a PIM-SM6 management boundary

(2) Configure global PIM-SM parameters

1) Configure the switch as a candidate BSR

2) Configure the switch as a candidate RP

5) Configure static RP

3) Configure the cache time of kernel multicast routes

4．Disable the PIM-SM protocol

1. Enable the PIM-SM protocol

To enable PIM-SM on a Layer 3 switch, enable PIM6 in Global Mode and then enable PIM-SM on specific interfaces in Interface Mode.

Command	Explanation
Global Mode	
[no] ipv6 pim multicast-routing	Enables PIM-SM6 for all interfaces. (To make PIM-SM work on specific interfaces, the following command must also be issued.) (Required)

Enable PIM-SM on specific interfaces:

Command	Explanation
Interface Mode	
[no] ipv6 pim sparse-mode [passive]	Enables PIM-SM on the specified interface. The no command disables PIM-SM (required).

2. Configure static multicast routing entries

Command	Explanation
Global Mode	
ipv6 mroute *<X:X::X:X>* *<X:X::X:X>* *<ifname>* *<.ifname>* **no ipv6 mroute *<X:X::X:X>* *<X:X::X:X>* [*<ifname>* *<.ifname>*]**	Configures a static multicast routing entry. The no command removes the specified static multicast routing entry.

3. Configure additional PIM-SM parameters

(1) Configure parameters for PIM-SM interfaces

1) Configure the interval for PIM-SM hello messages

Command	Explanation
Interface Mode	

ipv6 pim hello-interval <*interval*> **no ipv6 pim hello-interval**	Configures the interval for PIM-SM Hello messages. The no command restores the default interval.

2) Configure the hold time for PIM-SM6 hello messages

Command	Explanation
Interface Mode	
ipv6 pim hello-holdtime <*value*> **no ipv6 pim hello-holdtime**	Configures the value of the holdtime field in PIM-SM Hello messages. The no command restores the default holdtime.

3) Configure ACL for PIM-SM6 neighbors

Command	Explanation
Interface Mode	
ipv6 pim neighbor-filter <a*cess-list-name*> **no ipv6 pim neighbor-filter <a*cess-list-name*>**	Configures an ACL to filter PIM-SM6 neighbors. If sessions with a neighbor are denied by the ACL, sessions that have been established are dropped immediately and new sessions are not established.

4) Configure the interface as a PIM-SM6 boundary interface

Command	Explanation
Interface Mode	
ipv6 pim bsr-border **no ipv6 pim bsr-border**	Configures the interface as a PIM-SM6 boundary. BSR messages are neither sent nor received on a boundary interface, and the network connected to the interface is treated as a directly connected network. The no command removes the configuration.

5) Configure the interface as a PIM-SM6 management boundary

Command	Explanation
Interface Mode	
ipv6 pim scope-border <*500-599*>|<*acl_name*> **no ipv6 pim scope-border**	Configures the PIM-SM6 management boundary on the interface and applies an ACL to it. By default, ffx0::/13 is considered the scope of the management group. If an ACL is configured, the scope permitted by the ACL is the scope of the management group. acl_name must be a standard IPv6 ACL name. The no command removes the configuration.

(2) Configure global PIM-SM6 parameters

1) Configure the switch as a candidate BSR

Command	Explanation
Global Mode	
**ipv6 pim bsr-candiate {vlan <*vlan_id*>|<*ifname*>	tunnel <*1-50*>}[hash-mask-length] [priority]** **no Ipv6 pim bsr-candiate {vlan <*vlan_id*>|<*ifname*>	tunnel <*1-50*>}[hash-mask-length] [priority]**	Configures the switch as a PIM-SM candidate BSR so that it competes with other candidate BSRs to become the BSR. The no command cancels the candidate BSR configuration.

2) Configure the switch as a candidate RP

Command	Explanation
Global Mode	

ipv6 pim rp-candiate {vlan<*vlan-id*>| loopback<*index*>|<*ifname*>} [<*group range*>] [<*priority*>] **no ipv6 pim rp-candiate**	Configures the switch as a PIM-SM candidate RP so that it competes with other candidate RPs to become the RP. The no command cancels the candidate RP configuration.

3) Configure static RP

Command	Explanation
Global Mode	
ipv6 pim rp-address <*rp-address*> [<*group-range*>] **no ipv6 pim rp-address <*rp-address*> {all|<*group-range*>}**	Configures the address of the candidate RP. The no command removes the candidate RP configuration.

4) Configure the cache time of kernel multicast routes

Command	Explanation
Global Mode	
ipv6 multicast unresolved-cache aging-time <value> **no ipv6 multicast unresolved-cache aging-time**	Configures the cache time of kernel multicast routes. The no command restores the default value.

4. Disable the PIM-SM protocol

Command	Explanation
Interface Mode	
no ipv6 pim sparse-mode	Disables PIM-SM6 on the interface.
Global Mode	
no ipv6 pim sparse-mode	Disables PIM-SM globally.

12.8.2.3 PIM-SM6 Typical Application

As shown in the following figure, add the ports of SwitchA, SwitchB, SwitchC, and SwitchD to the corresponding VLANs, and enable PIM-SM on each VLAN interface.

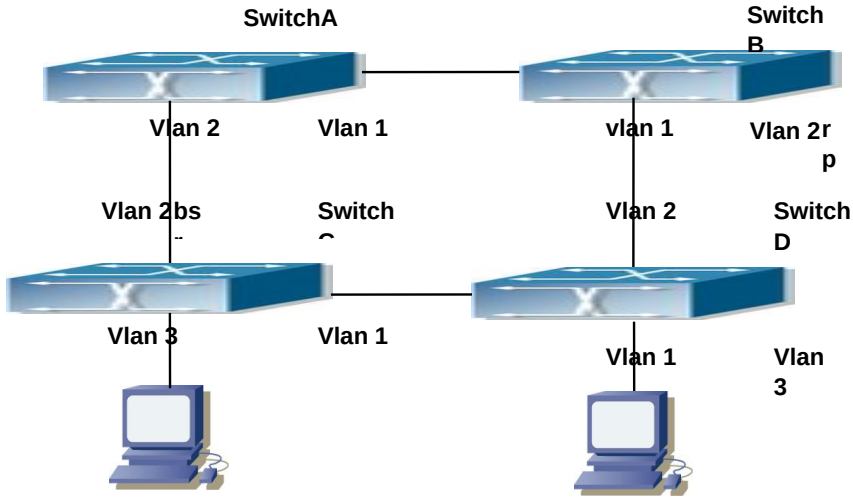


Figure 12-22 Typical PIM-SM environment

The configuration procedure for SwitchA, SwitchB, SwitchC, and SwitchD is as follows:

(1) Configure SwitchA:

```
Switch(config)#ipv6 pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ipv6 address 2000:12:1:1::1/64
Switch(Config-if-Vlan1)#ipv6 pim sparse-mode
```

```
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)#ipv6 address 2000:13:1:1::1/64
Switch(Config-if-Vlan2)#ipv6 pim sparse-mode
```

(2) Configure SwitchB:

```
Switch(config)#ipv6 pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ipv6 address 2000:12:1:1::2/64
Switch(Config-if-Vlan1)#ipv6 pim sparse-mode
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)#ipv6 address2000:24:1:1::2/64
Switch(Config-if-Vlan2)#ipv6 pim sparse-mode
Switch(Config-if-Vlan2)#exit
Switch(config)#ipv6 pim rp-candidate vlan2
```

(3) Configure SwitchC:

```
Switch(config)#ipv6 pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ipv6 address 2000:34:1:1::3/64
Switch(Config-if-Vlan1)#ipv6 pim sparse-mode
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)#ipv6 address 2000:13:1:1::3/64
Switch(Config-if-Vlan2)#ipv6 pim sparse-mode
Switch(Config-if-Vlan2)#exit
Switch(config)#interface vlan 3
Switch(Config-if-Vlan3)#ipv6 address 2000:30:1:1::1/64
Switch(Config-if-Vlan3)#ipv6 pim sparse-mode
Switch(Config-if-Vlan3)#exit
Switch(config)#ipv6 pim bsr-candidate vlan2 30 10
```

(4) Configure SwitchD:

```
Switch(config)#ipv6 pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-if-Vlan1)#ipv6 address 2000:34:1:1::4/64
Switch(Config-if-Vlan1)#ipv6 pim sparse-mode
Switch(Config-if-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-if-Vlan2)#ipv6 address 2000:24:1:1::4/64
Switch(Config-if-Vlan2)#ipv6 pim sparse-mode
Switch(Config-if-Vlan2)#exit
Switch(config)#interface vlan 3
Switch(Config-if-Vlan3)#ipv6 address 2000:40:1:1::1/64
Switch(Config-if-Vlan3)#ipv6 pim sparse-mode
```

12.8.2.4 PIM-SM6 Troubleshooting

PIM-SM may not work properly because of problems such as faulty physical connections or incorrect configuration. Pay attention to the following:

- ☞ Ensure that the physical connections are correct.
- ☞ Ensure that the interface and link protocol are up (use the **show interface** command).
- ☞ The multicast protocol uses unicast routes for RPF checks, so first ensure that the unicast routes are correct.
- ☞ PIM-SM requires an RP and a BSR. First use the **show ipv6 pim bsr-router** command to check whether BSR information exists. If not, check whether there is a unicast route to the BSR.
- ☞ Use the **show ipv6 pim rp-hash** command to check whether the RP information is correct. If there is no RP information, check the unicast routes.

If the PIM-SM problem persists, use debug commands such as **debug ipv6 pim** or **debug ipv6 pim bsr**, record the debug information for three minutes, and send it to EnGenius technical support.

12.8.3 Anycast RP v6

12.8.3.1 Introduction to Anycast RP v6

Anycast RP v6 is a PIM-based technology that provides RP redundancy so that service recovers as quickly as possible when an RP becomes unavailable.

The core concept of Anycast RP v6 is that the same RP address, configured throughout the network, exists on multiple multicast servers. (Most commonly, every device providing Anycast RP configures the RP address with the longest mask on a loopback interface.) Unicast routing ensures that PIM routers always find the nearest RP, which provides a shorter and faster path to the RP in a large network. If the RP in use becomes unavailable, unicast routing ensures that the PIM router quickly finds a path to a new RP so that multicast service recovers in time. Multiple RPs introduce a new problem: if the multicast source and the receivers register with different RPs, some receivers cannot receive data from the multicast source (Register messages naturally go to the nearest RP). Therefore, to keep all RPs in communication, Anycast RP requires the RP nearest to the multicast source to forward the source's Register messages to all other RPs, so that all receivers joined to any RP can find the multicast source.

PIM-based Anycast RP is implemented as follows: every switch configured with Anycast RP maintains an Anycast RP list and uses a separate address to identify each RP to the others. When an Anycast RP device receives a Register message, it sends the Register message to the other Anycast RP devices using its own address as the source address, to notify them of the original destination.

12.8.3.2 Anycast RP v6 Configuration Tasks

1. Enable the Anycast RP v6 function
2. Configure Anycast RP v6

1. Enable the Anycast RP v6 function

Command	Explanation
Global Mode	
ipv6 pim anycast-rp **no ipv6 pim anycast-rp**	Enables the Anycast RP function. (Required) The no command disables the Anycast RP function globally.

2. Configure Anycast RP v6

(1) Configure the candidate RP

Command	Explanation
Global Mode	
ipv6 pim rp-candidate {vlan<*vlan-id*> | loopback<*index*> |<*ifname*>} [<*A:B::C:D*>] [<*priority*>] **no ipv6 pim rp-candidate**	Configures the candidate RP. PIM-SM allows a loopback interface to be a candidate RP. (Required) Note that with Anycast RP, either a loopback interface or a regular Layer 3 VLAN interface can be configured as the candidate RP. To ensure that PIM routers in the network can find where the RP is located, the route of the candidate RP interface must be added to the routing protocol. The no command cancels the candidate RP configured on this router.

(2) Configure the self-rp-address (the RP communication address of this router)

Command	Explanation
Global Mode	
ipv6 pim anycast-rp self-rp-address A:B::C:D **no ipv6 pim anycast-rp self-rp-address**	Configures the self-rp-address of this router (as an RP). This address uniquely identifies this router when it communicates with other RPs. (Required) The **self-rp-address** has two functions: 1 When this router (as an RP) receives a Register message unicast from a DR, it forwards the Register message to all other RPs in the network to notify them of the (S,G) source state. When forwarding the Register message, this router changes its source address to the self-rp-address. 2 When this router (as an RP) receives a Register message unicast from another RP, that is, a Register message whose destination is the self-rp-address of this router, it creates the (S,G) state and sends back a Register-Stop message whose destination address is the source address of the Register message. Note: The self-rp-address must be the address of a Layer 3 interface on this router, but it can be configured even if the interface does not yet exist. The self-rp-address must be unique. The no command cancels the self-rp-address that this router uses to communicate with other RPs.

(3) Configure the other-rp-address (the communication addresses of other RPs)

Command	Explanation
Global Mode	
ipv6 pim anycast-rp <*anycast-rp-addr*> <*other-rp-addr*> **no ipv6 pim anycast-rp <*anycast-rp-addr*> <*other-rp-addr*>**	Configures the anycast-rp-addr on this router (as an RP). This unicast address is the RP address configured on multiple RPs in the network and matches the address of the candidate RP interface (or loopback interface). The **anycast-rp-addr** works as follows: 1 Although multiple anycast-rp-addr addresses can be configured, only the one that matches the currently configured candidate RP address takes effect. Only then does the other-rp-address associated with this anycast-rp-addr take effect. 2 The anycast-rp-addr can be configured even if the corresponding interface does not yet exist. Configures on this router (as an RP) the other-rp-addresses of the other RPs it communicates with. Each unicast address identifies another RP and is used for communication with the local router. The **other-rp-address** has two functions: 1 When this router (as an RP) receives a Register message unicast from a DR, it

	forwards the message to the other RPs in the network to notify all RPs of the (S,G) source state. When forwarding, the router changes the destination address of the Register message to the other-rp-address. 2 Multiple other-rp-addresses can be configured for one anycast-rp-addr. When a Register message is received from a DR, it is forwarded to each of these RPs in turn. The no command cancels the other-rp-address used to communicate with this router.

12.8.3.3 Anycast RP v6 Configuration Examples

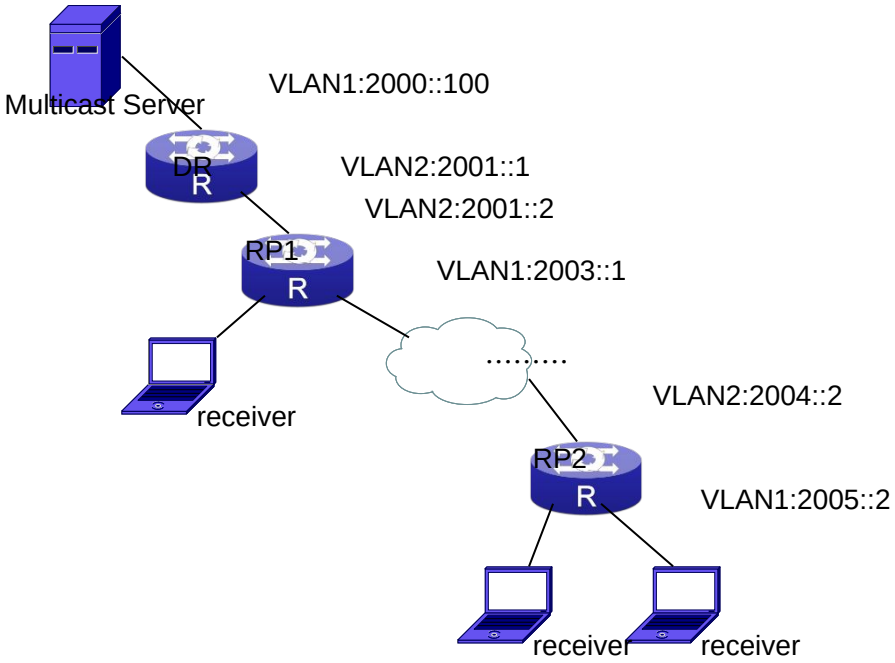


Figure 12-13 Anycast RP v6 function of a router

The configuration steps are as follows:

RP1 Configuration:

```
Switch#config
Switch(config)#interface loopback 1
Switch(Config-if-Loopback1)#ipv6 address 2006::1/128
Switch(Config-if-Loopback1)#exit
Switch(config)#ipv6 pim rp-candidate loopback1
Switch(config)#ipv6 pim bsr-candidate vlan 1
Switch(config)#ipv6 pim multicast-routing
Switch(config)#ipv6 pim anycast-rp
Switch(config)#ipv6 pim anycast-rp self-rp-address 2003::1
Switch(config)#ipv6 pim anycast-rp 2006::1 2004::2
```

RP2 Configuration:

```
Switch#config
Switch(config)#interface loopback 1
Switch(Config-if-Loopback1)#ipv6 address 2006::1/128
Switch(Config-if-Loopback1)#exit
Switch(config)#ipv6 pim rp-candidate loopback1
Switch(config)#ipv6 pim multicast-routing
Switch(config)#ipv6 pim anycast-rp
```

```
Switch(config)#ipv6 pim anycast-rp self-rp-address 2004::2
Switch(config)#ipv6 pim anycast-rp 2006::1 2003::1
```

Note that the route of the loopback interface must be advertised: with MBGP4+, use the network command; with RIPng, use the route command.

12.8.3.4 Anycast RP v6 Troubleshooting

Anycast RP v6 may not work properly because of problems such as faulty physical connections or incorrect configuration. Pay attention to the following:

- ☞ Ensure that the physical connections are correct
- ☞ Ensure that PIM-SM6 is operating normally
- ☞ Ensure that Anycast RP is enabled in Global Mode
- ☞ Ensure that the self-rp-address is configured correctly in Global Mode
- ☞ Ensure that the other-rp-address is configured correctly in Global Mode
- ☞ Ensure that the routes of all interfaces, including the loopback interface used as the RP, are correctly added
- ☞ Use the **show ipv6 pim anycast rp status** command to check whether the Anycast RP configuration is correct

If the Anycast RP problem persists, use debug commands such as **debug ipv6 pim anycast-rp**, record the debug information for three minutes, and send it to EnGenius technical support.

12.8.4 PIM-SSM6

12.8.4.1 Introduction to PIM-SSM6

Source-Specific Multicast (PIM-SSM6) is a newer multicast service protocol. With PIM-SSM6, a multicast session is identified by both the multicast group address and the multicast source address. In SSM6, hosts can join multicast groups as easily and efficiently as in traditional PIM-SM6, but the shared tree and RP management of PIM-SM6 are not needed. In SSM6, an SPT is built for each (S,G), where G is the multicast group address and S is the address of the multicast source sending data to G. An (S,G) pair is called an SSM6 channel. SSM6 is best suited to one-to-many multicast services, such as online sports video channels and news channels. By default, the SSM6 multicast group address range is limited to ff3x::/32; however, this range can be extended as needed.

PIM-SSM6 is also supported in a PIM-DM6 environment.

12.8.4.2 PIM-SSM6 Configuration Task List

Command	Explanation
Global Mode	
ipv6 pim ssm {default|range *<access-list-number>*} **no ipv6 pim ssm**	Configures the address range of PIM-SSM multicast groups. The no command disables this configuration.

12.8.4.3 PIM-SSM6 Configuration Example

As shown in the figure below, the ports of switchA, switchB, switchC, and switchD are assigned to different VLANs, and PIM-SM6 or PIM-DM6 is enabled on all VLAN interfaces. This example uses PIM-SM6.

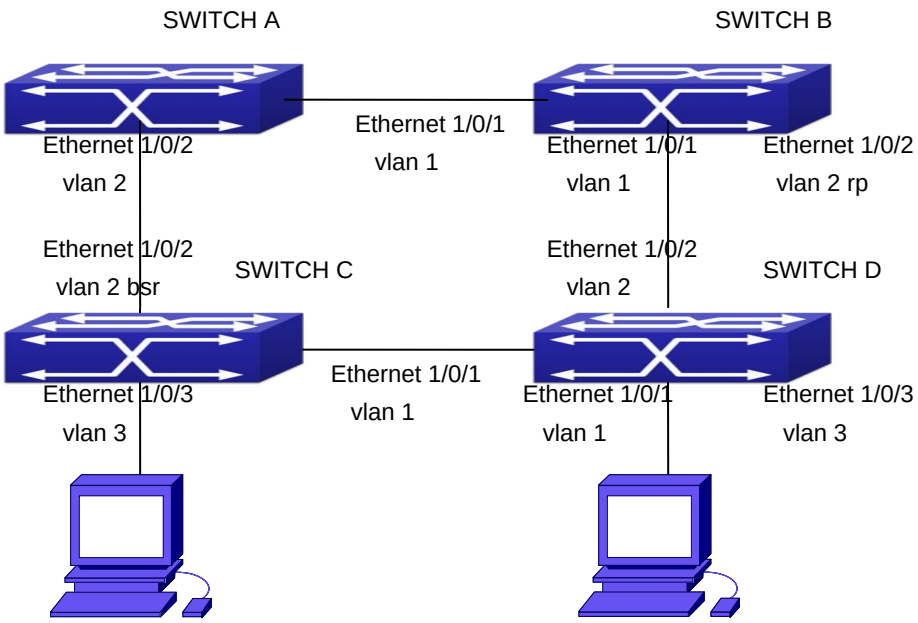


Figure 12-34 Typical PIM-SSM environment

The configurations of switchA, switchB, switchC, and switchD are as follows:

(1) Configuration of switchA:

```
Switch(config)#ipv6 pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-If-Vlan1)# ipv6 address 2000:12:1:1::1/64
Switch(Config-If-Vlan1)# ipv6 pim sparse-mode
Switch(Config-If-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-If-Vlan2)# ipv6 address 2000:13:1:1::1/64
Switch(Config-If-Vlan2)# ipv6 pim sparse-mode
Switch(Config-If-Vlan2)#exit
Switch(config)#ipv6 access-list 500 permit ff1e::1/64
Switch(config)#ip pim ssm range 500
```

(2) Configuration of switchB:

```
Switch(config)#ipv6 pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-If-Vlan1)# ipv6 address 2000:12:1:1::2/64
Switch(Config-If-Vlan1)# ipv6 pim sparse-mode
Switch(Config-If-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-If-Vlan2)# ipv6 address2000:24:1:1::2/64
Switch(Config-If-Vlan2)# ipv6 pim sparse-mode
Switch(Config-If-Vlan2)# exit
Switch(config)# ipv6 pim rp-candidate vlan2
Switch(config)#ipv6 access-list 500 permit ff1e::1/64
Switch(config)#ip pim ssm range 500
```

(3) Configuration of SwitchC:

```
Switch(config)#ipv6 pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-If-Vlan1)# ipv6 address 2000:34:1:1::3/64
Switch(Config-If-Vlan1)# ipv6 pim sparse-mode
Switch(Config-If-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-If-Vlan2)# ipv6 address 2000:13:1:1::3/64
Switch(Config-If-Vlan2)# ipv6 pim sparse-mode
```

```
Switch(Config-If-Vlan2)#exit
Switch(config)#interface vlan 3
Switch(Config-If-Vlan3)# ipv6 address 2000:30:1:1::1/64
Switch(Config-If-Vlan3)# ipv6 pim sparse-mode
Switch(Config-If-Vlan3)# exit
Switch(config)# ipv6 pim bsr-candidate vlan2 30 10
Switch(config)#ipv6 access-list 500 permit ff1e::1/64
Switch(config)#ip pim ssm range 500
```

(4) Configuration of SwitchD:

```
Switch(config)#ipv6 pim multicast-routing
Switch(config)#interface vlan 1
Switch(Config-If-Vlan1)# ipv6 address 2000:34:1:1::4/64
Switch(Config-If-Vlan1)# ipv6 pim sparse-mode
Switch(Config-If-Vlan1)#exit
Switch(config)#interface vlan 2
Switch(Config-If-Vlan2)# ipv6 address 2000:24:1:1::4/64
Switch(Config-If-Vlan2)# ipv6 pim sparse-mode
Switch(Config-If-Vlan2)#exit
Switch(config)#interface vlan 3
Switch(Config-If-Vlan3)# ipv6 address 2000:40:1:1::1/64
Switch(Config-If-Vlan3)# ipv6 pim sparse-mode
Switch(Config-If-Vlan3)#exit
Switch(config)#ipv6 access-list 500 permit ff1e::1/64
Switch(config)#ip pim ssm range 500
```

12.8.4.4 PIM-SSM6 Troubleshooting

PIM-SSM6 may not work because of physical connection failures or misconfiguration. To troubleshoot, check the following:

- ☞ Make sure that the physical links are connected correctly.
- ☞ Make sure that the data link layer is up (use the **show interface** command).
- ☞ Make sure that PIM6 is enabled in Global Mode (see the **ipv6 pim multicast-routing** command).
- ☞ Make sure that PIM-SM6 is configured on the interface (see the **ipv6 pim sparse-mode** command).
- ☞ Make sure that SSM6 is configured in Global Mode.
- ☞ The multicast protocol uses unicast routes for RPF checks, so verify the unicast routes first.

If the problem persists, enable **debug ipv6 pim event and debug ipv6 pim packet**, save the debug information for three minutes, and send it to EnGenius technical support.

12.8.5 IPv6 DCSCM

12.8.5.1 Introduction to IPv6 DCSCM

IPv6 DCSCM (Destination Control and Source Control Multicast) covers three aspects: multicast source control, multicast user control, and service-priority-oriented multicast policy.

IPv6 DCSCM controls multicast sources as follows:

1. If source-controlled multicast is configured on the edge switches, only multicast data of the specified groups from the specified sources can pass.

2. The RP switches at the core of PIM-SM respond directly with REGISTER_STOP to REGISTER messages that are not from the specified sources and groups, and no entry is created. (This is implemented in the PIM-SM module.)

IPv6 DCSCM controls multicast users by controlling the MLD messages they send, so the control is implemented in the MLD snooping and MLD modules. The control logic includes three methods:

control by the VLAN+MAC of the message sender, control by the IP address of the message sender, and control by the ingress port of the message. MLD snooping can use all three methods at the same time, while the MLD module, at Layer 3, can control only by the sender's IP address.

IPv6 DCSCM implements service-priority-oriented multicast policy as follows: for the specified multicast data, a user-specified priority is set at the access point, so that the data is sent through trunks at a higher priority and is sent across the entire network at the user-specified priority.

12.8.5.2 IPv6 DCSCM Configuration Task Sequence

1．Source control configuration

2．Destination control configuration

3．Multicast policy configuration

1． **Source control configuration**

Source control configuration has three steps. The first step is to enable source control globally with the following command:

Command	Explanation
Global Mode	
ipv6 multicast source-control(necessary) **no ipv6 multicast source-control**	Enables source control globally. The no command disables source control globally. Note that once source control is enabled globally, all multicast packets are dropped by default. All other source control configurations can be made only after source control is enabled globally, and source control can be disabled globally only after all configured rules are disabled.

The next step is to configure source control rules. They are configured in the same way as ACLs, using ACL numbers 8000 to 8099, and each rule number can contain 10 rules. Note that the rules are ordered: the earliest configured rule comes first. Once a rule is matched, later rules do not take effect, so globally applicable rules should be configured last. The command is as follows:

Command	Explanation
Global Mode	
[no] ipv6 access-list <*8000-8099*> {deny|permit} {{<*source/M*>}|{host-source <*source-host-ip*>}|any-source} {{<*destination/M*> }|{host-destination <*destination-host-ip*>}|any-destination}	Configures source control rules. The rules take effect only when applied to a specified port. The no command deletes the specified rule.

The last step is to apply the rules to the specified port.

Note: Because the configured rules occupy hardware entries, configuring too many rules may fail when the hardware entries are full. Keep the rules as simple as possible. The command is as follows:

Command	Explanation
Port Mode	
[no] ipv6 multicast source-control access-group <*8000-8099*>	Applies a source control rule to a port. The no command cancels this configuration.

2． **Destination control configuration**

Destination control configuration is similar to source control configuration and also has three steps:

First, enable destination control globally. Because destination control prevents unauthorized users from receiving multicast data, once it is enabled globally, the switch stops flooding received multicast

data. Therefore, do not connect two or more other Layer 3 switches in the same VLAN as a switch with destination control enabled. The command is as follows:

Command	Explanation
Global Mode	
multicast destination-control(necessary)	Enables IPv4 and IPv6 destination control globally. The no command disables destination control globally. All other destination control configurations take effect only after destination control is enabled globally.

The next step is to configure destination control rules, which are similar to source control rules but use ACL numbers 9000 to 10099.

Command	Explanation
Global Mode	
[no] ipv6 access-list <*9000-10099*> {deny| permit} {{<*source/M*>}|{host-source <*source-host-ip*>}|any-source} {{<*destination/M*>}| {host-destination <*destination-host-ip*>}|any-destination}	Configures destination control rules. The rules take effect only when applied to a specified source IP address, VLAN-MAC, or port. The no command deletes the specified rule.

The last step is to apply the rules to the specified source IP address, source VLAN-MAC, or port. Note that all of these rule bindings can be used only when MLD snooping is enabled; otherwise, only source IP rules can be used, by the MLD protocol. The command is as follows:

Command	Explanation
Port Mode	
[no] ipv6 multicast destination-control access-group <*9000-10099*>	Applies a destination control rule to a port. The no command cancels the configuration.
Global Mode	
[no] ipv6 multicast destination-control <*1-4094*> <*macaddr*> access-group <*9000-10099*>	Applies destination control rules to the specified VLAN-MAC. The no command cancels the configuration.
[no] ipv6 multicast destination-control <*IPADDRESS/M*> access-group <*9000-100999*>	Applies destination control rules to the specified source IPv6 address/mask. The no command cancels the configuration.

3． **Multicast policy configuration**

Multicast policy assigns a priority to specified multicast data to meet special requirements. Note that the priority takes effect only when the multicast data is transmitted through a trunk. The configuration is simple and requires only one command, which sets the priority for the specified multicast traffic:

Command	Explanation
Global Mode	
[no] ipv6 multicast policy <*IPADDRESS/M*> <*IPADDRESS/M*> cos <*priority*>	Configures a multicast policy that sets the priority for sources and groups in a specified range. The valid priority range is 0 to 7.

12.8.5.3 IPv6 DCSCM Typical Examples

1. Source control

To prevent an edge switch from sending multicast data arbitrarily, configure the edge switch so that only port Ethernet1/0/4 can send multicast data, and only for group ff1e::1. The uplink port Ethernet1/0/25 can forward multicast data without restriction. The configuration is as follows:

```
Switch(config)#ipv6 access-list 8000 permit any-source ff1e::1
Switch(config)#ipv6 access-list 8001 permit any any
Switch(config)#ipv6 multicast source-control
Switch(config)#interface Ethernet1/0/4
Switch(Config-If-Ethernet1/0/4)#ipv6 multicast source-control access-group
8000
Switch(config)#interface Ethernet1/0/25
Switch(Config-If-Ethernet1/0/25)#ipv6 multicast source-control access-group
8001
```

2. Destination control

To prevent users on segment fe80::203:fff:fe01:228a/64 from joining group ff1e::1/64, configure the switch as follows:

First, enable MLD snooping in the VLAN where the users are located (VLAN2 in this example).

```
Switch(config)#ipv6 mld snooping
Switch(config)#ipv6 mld snooping vlan 2
```

Then configure the destination control access list and apply it to the specified IPv6 address.

```
Switch(config)#ipv6 access-list 9000 deny any ff1e::1/64
Switch(config)#ipv6 access-list 9000 permit any any
Switch(config)#multicast destination-control
Switch(config)#ipv6 multicast destination-control fe80::203:fff:fe01:228a/64
access-group 9000
```

Thus, users on this segment can join only groups other than 2ff1e::1/64.

3. Multicast policy

Server 2008::1 sends important multicast data to group ff1e::1. Configure its access switch as follows:

```
Switch(config)#ipv6 multicast policy 2008::1/128 ff1e::1/128 cos 4
```

This multicast flow then has priority 4 when it passes through the trunk port of this switch to another switch. (Priority 4 is generally relatively high. Data with even higher priority may be protocol data; if a higher priority is set for multicast traffic, the switch protocols may operate abnormally when there is too much multicast data.)

12.8.5.4 IPv6 DCSCM Troubleshooting

The IPv6 DCSCM module works like an ACL, so most problems are caused by improper configuration. Read the instructions above carefully.

12.8.6 MLD

12.8.6.1 Introduction to MLD

MLD (Multicast Listener Discovery) is the protocol used to discover multicast group members (receivers) for IPv6 multicast. It is similar to IGMP in IPv4 multicast: MLD version 1 corresponds to IGMP version 2, and MLD version 2 corresponds to IGMP version 3. The current firmware supports MLDv1 and MLDv2.

IPv6 multicast hosts can join or leave a multicast group at any location and at any time, regardless of the total number of group members. It is neither necessary nor possible for a multicast switch to store the relationships among all host members. The multicast switch only uses MLD to find out whether there are receivers of a given multicast group on the network segment connected to each port. Each host only needs to keep track of the multicast groups it has joined.

MLD is asymmetric between the host and the switch: the host responds to MLD Query messages from the multicast switch with membership Report messages. The switch periodically sends membership Query messages and determines from the responses whether any host in its subnets

has joined a specific group. After it receives a report that a host has left a group, it sends a query for that group to confirm whether any members remain.

MLD has three types of protocol messages: Query, Report, and Done (which corresponds to Leave in IGMPv2). As in IGMPv2, Query messages include General Query and Group-Specific Query. A General Query uses the all-hosts multicast address FF02::1 as the destination address and a group address of 0, while a Group-Specific Query uses the group address as the destination address. MLD messages use types 130, 131, and 132 for the three message types above. The other logic is basically the same as in IGMPv2.

MLD version 2 uses FF02::16 as the destination address of membership reports and 143 as the message type. The other logic of MLD version 2 is similar to that of IGMP version 3.

12.8.6.2 MLD Configuration Task List

1、 Enable MLD (Required)

2、 Configure MLD auxiliary parameters (Required)

(1) Configure MLD group parameters

1) Configure MLD group filter conditions

(2) Configure MLD query parameters

1) Configure the interval for sending MLD query messages

2) Configure the maximum response time of MLD queries

3) Configure the MLD query timeout

3、 Disable MLD

1. **Enable MLD**

There is no dedicated command for enabling MLD on EnGenius ECS8830F/ECS8854F Layer 3 switches. MLD starts automatically as soon as any IPv6 multicast protocol is enabled on the corresponding interface.

Command	Explanation
Global Mode	
Ipv6 pim multicast-routing	Enables the global IPv6 multicast protocol, which is a prerequisite for enabling MLD. The no command disables the IPv6 multicast protocol and MLD. (Required)
Command	**Explanation**
Interface Mode	
ipv6 pim dense-mode | ipv6 pim sparse-mode	Enables MLD. The no command disables MLD. (Required)

2. **Configure MLD auxiliary parameters**

(1) Configure MLD group parameters

1） Configure MLD group filter conditions

Command	**Explanation**
Interface Mode	
ipv6 mld access-group *<acl_name>* **no ipv6 mld access-group**	Configures the MLD group filter conditions on the interface. The no command cancels the filter conditions.

(2) Configure MLD query parameters

1) Configure the interval for sending MLD query messages

2) Configure the maximum response time of MLD queries

3) Configure the MLD query timeout

Command	Explanation
Interface Mode	
ipv6 mld query-interval <*time_val*> **no ipv6 mld query-interval**	Configures the interval for periodically sending MLD query messages. The no command restores the default value.
ipv6 mld query-max-response-time <*time_val*> **no ipv6 mld query-max-response-time**	Configures the maximum response time of MLD queries on the interface. The no command restores the default value.
ipv6 mld query-timeout <*time_val*> **no ipv6 mld query-timeout**	Configures the MLD query timeout on the interface. The no command restores the default value.

3. **Disable MLD**

Command	Explanation
Interface Mode	
no ipv6 pim dense-mode | no ipv6 pim sparse-mode | no ipv6 pim multicast-routing (Global Mode)	Disables MLD.

12.8.6.3 MLD Typical Application

As shown in the following figure, add the ports of Switch A and Switch B to the corresponding VLANs, and enable PIM6 on each VLAN interface.

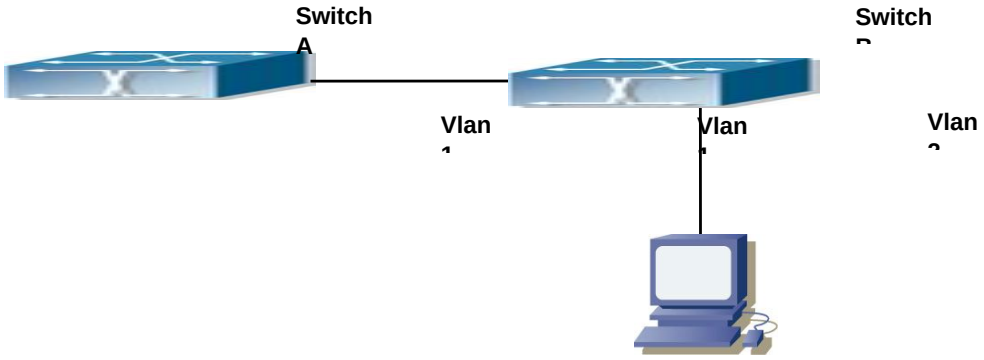


Figure 12-45 Network topology

The configuration procedure for SwitchA and SwitchB is as follows:

(1) Configure SwitchA:

```
Switch (config) #ipv6 pim multicast-routing
Switch (config) #ipv6 pim rp-address 3FFE::1
Switch (config) #interface vlan 1
Switch (Config-if-Vlan1) #ipv6 address 3FFE::1/64
Switch (Config-if-Vlan1) #ipv6 pim sparse-mode
```

(2) Configure SwitchB:

```
Switch (config) #ipv6 pim multicast-routing
Switch (config) #ipv6 pim rp-address 3FFE::1
Switch (config) #interface vlan1
Switch (Config-if-Vlan1) #ipv6 address 3FFE::2/64
Switch (Config-if-Vlan1) #ipv6 pim sparse-mode
Switch (Config-if-Vlan1) #exit
Switch (config) #interface vlan2
Switch (Config-if-Vlan2) #ipv6 address 3FFA::1/64
Switch (Config-if-Vlan2) #ipv6 pim sparse-mode
```

```
Switch (Config-if-Vlan2) #ipv6 mld query-timeout 150
```

12.8.6.4 MLD Troubleshooting

MLD may not work properly because of problems such as faulty physical connections or incorrect configuration. Pay attention to the following:

- ☞ Ensure that the physical connections are correct.
- ☞ Ensure that the interface and link protocol are up (use the **show interface** command).
- ☞ Ensure that a multicast protocol is enabled on the interface.
- ☞ Ensure that the timer settings of all routers on the same network segment are consistent; the default settings are recommended.
- ☞ The multicast protocol uses unicast routes for RPF checks, so first ensure that the unicast routes are correct.

If the MLD problem persists, use debug commands such as **debug ipv6 MLD event/packet**, record the debug information for three minutes, and send it to EnGenius technical support.

12.8.7 MLD Snooping

12.8.7.1 Introduction to MLD Snooping

MLD (Multicast Listener Discovery) implements multicast in IPv6. Multicast-capable network devices such as routers use MLD to discover multicast listeners, and listeners that want to join a multicast group use MLD to inform the router that they want to receive data from a given multicast address. All of this is done through MLD message exchange. First, the router sends an MLD Multicast Listener Query message to the multicast address that reaches all listeners (ff02::1). A listener that wants to join the multicast address then sends an MLD Multicast Listener Report back through that multicast address.

MLD Snooping means listening to MLD. With MLD Snooping, the switch prevents multicast traffic from flooding and forwards multicast traffic only to ports associated with multicast devices. The switch listens to the MLD messages between multicast routers and listeners, maintains the multicast group forwarding list based on what it learns, and forwards multicast packets according to that list.

The switch implements MLD Snooping and supports MLDv2, so users can receive IPv6 multicast through the switch.

12.8.7.2 MLD Snooping Configuration Tasks

1. Enable the MLD Snooping function

2. Configure MLD Snooping

1. Enable the MLD Snooping function

Command	Explanation
Global Mode	
ipv6 mld snooping **no ipv6 mld snooping**	Enables MLD Snooping globally. The **no ipv6 mld snooping** command disables MLD Snooping globally.

2. Configure MLD Snooping

Command	Explanation
Global Mode	
ipv6 mld snooping vlan <*vlan-id*> **no ipv6 mld snooping vlan <*vlan-id*>**	Enables MLD Snooping on the specified VLAN. The no command disables MLD Snooping on the specified VLAN.
ipv6 mld snooping vlan <*vlan-id*>	Configures the maximum number of groups that

limit {group <*g_limit*> | source <*s_limit*>} **no ipv6 mld snooping vlan <*vlan-id*> limit**	MLD Snooping can join and the maximum number of sources in each group. The no command restores the default.
ipv6 mld snooping vlan <*vlan-id*> l2-general-querier **no ipv6 mld snooping vlan <*vlan-id*> l2-general-querier**	Configures the VLAN as a Layer 2 general querier; this is recommended on each segment. The no command cancels the Layer 2 general querier configuration.
ipv6 mld snooping vlan <*vlan-id*> mrouter-port interface <*interface –name*> **no ipv6 mld snooping vlan <*vlan-id*> mrouter-port interface <*interface –name*>**	Configures a static mrouter port in the specified VLAN. The no command cancels the mrouter port configuration.
ipv6 mld snooping vlan <*vlan-id*> mrouter-port learnpim6 **no ipv6 mld snooping vlan <*vlan-id*> mrouter-port learnpim6**	Enables the specified VLAN to learn mrouter ports from PIMv6 packets. The no command disables the function.
ipv6 mld snooping vlan <*vlan-id*> mrpt <*value*> **no ipv6 mld snooping vlan <*vlan-id*> mrpt**	Configures the keepalive time of the mrouter port. The no command restores the default.
ipv6 mld snooping vlan <*vlan-id*> query-interval <*value*> **no ipv6 mld snooping vlan <*vlan-id*> query-interval**	Configures the query interval. The no command restores the default.
ipv6 mld snooping vlan <*vlan-id*> immediate-leave **no ipv6 mld snooping vlan <*vlan-id*> immediate-leave**	Enables immediate leave from multicast groups for MLD Snooping on the specified VLAN. The no command cancels the immediate-leave configuration.
ipv6 mld snooping vlan <*vlan-id*> query-mrsp <*value*> **no ipv6 mld snooping vlan <*vlan-id*> query-mrsp**	Configures the maximum query response time. The no command restores the default.
ipv6 mld snooping vlan <*vlan-id*> query-robustness <*value*> **no ipv6 mld snooping vlan <*vlan-id*> query-robustness**	Configures the query robustness. The no command restores the default.
ipv6 mld snooping vlan <*vlan-id*> suppression-query-time <*value*> **no ipv6 mld snooping vlan <*vlan-id*> suppression-query-time**	Configures the suppression query time. The no command restores the default.
Ipv6 mld snooping vlan <*vlan-id*> static-group <*X:X::X:X*> [source <*X:X::X:X*>] interface [ethernet | port-channel] <*IFNAME*> **no ipv6 mld snooping vlan <*vlan-id*> static-group <*X:X::X:X*> [source <*X:X::X:X*>] interface [ethernet | port-channel] <*IFNAME*>**	Configures a static group on the specified port of the VLAN. The no command cancels this configuration.

12.8.7.3 MLD Snooping Examples

Scenario 1: MLD Snooping Function

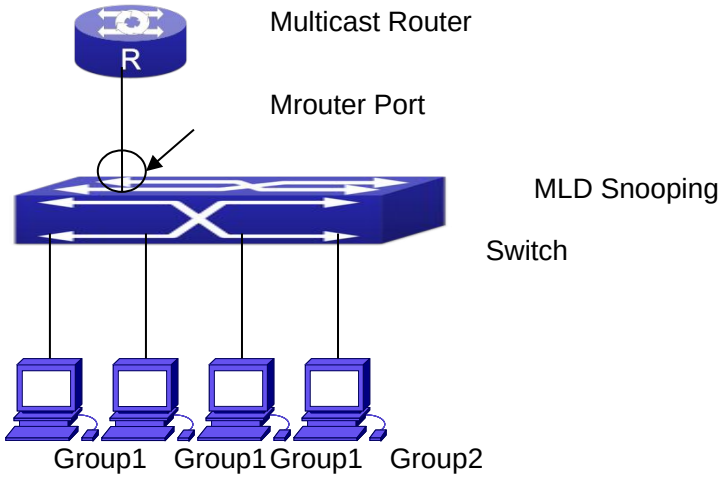


Figure 12-16 Enabling MLD Snooping on the switch

As shown above, VLAN 100 on the switch consists of ports 1, 2, 6, 10, and 12. Four hosts are connected to ports 2, 6, 10, and 12, and the multicast router is connected to port 1. Suppose MLD Snooping is needed on VLAN 100. By default, both global MLD Snooping and MLD Snooping on each VLAN are disabled, so first enable global MLD Snooping and MLD Snooping on VLAN 100, and then set port 1 of VLAN 100 as an mrouter port.

The configuration procedure is as follows:

```
Switch#config
Switch(config)#ipv6 mld snooping
Switch(config)#ipv6 mld snooping vlan 100
Switch(config)#ipv6 mld snooping vlan 100 mrouter-port interface ethernet
1/0/1
```

Multicast configuration:

Assume there are two multicast servers, Multicast Server 1 and Multicast Server 2. Programs 1 and 2 are provided by Multicast Server 1, and program 3 by Multicast Server 2, using group addresses Group 1, Group 2, and Group 3, respectively. A multicast application is running on all four hosts: the two hosts connected to ports 2 and 6 are playing program 1, the host connected to port 10 is playing program 2, and the host connected to port 12 is playing program 3.

MLD Snooping results:

The multicast table on VLAN 100 shows that ports 1, 2, and 6 are in (Multicasting Server 1, Group1), ports 1 and 10 are in (Multicasting Server 1, Group2), and ports 1, 121, and 12 are in (Multicasting Server 2, Group3).

All four hosts receive the programs they are interested in. Ports 2 and 6 receive no traffic from programs 2 and 3, port 10 receives no traffic from programs 1 and 3, and port 12 receives no traffic from programs 1 and 2.

Scenario 2: MLD L2 general querier

Figure 12-57 Switch as the MLD querier

The configuration of switch B is the same as that of the switch in scenario 1, and switch 1 replaces the multicast router of scenario 1. Assume that VLAN 60 on switch 1 contains ports 1, 2, 10, and 12, with port 1 connected to the multicast server and port 2 connected to switch 2. To send queries periodically, enable global MLD Snooping and run **mld snooping vlan 60 l2-general-querier** to make VLAN 60 a Layer 2 general querier.

The configuration procedure is as follows:

```
SwitchA#config
SwitchA(config)#ipv6 mld snooping
```

```
SwitchA(config)#ipv6 mld snooping vlan 60
SwitchA(config)#ipv6 mld snooping vlan 60 l2-general-querier
SwitchB#config
SwitchB(config)#ipv6 mld snooping
SwitchB(config)#ipv6 mld snooping vlan 100
SwitchB(config)#ipv6 mld snooping vlan 100 mrouter interface ethernet 1/0/1
```

Multicast configuration:

Same as scenario 1

MLD Snooping results:

Same as scenario 1

Scenario 3: Working with Layer 3 multicast protocols

The switch used in scenario 1 is replaced with a router, and the other configurations remain the same. The multicast and IGMP snooping configurations are the same as in scenario 1. Configure PIM-SM6 on the router and enable PIM-SM6 on VLAN 100 (use the same PIM mode as the connected multicast router).

The configuration is as follows:

```
switch#config
switch(config)#ipv6 pim multicast-routing
switch(config)#interface vlan 100
switch(config-if-vlan100)#ipv6 pim sparse-mode
```

When a Layer 3 multicast protocol is enabled, MLD snooping does not install entries. It performs only the following tasks:

- Removes the Layer 2 multicast entries.
- Provides query functions to Layer 3, with VLAN, S, and G as parameters.
- When Layer 3 MLD is disabled, resumes installing Layer 2 multicast entries.

Looking up the Layer 3 IP6MC entries shows that the ports can be indicated by the Layer 3 multicast entries. This ensures that MLD Snooping works with the Layer 3 multicast protocols.

12.8.7.4 MLD Snooping Troubleshooting

MLD Snooping may not work properly because of physical connection failures, incorrect configuration, and so on. Ensure the following:

- ☞ Ensure that the physical connections are correct.
- ☞ Ensure that MLD Snooping is enabled in Global Mode (use **ipv6 mld snooping**).
- ☞ Ensure that MLD Snooping is configured on the VLAN in Global Mode (use **ipv6 mld snooping vlan <vlan-id>**).
- ☞ Ensure that a VLAN is configured as an L2 general querier or that a static mrouter port is configured in the segment.
- ☞ Use the relevant show command to check whether the MLD snooping information is correct.

12.9 IPv6 Security RA

12.9.1 Introduction to IPv6 Security RA

In IPv6 networks, the topology generally consists of routers, Layer 2 switches, and IPv6 hosts. Routers usually advertise RAs containing the link prefix, link MTU, and other information. When IPv6 hosts receive an RA, they create a link address and set the router that sent the RA as their default router to enable IPv6 communication. If a malicious IPv6 host sends RAs that cause normal IPv6 users to set the malicious host as their default router, the malicious user can capture other users'

information, which threatens network security. At the same time, the normal users obtain incorrect addresses and cannot connect to the network. To implement security RA, configure switch ports to reject malicious RA messages, which prevents malicious RAs from being forwarded to a certain extent and keeps the network operating normally.

12.9.2 IPv6 Security RA Configuration Task Sequence

1. Globally enable IPv6 security RA
2. Enable IPv6 security RA on a port
3. Display and debug IPv6 security RA information

1. Globally enable IPv6 security RA

Command	Explanation
Global Mode	
ipv6 security-ra enable **no ipv6 security-ra enable**	Enables or disables IPv6 security RA globally.

2. Enable IPv6 security RA on a port

Command	Explanation
Port Mode	
ipv6 security-ra enable **no ipv6 security-ra enable**	Enables or disables IPv6 security RA on the port.

3. Display and debug IPv6 security RA information

Command	Explanation
Admin Mode	
debug ipv6 security-ra **no debug ipv6 security-ra**	Enables debug output for the IPv6 security RA module. The no command disables the debug output.
show ipv6 security-ra [interface <*interface-list*>]	Displays the untrusted ports and whether security RA is enabled globally.

12.9.3 IPv6 Security RA Typical Examples

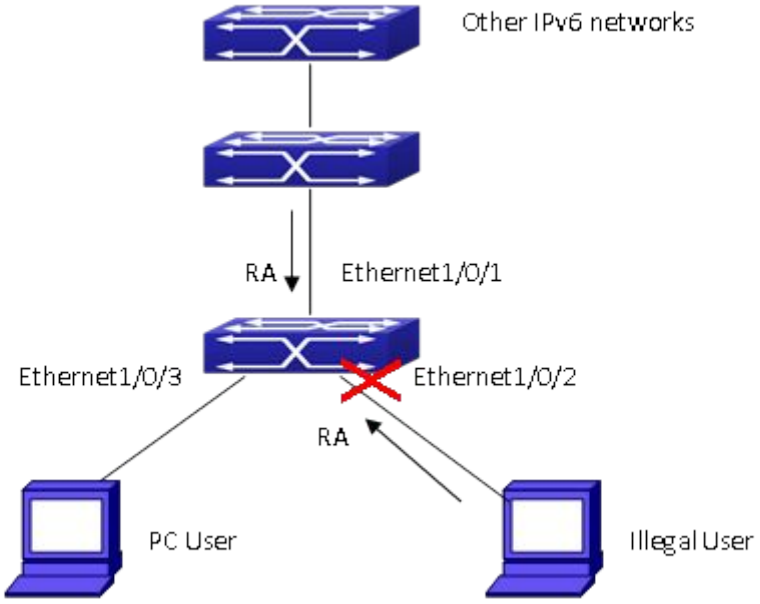


Figure 12-18 IPv6 security RA diagram

Note: If the illegal user in the figure advertises RAs, the normal user receives them, sets the malicious IPv6 host as its default router, and changes its own address. As a result, the normal user cannot connect to the network. To prevent this, configure security RA on port 1/0/2 of the switch so that RAs from the illegal user do not affect the normal user.

Switch configuration:

```
Switch#config
Switch(config)#ipv6 security-ra enable
Switch(Config-If-Ethernet1/0/2)# ipv6 security-ra enable
```

12.9.4 IPv6 Security RA Troubleshooting

IPv6 security RA is simple. If the function does not work as expected after it is configured:

- ☞ Check whether the switch is configured correctly.
- ☞ Check whether the switch has rules that conflict with the security RA function; such rules cause RA messages to be forwarded.

12.10 SAVI Configuration

12.10.1 Introduction to SAVI

SAVI (Source Address Validation Improvement) is a security authentication method that validates source addresses at node-level granularity. Using the CPS (Control Packet Snooping) mechanism, it monitors the exchange of related protocol packets (such as ND and DHCPv6) to obtain trusted node information (such as port and MAC address), called anchor information. It then binds the anchor information to the node source address and installs corresponding filter rules that allow only matching packets to pass, thereby validating node source addresses.

According to protocol packet type, SAVI includes ND Snooping, DHCPv6 Snooping, and RA Snooping. ND Snooping inspects ND protocol packets and creates bindings for IPv6 addresses that nodes obtain through stateless address autoconfiguration. DHCPv6 Snooping inspects DHCPv6 protocol packets and creates bindings for IPv6 addresses that nodes obtain through stateful address configuration. RA Snooping prevents unauthorized nodes from sending spoofed RA packets.

12.10.2 SAVI Configuration

SAVI configuration task list:

1. Enable or disable SAVI
2. Enable or disable SAVI application scenarios
3. Configure SAVI bindings
4. Configure the global max-dad-delay for SAVI
5. Configure the global max-dad-prepare-delay for SAVI
6. Configure the global max-slaac-life for SAVI
7. Configure the SAVI bind-protect lifetime
8. Enable or disable SAVI prefix check
9. Configure the IPv6 address prefix for a link
10. Configure the number of IPv6 address filter entries
11. Configure the check mode for SAVI binding conflicts
12. Enable or disable user authentication
13. Enable or disable DHCPv6 trust on a port

14. Enable or disable ND trust on a port
15. Configure the binding number

1. **Enable or disable SAVI**

Command	Explanation
Global Mode	
savi enable **no savi enable**	Enables SAVI globally. The no command disables the function.

2. **Enable or disable SAVI application scenarios**

Command	Explanation
Global Mode	
savi ipv6 {dhcp-only | slaac-only | dhcp-slaac} enable **no savi ipv6 {dhcp-only | slaac-only | dhcp-slaac} enable**	Enables the specified SAVI application scenario. The no command disables it.

3. **Configure SAVI bindings**

Command	Explanation
Global Mode	
savi ipv6 check source binding ip <*ip-address*> mac <mac-address> interface <*if-name*> {type [slaac | dhcp] lifetime <*lifetime*> | type static} **no savi ipv6 check source binding ip <i*p-address*> interface <*if-name*>**	Configures a static or dynamic binding manually. The no command deletes the configured binding. This command can be configured when the global savi enable, slaac-only enable, dhcp-only enable, or dhcp-slaac enable function is configured.

4. **Configure the global max-dad-delay for SAVI**

Command	Explanation
Global Mode	
savi max-dad-delay <*max-dad-delay*> **no savi max-dad-delay**	Configures the maximum lifetime of a SAVI binding in the DETECTION state. The no command restores the default value.

5. **Configure the global max-dad-prepare-delay for SAVI**

Command	Explanation
Global Mode	
savi max-dad-prepare-delay <*max-dad-prepare-delay*> **no savi max-dad-prepare-delay**	Configures the maximum redetection lifetime of a SAVI binding. The no command restores the default value.

6. **Configure the global max-slaac-life for SAVI**

Command	Explanation
Global Mode	
savi max-slaac-life <*max-slaac-life*> **no savi max-slaac-life**	Configures the lifetime of a dynamic SLAAC binding in the BOUND state. The no command restores the default value.

7. **Configure the SAVI bind-protect lifetime**

Command	Explanation
Global Mode	
savi timeout bind-protect <*protect-time*>	Configures how long bindings on a port are protected after the port goes from up

no savi timeout bind-protect	to down. The no command restores the default value.

8. **Enable or disable SAVI prefix check**

Command	Explanation
Global Mode	
ipv6 cps prefix check enable **no ipv6 cps prefix check enable**	Enables the SAVI address prefix check. The no command disables the function.

9. **Configure the IPv6 address prefix for a link**

Command	Explanation
Global Mode	
ipv6 cps prefix <*ip-address*> vlan <*vid*> **no ipv6 cps prefix <*ip-address*>**	Configures the IPv6 address prefix for a link manually. The no command deletes the configured address prefix.

10. **Configure the number of IPv6 address filter entries**

Command	Explanation
Global Mode	
savi ipv6 mac-binding-limit <limit-num> **no savi ipv6 mac-binding-limit**	Configures the maximum number of dynamic bindings for the same MAC address. The no command restores the default value. Note: This limit applies only to dynamic bindings, not to static bindings.

11. **Configure the check mode for SAVI binding conflicts**

Command	Explanation
Global Mode	
savi check binding <*simple* | *probe*> mode **no savi check binding mode**	Configures the check mode for binding conflicts. The no command deletes the check mode.

12. **Enable or disable user authentication**

Command	Explanation
Port Mode	
savi ipv6 check source [ip-address mac-address | ip-address | mac-address] **no savi ipv6 check source**	Enables user source authentication on the port. The no command disables the function.

13. **Enable or disable DHCPv6 trust on a port**

Command	Explanation
Port Mode	
ipv6 dhcp snooping trust **no ipv6 dhcp snooping trust**	Configures the port as a DHCPv6 trusted port. The no command disables the trust function (the port changes from trusted to untrusted).

14. **Enable or disable ND trust on a port**

Command	Explanation
Port Mode	
ipv6 nd snooping trust **no ipv6 nd snooping trust**	Configures the port as a SLAAC-trusted and RA-trusted port. The no command removes the port's trust function.

15. **Configure the binding number**

Command	Explanation
Port Mode	
savi ipv6 binding num <*limit-num*> **no savi ipv6 binding num**	Configures the maximum number of bindings on the port. The no command restores the default value. Note: This limit applies only to dynamic bindings, not to static bindings.

12.10.3 SAVI Typical Application

In practice, SAVI is usually deployed on access-layer switches to validate the source addresses of directly connected nodes. SAVI has four typical application scenarios: DHCP-Only, Slaac-Only, DHCP-Slaac, and static binding. Select the scenario that matches the actual requirements. In a dual-stack network, using SAVI together with IPv4 DHCP snooping provides both IPv4 and IPv6 source address authentication.

Typical network topology for SAVI:

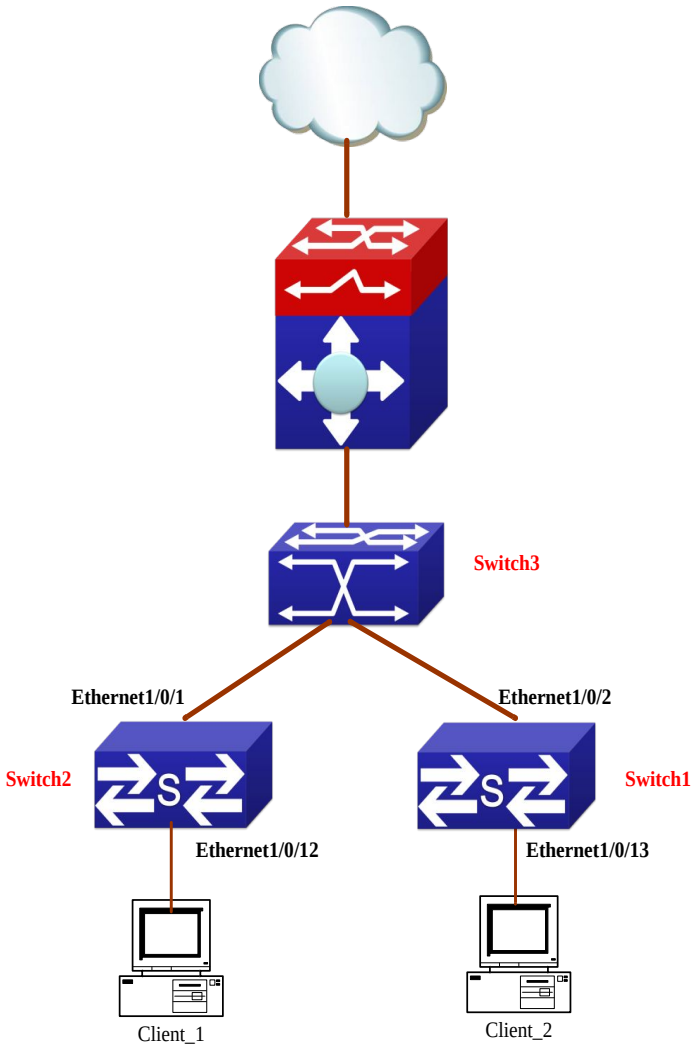


Figure 12-319

Client_1 and Client_2 are two users' PCs with IPv6 installed. They connect to port Ethernet1/0/12 of Switch1 and port Ethernet1/0/13 of Switch2, respectively, where the SAVI source address check is enabled. Ethernet1/0/1 and Ethernet1/0/2 are the uplink ports of Switch1 and Switch2, respectively,

with DHCP trust and ND trust enabled. Aggregation switch Switch3 runs the DHCPv6 server and router advertisement functions.

Configuration steps for the SAVI DHCP-SLAAC scenario:

```
Switch1>enable
Switch1#config
Switch1(config)#savi enable
Switch1(config)#savi ipv6 dhcp-slaac enable
Switch1(config)#savi check binding probe mode
Switch1(config)#interface ethernet1/0/1
Switch1(config-if-ethernet1/0/1)#ipv6 dhcp snooping trust
Switch1(config-if-ethernet1/0/1)#ipv6 nd snooping trust
Switch1(config-if-ethernet1/0/1)#exit
Switch1(config)#interface ethernet1/0/12-20
Switch1(config-if-port-range)#savi ipv6 check source ip-address mac-address
Switch1(config-if-port-range)#savi ipv6 binding num 4
Switch1(config-if-port-range)#exit
Switch1(config)#exit
Switch1#write
```

12.10.4 SAVI Troubleshooting

After ensuring that the client hardware and cables are working, check the following possible problems and suggested solutions:

- ☞ If IPv6 packets are filtered incorrectly after SAVI is enabled, ensure that SAVI is enabled globally. Then enable the global function of the SAVI scenario that matches the actual application, and enable port authentication.
- ☞ If clients cannot obtain IPv6 addresses from the DHCPv6 server after SAVI is enabled, ensure that DHCP trust is configured on the uplink port toward the DHCPv6 server.
- ☞ If node bindings cannot be created for new users after SAVI is enabled, check whether a maximum binding number is configured on the directly connected port and whether that number has been reached. If the binding number has reached the limit, configure a higher binding limit.
- ☞ If node bindings still cannot be created for new users after a higher binding limit is configured, check whether a binding number is configured for the same MAC address on the directly connected port and whether that number has been reached. If it has, configure a higher limit.

12.11 IPv6 VRRPv3

12.11.1 Introduction to VRRPv3

VRRPv3 is a virtual router redundancy protocol for IPv6, designed based on VRRP (VRRPv2) for IPv4. A brief introduction follows.

In a TCP/IP network, routers must be specified to ensure communication between devices that are not physically connected. Currently, the two most common methods of specifying routers are learning them dynamically through routing protocols (such as the interior routing protocols RIP and OSPF) and configuring them statically. Running a dynamic routing protocol on every terminal is unrealistic, because most client operating systems do not support dynamic routing protocols; even if they do, they are limited by management overhead, convergence, security, and many other issues. Therefore, terminal IP devices commonly use static routing, which usually means specifying one or more default gateways for them. Static routing simplifies network management and reduces the communication overhead of terminal devices, but it has a disadvantage: if the router acting as the default gateway fails, communication fails for all hosts that use this gateway as their next hop. Even if multiple default gateways are configured, the terminal devices cannot switch to a new gateway until they are

rebooted. A virtual router redundancy protocol (VRRP) effectively avoids the flaws of statically specified gateways.

VRRP involves two pairs of important concepts: VRRP routers and virtual routers, and master routers and backup routers. VRRP routers are routers running VRRP; they are physical entities. Virtual routers are created by VRRP; they are logical concepts. A group of VRRP routers cooperates to form a virtual router, which appears externally as a logical router with a unique fixed IP address and MAC address. The routers in the same VRRP group play one of two mutually exclusive roles: master router or backup router. A VRRP group can have only one master router but one or more backup routers. VRRPv3 uses an election policy to select a master router from the router group; the master responds to ND (Neighbor Discovery) neighbor solicitation messages (ARP in IPv4) and forwards IP data packets, while the other routers in the group wait as backups. When the master router fails for any reason, a backup router becomes the master router after a delay of a few seconds. Because this switchover is very fast and requires no change of IP address or MAC address, it is transparent to terminal user systems.

In an IPv6 environment, hosts on a LAN usually learn the default gateway through the Neighbor Discovery Protocol (NDP), based on router advertisement messages received periodically from routers. IPv6 NDP includes a mechanism called Neighbor Unreachability Detection, which checks whether a neighbor node has failed by sending unicast neighbor solicitation messages to it. To reduce the overhead of sending neighbor solicitation messages, they are sent only to neighbor nodes that are carrying traffic, and only if there has been no indication for a period of time that the router is up. With the default parameters, Neighbor Unreachability Detection takes about 38 seconds to detect an unreachable router, which is a significant delay for users and may cause timeouts in some transport protocols. Compared with NDP, VRRP provides fast default gateway switchover. In VRRP, a backup router can take over from an unavailable master router in about 3 seconds (with default parameters), and this process requires no interaction with hosts, so it is transparent to them.

12.11.1.1 VRRPv3 Message Format

VRRPv3 has its own message format. VRRP messages communicate the router priority and the Master state within the backup group. They are encapsulated in IPv6 packets and sent to a specified IPv6 multicast address. The VRRPv3 message format is shown in the figure below. The source address of the IPv6 packet carrying a VRRPv3 message is the local address of the outbound interface, and the destination address is an IPv6 multicast address (the multicast address allocated to VRRPv3 is FF02:0:0:0:0:0:0:12). The hop limit must be 255, and the next header value is 112 (indicating a VRRP message).

The fields of a VRRPv3 message are described below:

Version: The VRRP version; the value is 3.

Type: The VRRP message type. There is only one type, ADVERTISEMENT, with the value 1.

Virtual Rtr ID: The ID of the virtual router.

Priority: The priority, ranging from 0 to 255.

Count IPv6 Addr: The number of IPv6 addresses in the VRRPv3 message; the minimum is 1.

Rsvd: Reserved field; the value is 0.

Adver Int: The advertisement interval of VRRPv3 messages, in seconds.

Checksum: The checksum, calculated over the entire VRRPv3 message and an IPv6 pseudo-header (see RFC 2460 for details).

IPv6 Address(es): One or more IPv6 addresses associated with the virtual router. Their number equals "Count IPv6 Addr", and the first one must be the virtual IPv6 address of the virtual router.

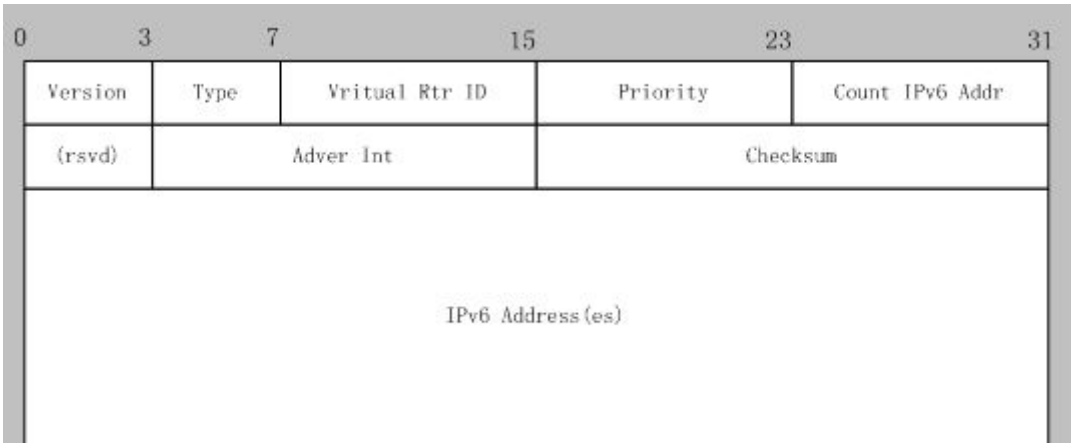


Figure 12-20 VRRPv3 message format

12.11.1.2 VRRPv3 Working Mechanism

VRRPv3 works in the same way as VRRPv2, mainly through the exchange of VRRP advertisement messages. It is briefly described as follows:

Each virtual router has a unique ID, the VRID, ranging from 1 to 255. Externally, the virtual router has a unique virtual MAC address in the format 00-00-5E-00-02-{VRID} (in VRRPv2, the format is 00-00-5E-00-01-{VRID}). The master router uses this MAC address to respond to ND neighbor solicitations (ARP requests in VRRPv2). Thus, regardless of switchovers, terminal devices always see the same IP and MAC addresses, which minimizes the impact of switchovers on them.

There is only one kind of VRRP control message: the VRRP advertisement. It is encapsulated in IP multicast packets, with multicast addresses in the format FF02:0:0:0:0:0:XXXX:XXXX. For consistency with the VRRPv2 multicast address (224.0.0.18), VRRPv3 advertisements can use the multicast address FF02:0:0:0:0:0:0:12, and advertisements are limited to the same LAN. This ensures that the same VRIDs can be reused in different networks. To reduce bandwidth overhead, only master routers send VRRP advertisements periodically. A backup router starts a new VRRP election if it receives no VRRP advertisement for three consecutive advertisement intervals or if it receives an advertisement with priority 0.

In a VRRP router group, the master router is elected by priority. The VRRP priority range is 0-255. If the IP address of a VRRP router is the same as the virtual router address, that router is called the IP address owner in the VRRP group, and it automatically has the highest priority, 255. Priority 0 is usually used when the IP address owner gives up the master role. The configurable priority range is 1-254. Priorities can be set according to link speed and cost, router performance and reliability, and other management policies. In the master election, the router with the higher priority wins, so if there is an IP address owner in the VRRP group, it is always the master router. Among candidate routers with the same priority, the one with the larger IP address wins. VRRP also provides a preemption policy: if it is configured, a backup router with a higher priority preempts the master role from a current master router with a lower priority.

To avoid returning a physical MAC address when the virtual IP is pinged, the virtual IP cannot be the real IP address of an interface. Therefore, all interfaces participating in the backup group election are backups by default.

12.11.2 VRRPv3 Configuration

12.11.2.1 Configuration Task List

1. Create or delete the virtual router (required)
2. Configure the virtual IPv6 address and interface of VRRPv3 (required)

3. Enable or disable the virtual router (required)
4. Configure VRRPv3 auxiliary parameters (optional)

(1) Configure VRRPv3 preempt mode

(2) Configure VRRPv3 priority

(3) Configure the VRRPv3 advertisement interval

(4) Configure the VRRPv3 monitored interface

1. Create or delete the virtual router

Command	Explanation
Global Mode	
router ipv6 vrrp <*vrid*> **no router ipv6 vrrp <*vrid*>**	Creates or deletes the virtual router.

2. Configure the virtual IPv6 address and interface of VRRPv3

Command	Explanation
VRRPv3 Router Mode	
virtual-ipv6 <*ipv6-address*> Interface {Vlan <*ID*> | *IFNAME* } **no virtual-ipv6 interface**	Configures the virtual IPv6 address and interface of VRRPv3. The no command deletes the virtual IPv6 address and interface.

3. Enable or disable the virtual router

Command	Explanation
VRRPv3 Router Mode	
enable	Enables the virtual router.
disable	Disables the virtual router.

4. Configure VRRPv3 auxiliary parameters

(1) Configure VRRPv3 preempt mode

Command	Explanation
VRRPv3 Router Mode	
preempt-mode {true| false}	Configures the VRRPv3 preempt mode.

(2) Configure VRRPv3 priority

Command	Explanation
VRRPv3 Router Mode	
priority < *priority* >	Configures the VRRPv3 priority.

(3) Configure the VRRPv3 advertisement interval

Command	Explanation
VRRPv3 Router Mode	
advertisement-interval <time>	Configures the VRRPv3 advertisement interval (in centiseconds).

(4) Configure the VRRPv3 monitored interface

Command	Explanation
VRRPv3 Router Mode	
circuit-failover {vlan <*ID*>| *IFNAME*} <*value_reduced*> **no circuit-failover**	Configures the VRRPv3 monitored interface. The no command deletes the monitored interface.

12.11.3 VRRPv3 Typical Examples

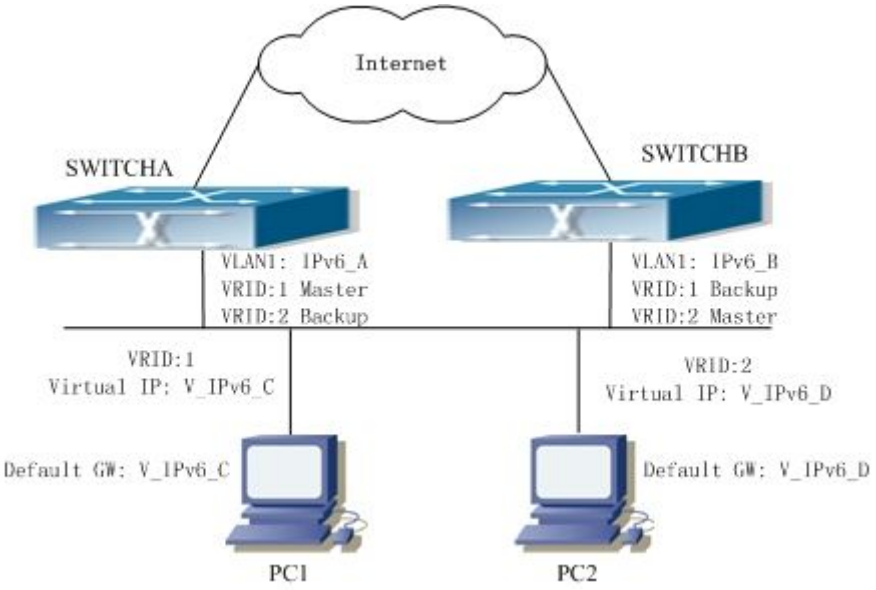


Figure 12-21 Typical VRRPv3 network topology

As shown in the figure, switch A and switch B back each other up: switch A is the master of backup group 1 and a backup in backup group 2, and switch B is the master of backup group 2 and a backup in backup group 1. The IPv6 addresses of switch A and switch B are "IPv6_A" and "IPv6_ B", respectively (IPv6_A and IPv6_B should be in the same segment). The virtual IPv6 addresses of backup group 1 and backup group 2 are "V_IPv6_C" and "V_IPV6_D", respectively, and the hosts' default IPv6 gateway addresses are configured as "V_IPv6_C" and "V_IPv6_D", respectively. (In practice, hosts usually learn their IPv6 gateway addresses automatically from router advertisements, so the IPv6 next hop of the hosts is somewhat random.) This configuration provides both router backup and traffic load sharing in the LAN.

Configuration of SwitchA:

```
SwitchA (config)#interface vlan 1
SwitchA (config)#router ipv6 vrrp 1
SwitchA (config-router)#virtual-ipv6 fe80::2 interface vlan 1
SwitchA (config-router)#priority 150
SwitchA (config-router)#enable
SwitchA (config)#router ipv6 vrrp 2
SwitchA (config-router)#virtual-ipv6 fe80::3 interface vlan 1
SwitchA (config-router)#enable
```

Configuration of SwitchB:

```
SwitchB (config)# interface vlan 1
SwitchB (config)# router ipv6 vrrp 2
SwitchB (config-router)# virtual-ipv6 fe80::3 interface vlan 1
SwitchB (config-router)# priority 150
SwitchB (config-router)# enable
SwitchB (config)# router ipv6 vrrp 1
SwitchB (config-router)# virtual-ipv6 fe80::2 interface vlan 1
SwitchB (config-router)# enable
```

12.11.4 VRRPv3 Troubleshooting

VRRPv3 may not work properly because of incorrect physical connections or configuration. Pay attention to the following:

- ☞ First, ensure that the physical connections are correct.
- ☞ Next, ensure that the interface and link protocol are up (use the **show ipv6 interface** command).

- ☞ Then, ensure that the IPv6 forwarding function is enabled (use the **ipv6 enable** command).
- ☞ Also, ensure that VRRPv3 is enabled on the interface.
- ☞ Check whether the timer settings of the routers (or Layer 3 switches) in the same backup group are the same.
- ☞ Check whether the virtual IPv6 addresses in the same backup group are the same.

Chapter 13 VPN Configuration

13.1 MPLS Overview

13.1.1 MPLS Overview

MPLS (Multiprotocol Label Switching) originated from IPv4 and was first designed to improve forwarding speed. Because "Multiprotocol" means that MPLS supports multiple protocols, its core technology can be extended to other network protocols, including IPv6 (Internet Protocol version 6), IPX (Internet Packet Exchange), AppleTalk, DECnet, and CLNP (Connectionless Network Protocol). MPLS combines fast switching with Layer 3 route forwarding, so it can meet the network requirements of various new applications.

13.1.1.1 MPLS Introduction

Forwarding Equivalence Class

MPLS is a class-based forwarding technology: it puts packets with the same forwarding mode into a class called a FEC (Forwarding Equivalence Class). Packets in the same FEC are treated in the same way in an MPLS network. A FEC is a group of Layer 3 packets that are forwarded along the same path, with the same priority, and in the same mode. Forwarding consists of two steps:

- ☞ Analyze the packet header and classify packets into FECs.
- ☞ Map the FEC to the next hop.

In traditional IP forwarding networks, each router processes the same packet with these two steps. A FEC can include one or more FEC elements. All of them are Layer 3 packets that can be mapped to the same LSP.

Currently, there are two types of FEC elements:

- ☞ Address Prefix: An address prefix identifies a FEC element. Its length ranges from 0 to the full address length. Each Address Prefix FEC element corresponds to a destination subnet.
- ☞ Host Address: A host address identifies a FEC element. Each element corresponds to one host address.

FEC classification rules are very flexible and can be any combination of source address, destination address, source port, destination port, protocol type, VPN, and so on. For example, in traditional IP forwarding with the longest prefix match algorithm, all packets destined for the same destination address belong to one FEC.

Label

In an MPLS network, each FEC is encoded at the edge LSR into a label, a short, fixed-length value. Before packets are forwarded, the label is added to their headers, turning them into labeled packets. In addition to the field that identifies the FEC, a label includes a CoS field, so it represents the FEC, precedence, and service class as a whole. An LSR classifies packets arriving on different ports into different FECs, which is the foundation of VPN. When an LSR creates a new FEC, it also creates a corresponding label and advertises it to all peers. An LSR maintains both incoming and outgoing labels. For load sharing, one FEC may correspond to multiple labels, but one label can represent only one FEC.

Labels are carried in the packet header, contain no topology information, and are only locally significant. A label is 4 bytes long. The following figure shows its encapsulation structure:

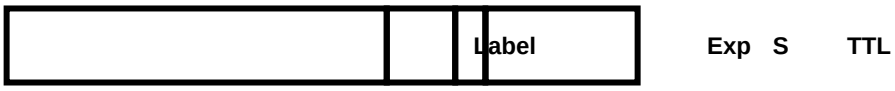


Figure 13-1 Label Encapsulation Structure

A label has four fields:

- Label: The label value, 20 bits long; it is used as a pointer for forwarding.
- Exp: 3 bits, used for QoS.
- S: 1 bit. MPLS supports a hierarchical label structure, that is, multiple label layers. The value 1 indicates the bottom label.
- TTL: 8 bits; it serves the same purpose as the TTL (Time To Live) in IP packets.

Like the VPI/VCI of ATM and the DLCI of Frame Relay, a label identifies a connection. If the link layer protocol has a label field, such as the VPI/VCI of ATM or the DLCI of Frame Relay, the label is encapsulated in that field; otherwise, it is encapsulated in a shim layer between the link layer and the IP layer. Therefore, labels can be supported by any link layer protocol.

Label Space

An LSR can assign a different label to a FEC according to its ingress port. As a result, packets from different ports can be forwarded independently, which is the basic foundation of VPN. To use labels more efficiently, MPLS provides the concept of label space, which acts as a label prefix. Because FECs in different label spaces can be assigned the same label, the label range is effectively expanded. The label space is meaningful only when labels are assigned, not when packets are forwarded.

Label Switching

Non-edge LSRs do not need to analyze the packet header. Instead, they use the label as a pointer to the next-hop egress port and a new label. The LSR replaces the old label of the labeled packet with the new one and forwards the packet through the specified egress port. Label switching simplifies and accelerates forwarding and enables applications such as VPN, QoS, and traffic engineering.

Label Switching Router

The LSR (Label Switching Router) is the basic element of an MPLS network; all LSRs support MPLS.

An LSR is a device that forwards packets according to their label values. An LSR that connects an IP routing network to an MPLS switching network is called an edge LSR. An edge LSR can add labels to IP packets and forward them along an LSP, or remove labels from MPLS packets and forward them according to IP routes. Each LSR must be assigned a globally unique LSR ID, which is usually an interface IP address of the LSR. Assume that LSRs Ru and Rd agree on the binding between label L and FEC F. Packets can then be forwarded from Ru to Rd based on label L. In this case, Ru is the upstream LSR and Rd is the downstream LSR; packets are always forwarded from the upstream LSR to the downstream LSR.

Label Switched Path

The path that a FEC follows in an MPLS network is called an LSP (Label Switched Path). Along the direction of data transmission, two adjacent LSRs on an LSP are called the upstream LSR and the downstream LSR. In the following figure, R2 is the downstream LSR of R1, and R1 is the upstream LSR of R2.

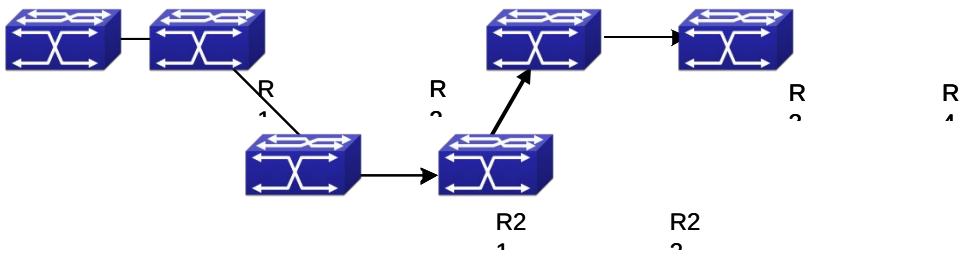


Figure 13-32 Label Switched Path LSP

Like a virtual circuit in ATM or Frame Relay, an LSP is a unidirectional path from the ingress of an MPLS network to its egress. Each router along the LSP is an LSR.

When downstream LSRs advertise labels to their upstream LSRs, the series of labels and the sequence of LSRs form an LSP. An LSP maps IP-layer routing information to a link layer switched path. An LSP is a unidirectional forwarding path along which packets are always forwarded from an upstream LSR to a downstream LSR. To forward packets in the opposite direction, a separate, independent LSP must be created. A FEC is always associated with an LSP; this association between FEC and LSP is called mapping packets to an LSP.

Rules for mapping packets to an LSP:

If exactly one LSP has a host-address FEC element whose address is the same as the packet's destination address, map the packet to that LSP.

If more than one LSP meets condition 1, map the packet to any one of them.

If exactly one LSP has an address-prefix FEC element that matches the packet, map the packet to that LSP.

If more than one LSP meets condition 3, select an LSP based on the longest prefix match principle.

If a packet must pass through a specific egress LSR, and an LSP exists whose bound prefix FEC element is the address of that egress LSR, map the packet to this LSP.

Additional rules:

If the destination address of the packet matches no LSP, the packet is sent along the LSP whose address is the same as its egress router, provided that this LSP has an address-prefix FEC element.

If a packet matches two LSPs, one with a host-address FEC element and the other with an address-prefix FEC element, always map the packet to the first one.

If the packet matches no LSP with a host-address FEC element, it must not be sent along an LSP even if that LSP's host-address FEC element is the same as the packet's egress router address.

LSPs are created based on connections, which result from topology information rather than from data flow demand. That is, the LSP is created whether or not the router forwards any data.

Label Merging

When an LSR maps multiple incoming labels to the same FEC, all of these incoming labels correspond to the same outgoing label and egress port. As a result, when packets with different labels reach the LSR, all outgoing packets carry the same label. This process is called label merging. Label merging can reduce the number of labels in the MPLS domain, but the ingress port information of packets may be lost.

If the LSR does not support label merging and receives multiple label requests, it sends a separate label request to the downstream LSR for each of them, whether or not they have the same FEC. If the LSR supports label merging, it sends only one label request.

Label Distribution Protocol

LDP (Label Distribution Protocol) is the MPLS control protocol, similar to the signaling protocols in traditional networks. Its functions include FEC classification, label distribution, and LSP creation and maintenance.

MPLS supports multiple label distribution protocols, including protocols designed specifically for label distribution, such as LDP and CR-LDP (Constraint-Based Routing using LDP), and existing protocols extended to distribute labels, such as BGP (Border Gateway Protocol) and RSVP (Resource Reservation Protocol). Manually configured static LSPs are also allowed.

LSP Tunnel Technology

MPLS supports LSP tunneling. Even if the path between an upstream LSR and a downstream LSR on an LSP is not provided by the routing protocol, MPLS allows a new LSP to be created between them, with the two LSRs as its start and end points. This new LSP is an LSP tunnel, which avoids traditional network layer tunnel encapsulation.

If a tunnel follows the same route as the routing protocol, it is a hop-by-hop routed tunnel; otherwise, it is an explicitly routed tunnel.

Multi-Layer Label Stack

If a packet is transmitted through more than one layer of LSP tunnels, it carries multiple layers of labels, called a label stack. At the ingress and egress of each tunnel, MPLS pushes or pops a label accordingly.

The label stack follows the last-in-first-out principle, so MPLS processes labels from the top of the stack.

MPLS does not limit the label stack depth. If the label stack depth of a packet is m, the label at the bottom of the stack is level 1, and the label at the top is level m. A packet without any label is treated as having an empty label stack (stack depth 0).

13.1.1.2 MPLS Network Introduction

MPLS Network Structure

As shown in the following figure, the basic unit of an MPLS network is the LSR, and a network of LSRs is called an MPLS domain.

An LSR at the edge of an MPLS domain that connects to other customer networks is called an LER (Label Edge Router), and an LSR inside the domain is a core LSR. Core LSRs can be routers that support MPLS or ATM-LSRs upgraded from ATM switches. LSRs in the domain communicate with each other through MPLS, while the LERs adapt the edge of the MPLS domain to traditional IP technologies.

After the ingress LER pushes a label onto a packet, the packet is transmitted along an LSP made up of a series of LSRs. The ingress LER is called the ingress, the egress LER is called the egress, and the routers in between are called transit routers.

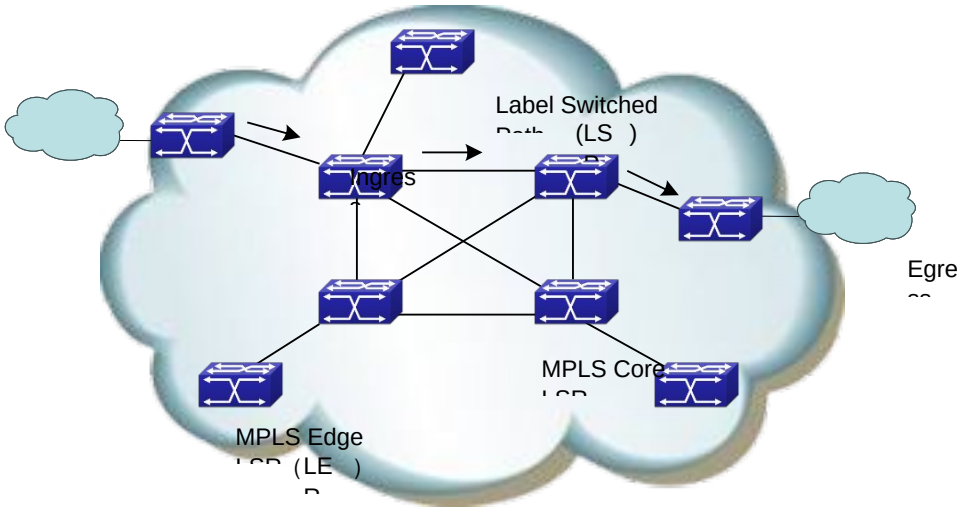


Figure 13-33 The MPLS Network Structure

The basic MPLS working process, based on the figure above, is as follows:

First, LDP and traditional routing protocols (such as OSPF and IS-IS) create the routing table and the LIB (Label Information Base) for FECs that require services.

The ingress LER receives packets, performs Layer 3 functions, determines the FEC of the packets, labels them, and generates MPLS labeled packets.

LSRs in the network then forward the packets according to their labels and the LFIB (Label Forwarding Information Base) without any Layer 3 processing.

Finally, the egress LER removes the label from the packet and forwards it by IP.

In summary, MPLS is neither a service nor an application, but a tunneling technology and a routing and switching platform that integrates label switching with network layer routing. This platform supports various upper-layer protocols and services and provides a degree of information security during transmission.

13.1.1.3 Introduction to MPLS and Routing Protocols

When LDP creates an LSP in hop-by-hop mode, it determines the next hop from the forwarding table of each LSR along the LSP. Because the forwarding table information is collected by routing protocols such as IGP and BGP, LDP is indirectly related to them.

In addition, existing protocols such as BGP and RSVP can also distribute MPLS labels after extension.

Some MPLS applications require extensions to routing protocols. For example, MPLS-based VPN requires a BGP extension so that BGP can distribute VPN (Virtual Private Network) routing information, and MPLS-based TE (Traffic Engineering) requires an OSPF or IS-IS extension to carry link state information.

13.1.1.4 MPLS Application Introduction

MPLS originally combined Layer 2 switching with Layer 3 routing to speed up route lookup. With the development of ASICs (Application-Specific Integrated Circuits), route lookup speed is no longer a network bottleneck, so the forwarding speed advantage of MPLS has disappeared.

However, MPLS combines the powerful Layer 3 routing of IP networks with the efficient forwarding mechanism of traditional Layer 2 networks, and uses a connection-oriented method on the forwarding plane, similar to current Layer 2 networks. As a result, it can easily integrate IP networks with Layer 2 networks such as ATM and Frame Relay, and provides better solutions for applications such as QoS, TE, and VPN.

MPLS-based VPN

Traditional VPNs transmit private data over the public network through tunneling protocols such as GRE, L2TP, and PPTP. Because an LSP is itself a public network tunnel, MPLS has an inherent advantage in implementing VPNs.

MPLS-based VPN connects the different sites of a private network through LSPs to form a single network. It also supports interconnection control between different VPNs.

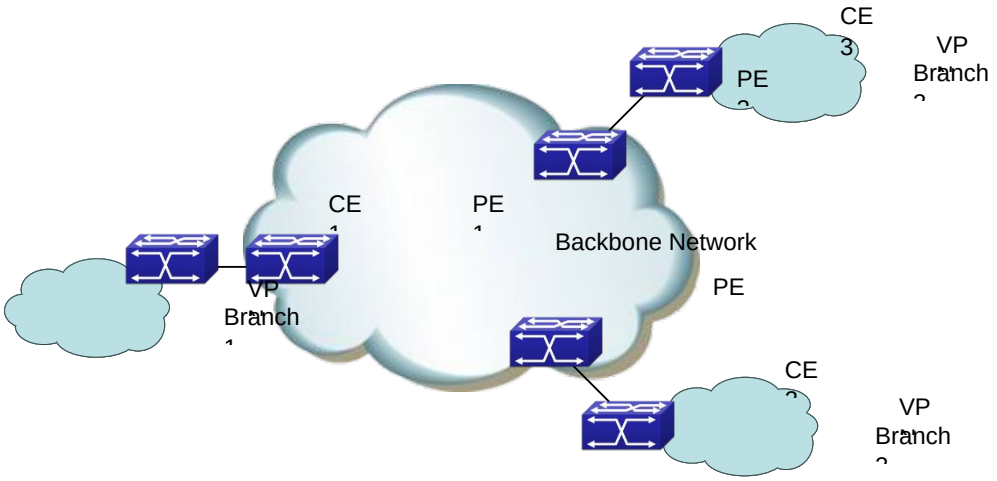


Figure 13-34 MPLS-based VPN

The figure above shows the basic structure of an MPLS-based VPN: a CE (Customer Edge) is a router, switch, or host; a PE (Provider Edge) is located in the backbone network.

A PE manages VPN customers, establishes LSPs between PEs, and distributes routes to the different sites of a VPN customer. Routes are usually distributed between PEs through LDP or extended BGP.

MPLS-based VPN supports overlapping IP addresses between different sites and interconnection between different VPNs. Unlike traditional routes, VPN routes carry additional site and VPN identification, so BGP must be extended to carry VPN routing information.

MPLS-based TE

MPLS-based TE combined with DiffServ can provide different services for traffic of different priorities while keeping network utilization high. It can therefore provide low-delay, low-loss services with guaranteed bandwidth for traffic such as voice and video.

Because TE is difficult to deploy across an entire network, the DiffServ model is usually used to implement QoS in real network solutions.

The basic mechanism of DiffServ is to map a service to a service class at the network edge according to the required quality of service. The service is uniquely identified by the DS field (derived from the ToS field) of the IP packet. Based on this field, routers in the backbone network apply preconfigured service policies to different services to guarantee their quality.

The service class mechanism and marking mechanism of DiffServ are similar to the MPLS label distribution mechanism. In fact, MPLS-based DiffServ is implemented by combining DS assignment with MPLS label distribution.

13.1.1.5 MPLS PHP

In an MPLS network, core LSRs forward packets according to their labels. The egress router (egress LER) removes the label before performing IP forwarding.

In simple MPLS applications where the egress router performs only IP forwarding, the label is actually useless. In such cases, the Penultimate Hop Popping (PHP) feature pops the label at the penultimate router so that the egress router does not need to process labels.

13.2 LDP

13.2.1 LDP Introduction

LDP is used to distribute labels in an MPLS label switching environment and applies only to networks capable of label switching. Working with traditional routing algorithms, LDP distributes labels, advertises <label, FEC> bindings, and creates and maintains the label forwarding information base and LSPs by exchanging messages over TCP connections. In an MPLS VPN environment, LDP is used to distribute public network labels.

LDP does not create routes. Instead, it obtains routes from the system, assigns labels to them, and advertises the labels to its upstream routers. For a FEC that has a downstream, LDP receives a label from the downstream, uses it as the outgoing label, and creates a label switched path, that is, an entry that swaps the incoming label for the outgoing label. If the label assigned by the downstream is 3, LDP creates an entry that pops the label.

LDP was defined in RFC 3036; the latest standard is RFC 5036. It exchanges label-to-route bindings over TCP connections between peers. LDP supports two neighbor discovery modes: the basic mode (automatic discovery) and the extended mode (specified peers). Automatic peer discovery uses UDP multicast messages sent to all routers (224.0.0.2). Both TCP and UDP messages use port 646.

The main process is as follows:

- ☞ Discover and maintain neighbors: After LDP is enabled globally and on an interface, it sends multicast Hello messages on that interface (unless multicast neighbor discovery is disabled) to announce its presence. The Hello messages carry its transport address, which is used for TCP connections. An adjacency is created when Hello messages are received from other LSRs, and it is maintained by sending Hello messages periodically.
- ☞ Establish and maintain sessions: LDP sessions are TCP-based. First, the transport address in the Hello message from the peer is compared with the local one; the end with the larger value becomes ACTIVE and the other becomes PASSIVE. The ACTIVE router initiates the TCP connection (which avoids connection collisions like those between BGP neighbors). After the TCP connection is established, the two ends send Initialization messages to negotiate session parameters. If the negotiation succeeds, the session is established. The two neighbors then send their local interface address lists and label information to each other. KEEPALIVE messages are sent to maintain the connection when there is no data.
- ☞ Create and maintain LSPs: Each pair of LSR peers needs a session to exchange label information; LSPs are created by exchanging FEC-label binding messages.
- ☞ Tear down sessions: If no message is received from the peer for a long time, LDP tears down the session and sends a Notification message to inform the peer.

Note that LDP does not assign labels to default routes or to BGP routes (unless explicitly configured).

13.2.1.1 Basic Concept of LDP

LDP Peer

To distribute a label for a FEC, LDP must advertise the label and its meaning in the MPLS network to create an LSP. LSRs that exchange label information through LDP are LDP peers. LDP peers learn each other's label mappings and other information.

LDP Session

Two LSRs establish an LDP session after exchanging LDP Discovery Hello messages. LSPs rely on LDP sessions to exchange messages such as label mapping and label release.

Establishing an LDP session involves two steps:

- ☞ Establish the transport connection.
- ☞ Initialize the session.

There are two types of LDP sessions:

- ☞ Local LDP session: The two LSRs that establish the session are directly connected.
- ☞ Remote LDP session: The two LSRs that establish the session are not directly connected.

LDP Message Type

There are four types of LDP messages:

- ☞ Discovery message: Announces and maintains the presence of an LSR in the network.
- ☞ Session message: Establishes, maintains, and terminates sessions between LDP peers.
- ☞ Advertisement message: Creates, changes, and deletes label-to-FEC mappings.
- ☞ Notification message: Provides advisory information and error notifications.

To ensure reliable delivery, LDP sends Session, Advertisement, and Notification messages over TCP, and sends Discovery messages over UDP.

13.2.1.2 Introduction to LDP Message Format

LDP PDU

An LDP PDU consists of an LDP header and one or more LDP messages. The LDP header format is as follows:

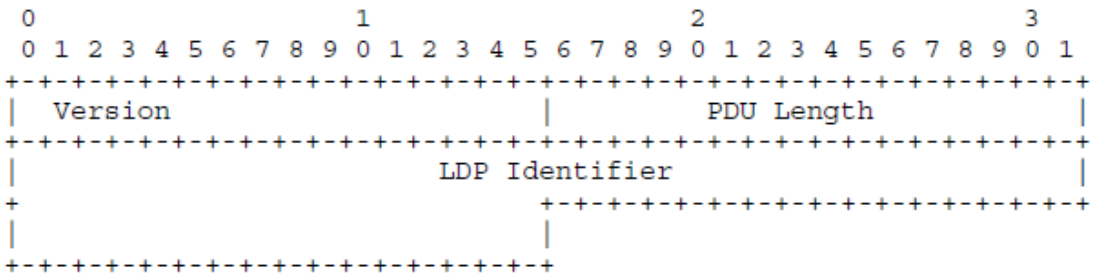


Figure 13-5 LDP Header Format

- ☞ Version: The LDP version, 1 byte. The current LDP version is 1.
- ☞ PDU Length: The total length of the LDP PDU in bytes, 2 bytes.
- ☞ LDP ID: The LDP identifier, 6 bytes. The first 4 bytes are the globally unique LSR ID, and the last 2 bytes are the label space ID, which is 0 for the global label space.

TLV Encoding

LDP encodes the parameters in LDP messages as TLVs (Type-Length-Value). The LDP TLV format is as follows:

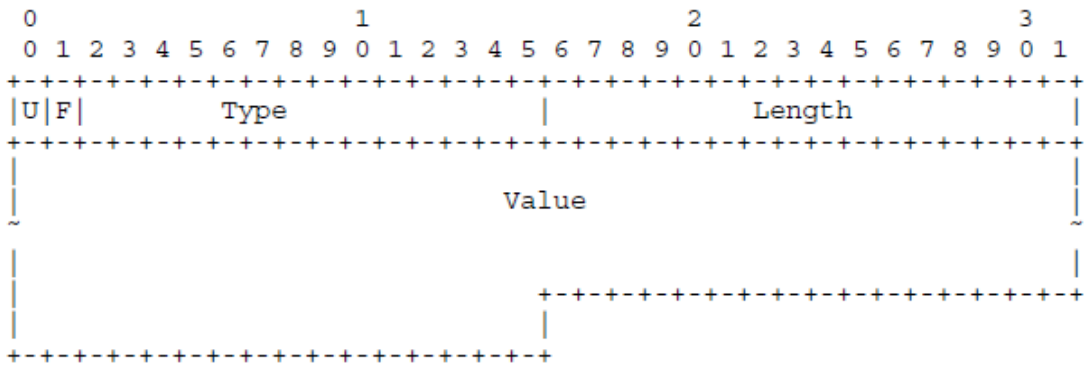


Figure 13-6 LDP TLV Format

- U bit: Unknown TLV flag, 1 bit. If the U flag is 0, the LSR notifies the source LSR of the packet and ignores the whole message; otherwise, it ignores this TLV and processes the other TLVs normally.
- F bit: Forward unknown TLV flag, 1 bit. This flag applies only to LDP messages that contain an unknown TLV with the U bit set to 1. If the F flag is 0, the unknown TLV is not forwarded; otherwise, it is forwarded.
- Type: The TLV type, 14 bits.
- Length: The length of the TLV Value field, 1 byte.
- Value: The Value field, whose length is specified by the Length field.
- The Value field of a TLV can contain other TLVs; that is, TLVs can be nested. TLVs do not need to be aligned on the first byte.

Currently defined TLV types:

TLV	Type
FEC	0x0100
Address List	0x0101
Hop Count	0x0103
Path Vector	0x0104
Generic Label	0x0200
ATM Label	0x0201
Frame Relay Label	0x0202
Status	0x0300
Extended Status	0x0301
Returned PDU	0x0302
Returned Message	0x0303
Common Hello Parameters	0x0400
IPv4 Transport Address	0x0401
Configuration Sequence Number	0x0402
IPv6 Transport Address	0x0403
Common Session Parameters	0x0500
ATM Session Parameters	0x0501
Frame Relay Session Parameters	0x0502
Label Request Message ID	0x0600
Vendor-Private	0x3E00- 0x3EFF
Experimental	0x3F00- 0x3FFF

13.2.1.3 LDP Label Management

In an MPLS system, the downstream LSR decides which label to assign to a specific FEC and notifies the upstream LSR. That is, labels are specified by the downstream LSR and distributed from downstream to upstream.

Label Advertisement Mode

In an MPLS domain, after label switching in the upstream LSR, packets are forwarded to the downstream LSR with the label assigned by the downstream LSR. The FEC labels assigned by the downstream LSR apply only between it and its upstream LSR and must be advertised to the upstream LSR. MPLS defines two label advertisement modes in which the downstream LSR passes labels to its upstream LSR:

- DoD (Downstream On Demand): The LSR assigns and advertises a label for a FEC only after receiving a label request message from the upstream LSR.
- DU (Downstream Unsolicited): The LSR assigns and advertises a label for a FEC without receiving a label request message from the upstream LSR. It sends label mapping information to the upstream LSR automatically.

The two modes can be mixed, with each LSR interface independently configured to use one of them. During initialization, the upstream and downstream LSRs must exchange their label advertisement mode information and agree on a mode; otherwise, LSP creation fails.

The following figure shows the LDP label advertisement process:

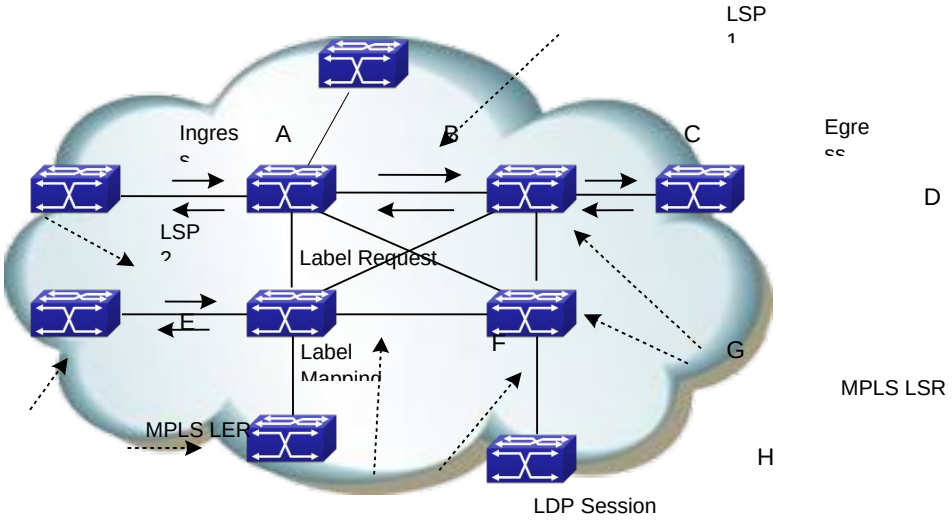


Figure 13-7 Label Advertisement Process

For example, for LSP1 in the figure above, LSR B is the upstream LSR of LSR C, and LSR C is the downstream LSR of LSR B.

The main difference between the two label advertisement modes is whether labels are advertised on demand (DoD) or unsolicited (DU).

The label advertisement process of each mode is as follows:

DoD (Downstream On Demand)

The upstream LSR sends its downstream LSR a Label Request message that carries a FEC description. The downstream LSR assigns a label to this FEC and responds to the upstream LSR with the bound label in a Label Mapping message.

When the downstream LSR sends the Label Mapping message depends on the label distribution control mode it uses.

In Ordered mode, it sends the Label Mapping message to the upstream LSR only after receiving a Label Mapping message from its own downstream LSR.

In Independent mode, it sends the Label Mapping message to the upstream LSR immediately, whether or not it has received a Label Mapping message from its downstream LSR.

The upstream LSR usually selects the downstream LSR based on the routing table. In the figure above, all LSRs along LSP1 work in Ordered mode, while LSR F on LSP2 works in Independent mode.

DU (Downstream Unsolicited)

After the LDP session is established, the downstream LSR automatically advertises Label Mapping messages to its upstream LSR. The upstream LSR saves the messages and processes them according to its label retention mode.

Label Distribution Control Mode

In an MPLS domain, LSRs create an LSP from the ingress router to the egress router by exchanging labels, based on the forwarding path created by the IGP in the domain. Only a complete path is useful for packet forwarding. Creating an LSP is the LSR label advertisement process, so controlling LSP creation means controlling LSR label advertisement. MPLS defines two LSP control modes that determine when the downstream LSR advertises labels to the upstream LSR:

- Ordered mode: An LSR advertises the label mapping of a FEC to its upstream LSR only when it already has the label mapping of the FEC's next hop or when it is the egress router of the FEC. Label advertisement for a flow starts at the egress router of the FEC and binds routers from downstream to upstream, which guarantees that the label-to-flow mapping is complete and consistent across the network. Ordered mode prevents loops more effectively.
- Independent mode: An LSR does not need to wait for the label of the FEC's next hop before advertising labels to its peers. It can send label mappings to connected LSRs at any time. In this mode, an LSR may advertise a label to its upstream LSR before receiving one from its downstream LSR. This mode speeds up LSP creation and aggregation.

An LSR is the egress router of a FEC if:

- The FEC refers to the LSR's own address;
- The FEC's next-hop router is outside the label switching network; or
- The FEC element crosses a routing domain boundary, such as another OSPF summary area or another OSPF or BGP autonomous system.

Label Retention Mode

The label retention mode determines how an LSR handles received label-to-FEC mappings that are currently unused. In DU mode, the upstream LSR may receive many <FEC, label> bindings from downstream LSRs; a binding is useful for label forwarding only when the downstream LSR that sent it is the local next hop of that FEC. MPLS defines two label retention modes to determine how currently unused bindings are handled.

- Conservative mode: The LSR retains only the label mappings received from the neighbor LSR that is its next hop. The advantage of this mode is that it creates and maintains only the labels that are meaningful for data forwarding, which is very important when the label space is limited (ATM switching).
- Liberal mode: The LSR retains the label mappings received from neighbor LSRs whether or not the neighbor is its next hop. The advantage of this mode is that the cost of handling route changes is very low; the disadvantage is that many unused labels are advertised and maintained.

In Liberal label retention mode, the LSR adapts rapidly to route changes; in Conservative mode, the LSR assigns and stores relatively few labels. Conservative retention mode is usually combined with DoD mode on LSRs with limited label space.

Basic Concepts of Label Switching

- NHLFE (Next Hop Label Forwarding Entry): Describes the operation to perform on the label, such as push or swap.
- FTN (FEC to NHLFE): The mapping of a FEC to an NHLFE on the ingress router.
- ILM (Incoming Label Map): The mapping of received labels to NHLFEs on an LSR.

The Label Switching Process

The ingress LER classifies the packets entering the network into FECs. Packets of the same FEC follow the same path (LSP) in the MPLS domain. The LSR assigns a label to incoming FEC packets and forwards them through the corresponding interface.

The label switching process is as follows:

- All LSRs along the LSP first create an ILM, whose entries are the mapping rules for incoming labels.
- The LSR maps the labels of received packets to NHLFEs.
- The LSR finds the corresponding NHLFE in the LIB based on the label, replaces the label with the new label, and forwards the labeled packet.

13.2.1.4 LDP Session

There are four steps in the LDP session process:

- Discovery
- Session establishment and maintenance
- LSP creation
- Session teardown

Discovery

In this step, the LSR sends Hello messages periodically to adjacent LSRs to announce its presence and establish a session. With the basic discovery mechanism, the LSR discovers its LDP peers automatically without manual configuration. There are two discovery mechanisms:

- Basic discovery mechanism

The basic discovery mechanism discovers local LDP peers (LSRs directly connected at the link layer) and creates local LDP sessions. In this mode, the LSR periodically sends LDP Link Hello messages in UDP packets to the "all routers on this subnet" multicast address.

LDP Link Hello messages carry the LDP ID of the interface and other related information. If the LSR receives an LDP Link Hello message on an interface, an LDP peer exists on that interface (at the link layer).

- Extended discovery mechanism

The extended discovery mechanism discovers remote LDP peers (LSRs not directly connected at the link layer) and creates remote LDP sessions. In this mode, the LSR periodically sends LDP Targeted Hello messages in UDP packets to a specified IP address.

LDP Targeted Hello messages carry the LDP ID of the interface and other related information. If the LSR receives an LDP Targeted Hello message on an interface, an LDP peer exists at the network layer.

Session Establishment and Maintenance

After discovering an LDP peer, the LSR establishes the session in two steps:

- Establish the transport layer connection, that is, a TCP connection between the LSRs.

- ☞ Initialize the session between the LSRs and negotiate all related parameters, such as the LDP version, label advertisement mode, timer values, and label space. After the negotiation succeeds, the session is established between the LSRs.

After the session is established, it is maintained by Keepalive messages.

LSP Creation

Creating an LSP means binding FECs to labels and advertising the bindings to adjacent LSRs along the LSP, which is done by LDP. In DoD mode, for example, the main steps are as follows:

When network routes change, if an edge router finds a new destination address in its routing table that belongs to no existing FEC, it creates a new FEC for this destination address. The edge LSR determines the route for the FEC, sends a label request message to its downstream LSR, and specifies the FEC for which the label is requested.

The downstream LSR that receives the label request message saves it, finds the next hop of the FEC in its local routing table, and sends a label request message to its own downstream LSR.

When the label request message reaches the destination router or the egress router of the MPLS network, if the router has available labels and considers the label request valid, it assigns a label to the FEC and sends a label mapping message containing the label information to its upstream LSR.

The LSR that receives the label mapping message checks the label request messages saved locally. If its database contains a label request that corresponds to the FEC label mapping message, the LSR assigns a label to the FEC, adds a new entry to its LFIB, and sends the label mapping information to its upstream LSR.

When the ingress LSR receives the label mapping message, it also adds a corresponding entry to its LFIB, which completes LSP creation.

Session Teardown

LDP maintains adjacencies by monitoring Hello messages and maintains sessions by monitoring Keepalive messages. If no Keepalive message is received within a certain period, the LDP session closes the connection.

Each LDP session can include one or more Hello adjacencies. LDP maintains Hello adjacencies with periodic Hello messages. If no LDP Discovery Hello message is received within a certain period, LDP deletes the Hello adjacency. When the last Hello adjacency in an LDP session is deleted, LDP sends a Notification message and closes the transport connection.

13.2.1.5 LDP Loop Detection

LSP creation in an MPLS domain must also prevent loops. The LDP loop detection mechanism detects and avoids LSP loops.

To detect loops in an MPLS domain, loop detection must be enabled on all LSRs. However, the loop detection configurations on the two ends of an LDP session do not have to be the same.

There are two LDP loop detection methods:

Maximum Hop Count

The hop count is the number of LSRs that label messages (label mapping and label request messages) have passed through. When an LSR transmits label information with the hop count parameter, it first increases the hop count by 1. When the hop count reaches the configured maximum value, a loop exists and LSP creation fails. A hop count of 0 means that the hop count is unknown. The hop count of label messages is always 0. The default maximum hop count is 255.

Path Vector

The path vector records path information in label mapping or label request messages. At each hop, the LSR checks whether its own LSR ID is in the record. Either of the following conditions indicates a loop, and LSP creation fails:

- ☞ The path vector already contains a record of this LSR.
- ☞ The hop count of the path exceeds the configured maximum value.

If its LSR ID is not found, the LSR adds a new record. The maximum path vector length is the same as the maximum hop count.

13.2.2 LDP Configuration

LDP Configuration Task List:

Enable MPLS globally (required)

Enable LDP (required)

Enable or disable the LDP module

Enable or disable label switching on an interface

Enable or disable LDP on an interface

Configure LDP parameters (optional)

Configure the LDP label management mode

Configure the LDP label retention mode

Configure the LDP label advertisement mode

Configure the LDP label control mode

Configure LDP loop detection

Enable or disable LDP loop detection

Set the maximum hop count for LDP loop detection

Configure specified LDP peers

Configure other LDP parameters

Configure the aging time or interval of each timer

Configure the LDP router ID

Configure the LDP TCP transport address

Configure whether LDP discovers peers through multicast Hello messages

Configure whether LDP imports BGP routes

Enable or disable LDP label merging capability

Configure whether LDP sends label release messages

Configure whether LDP retries when a label request is rejected

Configure LDP to receive Hello messages from specified targets

Clear LDP connections or adjacencies

Enable MPLS globally

Command	Explanation
Global Mode	
mpls enable **no mpls enable**	Required Enables MPLS. The no command disables MPLS.

Enable LDP

Basic LDP configuration on the switch is simple. Usually, you only need to enable LDP globally and on the interfaces where LDP will run. Note that label switching must be enabled on interfaces with LDP enabled.

Command	Explanation

Global Mode	
router ldp **no router ldp**	Required Enables LDP. The no command disables LDP. LDP is disabled by default.
Interface Mode	
label-switching **no label-switching**	Required Enables label switching on the interface. The no command disables label switching. Label switching is disabled by default.
ldp {enable|disable}	Required Enables or disables LDP on the interface. LDP is disabled by default.

Configure LDP parameters

Configure the LDP label management mode

Configure the LDP label retention mode

Configure the LDP label advertisement mode

Configure the LDP label control mode

Command	Explanation
Router Mode	
label-retention-mode {conservative| liberal}	Optional Configures the global label retention mode: conservative or liberal. The default is liberal.
advertisement-mode {downstream-on-demand|downstream-unsolicited}	Optional Configures the global label advertisement mode: **downstream-on-demand** or **downstream-Unsolicited.** This mode is related to the other two modes: changing it also changes the label retention mode and the global label path control mode. The default is **downstream-unsolicited**.
control-mode {ordered|independent}	Optional Configures the global label control mode: ordered or **independent**. The default is **independent**.
Interface Mode	
ldp label-retention-mode {conservative | liberal}	Optional Configures the label retention mode of the interface. The default is the same as the global configuration. If the interface configuration differs from the global configuration, the interface configuration takes effect.
ldp advertisement-mode {downstream-on-demand|downstream-unsolicited}	Optional Configures the label advertisement mode of the interface. The default is the same as the global configuration. If the interface configuration differs from the global configuration, the interface configuration takes effect.

Configure LDP loop detection

Enable or disable LDP loop detection

Configure the maximum hop count for LDP loop detection

Command	Explanation
Router Mode	

[no] loop-detection	Optional Enables LDP loop detection. The no command disables it.
[no] loop-detection-count <*count*>	Optional Configures the maximum hop count for LDP loop detection. The default is 255. The no command restores the default value.

Configure specified LDP peers

Command	Explanation
Router Mode	
[no] targeted-peer <*ip-addr*>	Optional Configures a remote targeted LDP peer.

Configure other LDP parameters

Configure the aging time or interval of each LDP timer

Configure the LDP router ID

Configure the LDP TCP transport address

Configure whether LDP discovers peers through multicast Hello messages

Configure whether LDP imports BGP routes

Configure whether LDP enables label merging capability

Configure whether LDP sends label release messages

Configure whether LDP retries when a label request is rejected

Configure LDP to receive Hello messages from specified targets

Command	Explanation
Router Mode	
[no] keepalive-interval <*interval*>	Optional Configures the interval for sending LDP Keepalive messages. The default is 10 seconds. The no command restores the default value.
[no] keepalive-timeout <*time-val*>	Optional Configures the LDP Keepalive timeout. The default is 30 seconds. The no command restores the default value.
[no] Hello-interval <*Hello-interval*>	Optional Configures the interval for sending multicast Hello messages. The default is 5 seconds. The no command restores the default value.
[no] hold-time <*hold-time* >	Optional Configures the hold time for LDP multicast peers. The default is 15 seconds. The no command restores the default value.
[no] targeted-peer-Hello-interval <*Hello -interval*>	Optional Configures the interval for sending Hello messages to specified targets. The default is 15 seconds. The no command restores the default value.
[no] targeted-peer-hold-time <*hold-time*>	Optional Configures the hold time for LDP targeted peers. The default is 45 seconds. The no command restores the default value.

Interface Mode	
[no] ldp keepalive-interval <*interval*>	Optional Configures the interval for sending LDP Keepalive messages on the specified interface. The no command restores the default value.
[no] ldp keepalive-timeout <*time-val*>	Optional Configures the LDP Keepalive timeout on the specified interface. The no command restores the default value.
[no] ldp Hello-interval <*Hello-interval*>	Optional Configures the interval for sending LDP multicast Hello messages on the specified interface. The no command restores the default value.
[no] ldp hold-time <*hold-time*>	Optional Configures the hold time for LDP multicast peers on the specified interface. The no command restores the default value.
[no] ldp targeted-peer-Hello-interval <*Hello-interval*>	Optional Configures the interval for sending LDP Hello messages to specified targets on the specified interface. The no command restores the default value.
[no] ldp targeted-peer-hold-time <*hold-time*>	Optional Configures the hold time for LDP targeted peers on the specified interface. The no command restores the default value.
Router Mode	
[no] router-id <*ip-addr*>	Optional Configures the LDP router ID, which is obtained automatically by default. The no command deletes the manually configured router ID; a valid interface IP address is then obtained automatically as the router ID.
[no] transport-address <*ip-addr*>	Optional Configures the IP address that LDP uses for TCP connections. This address must be the address of a loopback interface in the main VRF. The no command deletes the manual configuration, and LDP selects the TCP address automatically.
[no] multicast-Hellos	Optional Configures LDP to discover peers through multicast Hello messages. The no command disables this function. Multicast Hello is used by default.
[no] import-bgp-routes	Optional Configures LDP to import BGP routes. The no command disables this function. By default, BGP routes are not imported.
[no] global-merge-capability {merge-capable|non-merge-capable}	Optional Configures whether LDP enables the global label merging capability. The no command restores the default value.

[no] propagate-release	Optional Configures LDP to send label release messages to peers. The no command disables this function. By default, label release messages are not sent.
[no] request-retry	Optional Configures LDP to retry 5 times when a label request is rejected. The no command disables retries.
[no] request-retry-timeout <*time-val*>	Optional Configures the retry interval. The default is 5 seconds. The no command restores the default value.

Clear LDP connections or adjacencies

Command	Explanation
Admin Mode	
clear ldp adjacency {<*ip-addr*>|*}	Optional Clears the specified LDP adjacencies; "*" means all adjacencies.
clear ldp session {<*ip-addr*>|*}	Optional Clears the specified LDP sessions; "*" means all sessions.

13.2.3 LDP Typical Examples

Some LDP designs are intended for different network environments. In a typical Ethernet environment, LDP configuration is very simple. With the development of hardware, especially the popularity of Layer 3 switches, pure MPLS networks have lost importance compared with MPLS VPN.

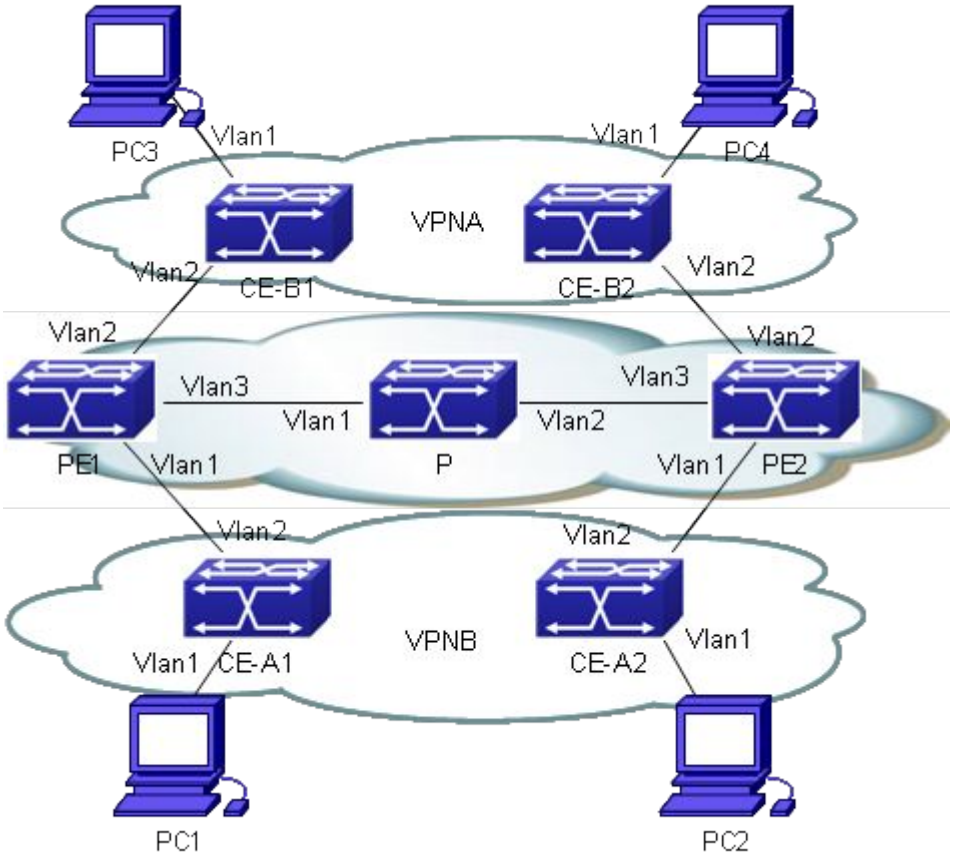


Figure 13-8 MPLS VPN Typical Example

The figure above shows a typical MPLS VPN example, in which PE1, P, and PE2 form the public network area, where switching is performed through MPLS. CE-A1 and CE-A2 form VPN-A, and CE-B1 and CE-B2 form VPN-B. Both VPNs communicate through label switching in the public network, so LDP must be configured to distribute and advertise labels in the public network area. To ensure route reachability, routes are advertised through OSPF.

LDP configuration on PE1:

```
PE1#config
PE1(config)#mpls enable
PE1(config)# router ldp
PE1(config-router)#exit
PE1(config)#interface vlan 3
PE1(config-if-Vlan3)#ip address 202.200.1.2 255.255.255.0
PE1(config-if-Vlan3)#ldp enable
PE1(config-if-Vlan3)#label-switching
PE1(config-if-Vlan3)#exit
PE1(config)#router ospf
PE1(config-router)#network 200.200.1.1/32 area 0
PE1(config-router)#network 202.200.1.0/24 area 0
PE1(config-router)#exit
```

LDP configuration on P:

```
P#config
P(config)#mpls enable
P(config)# router ldp
P(config-router)#exit
P(config)#interface vlan 1
P(config-if-Vlan1)#ip address 202.200.1.1 255.255.255.0
P(config-if-Vlan1)#ldp enable
P(config-if-Vlan1)#label-switching
P(config-if-Vlan1)#exit
```

```
P(config)#interface vlan 2
P(config-if-Vlan2)#ip address 202.200.2.1 255.255.255.0
P(config-if-Vlan2)#ldp enable
P(config-if-Vlan2)#label-switching
P(config-if-Vlan2)#exit
P(config)#router ospf
P(config-router)#network 202.200.1.0/24 area 0
P(config-router)#network 202.200.2.0/24 area 0
P(config-router)#exit
```

LDP configuration on PE2:

```
PE2#config
PE2(config)#mpls enable
PE2(config)# router ldp
PE2(config-router)#exit
PE2(config)#interface vlan 3
PE1(config-if-Vlan3)#ip address 202.200.2.2 255.255.255.0
PE2(config-if-Vlan3)#ldp enable
PE2(config-if-Vlan3)#label-switching
PE2(config-if-Vlan3)#exit
PE2(config)#router ospf
PE2(config-router)#network 200.200.1.2/32 area 0
PE2(config-router)#network 202.200.2.0/24 area 0
PE2(config-router)#exit
```

For the BGP configuration, see the BGP/MPLS VPN typical examples.

13.2.4 LDP Troubleshooting

When LDP is configured and used, problems such as incorrect physical connections and configuration errors may cause LDP to fail. To avoid them, note the following:

- First, make sure that LDP is enabled globally and on the active interface. Note that LDP can be enabled on an interface only after it is enabled globally.
- Second, after the connection is established, use the "show ldp interface" command to check whether LDP is correctly enabled on the interface. If LDP is enabled but the interface is not displayed, the interface may not be up, or label switching may not be configured on it.
- Next, make sure that the adjacent interfaces are in the same subnet, and use the "show ldp adjacency" command to check whether LDP discovers peers and establishes adjacencies with them. If no peer is discovered or no adjacency is established, the interfaces may be in different subnets, or the local device or its remote neighbor may have multicast Hello disabled. In addition, the LSR ID is the default address used to establish the TCP connection, so make sure that the route to the LSR ID is advertised to the remote end.
- Use the "show ldp session" command to check whether the LDP session state is operational; LDP sessions can exchange messages only in this state. If the LDP session cannot be established, use "show ldp" to check the TCP addresses of both peers, and check the routing table to make sure that the route to the remote end is reachable.

Finally, if all the above steps succeed, use "show ldp fec" to check the routes imported by LDP and their information, or use "show mpls ftn" and "show mpls ilm" to check the created entries.

13.3 MPLS VPN

13.3.1 BGP/MPLS VPN Introduction

13.3.1.1 BGP/MPLS VPN Network Structure

BGP/MPLS VPN is a PE-based L3VPN technology among the VPN solutions offered by service providers. It uses BGP to advertise VPN routes and MPLS to forward VPN packets in the provider backbone network.

BGP/MPLS VPN networking is flexible and scalable, and it supports MPLS QoS and MPLS TE easily, so it is increasingly widely used.

The BGP/MPLS VPN model consists of three parts: CE, PE, and P.

- P router: Provider router. It is located in the MPLS domain and switches MPLS data flows quickly based on labels. A P router receives MPLS packets, swaps their labels, and forwards them.
- PE router: Provider Edge router. It is located at the edge of the MPLS domain and converts between IP packets and MPLS packets. A PE router receives IP packets, pushes MPLS labels, and outputs MPLS packets; or it receives MPLS packets, pops their labels, and outputs IP packets. On a PE router, ports connected to P routers or other PE routers are "public network ports" and are configured with public network IP addresses; ports connected to CE routers are "private network ports" and are configured with private network addresses.
- CE router: Customer Edge router. It is located at the edge of the customer IP domain and connects directly to a PE router. It aggregates customer data and forwards the routing information of the customer IP domain to the PE router.

The following figure shows a BGP/MPLS VPN network:

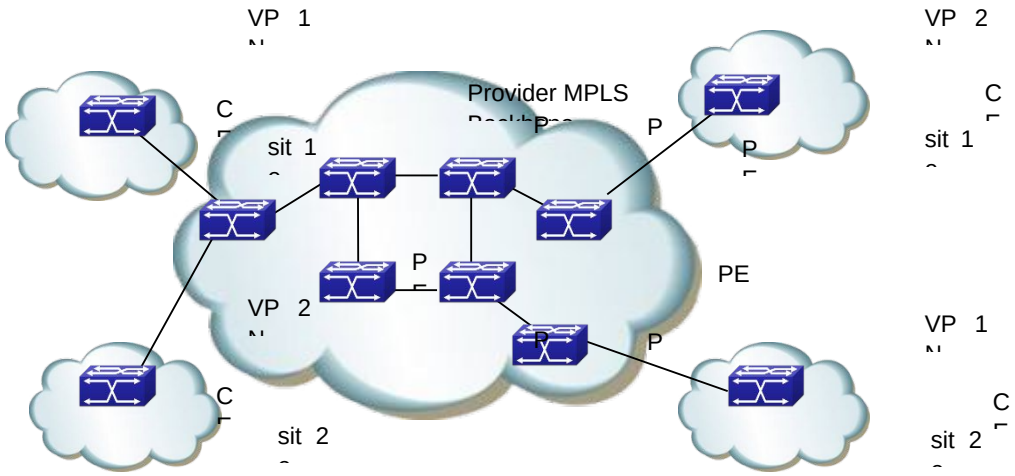


Figure 13-35 BGP/MPLS VPN Networking

CE and PE are divided according to the management areas of the SP and the customer; CE and PE form the boundary between the two areas.

A CE is usually a router. After it establishes an adjacency with its directly connected PE, the CE advertises local VPN routes to the PE and learns remote VPN routes from the PE. CE and PE exchange routing information through BGP/IGP or static routes.

After learning local VPN routes from a CE, a PE exchanges VPN routing information with other PEs through BGP. A PE maintains only the routes of the VPNs directly connected to it, not all VPN routes in the service provider network.

A P router maintains only routes to PEs and does not learn any VPN routing information.

When VPN traffic is transmitted in the MPLS backbone network, the ingress PE acts as the ingress LSR (Label Switching Router), the egress PE acts as the egress LSR, and the P router acts as the transit LSR.

13.3.1.2 Basic Concepts of BGP/MPLS VPN

Site

"Site" is a concept often used in VPN descriptions. It can be understood as follows:

- A site is a set of IP systems that have IP connectivity with each other. This connectivity does not depend on the SP network.
- Sites are divided according to device topology, not device location, although in most cases the devices in a site are physically close to each other.
- The devices in a site can belong to multiple VPNs; in other words, a site can belong to multiple VPNs.
- A site connects to the SP network through a CE. One site can include multiple CEs, but a CE can belong to only one site.

Multiple sites connected to the same SP network can be divided into different sets according to specific policies, and only sites in the same set are allowed to communicate with each other through the SP network. Such a set is a VPN.

VRF

A VRF (VPN Routing and Forwarding instance) consists of a VPN IP routing table and a VPN IP forwarding table (the forwarding table contains MPLS encapsulation information); it is the core of MPLS VPN packet forwarding. Each VPN has its own independent VRF. The address spaces of the VRFs of different VPNs can overlap. A PE/P router in an MPLS VPN network usually contains multiple independent VRFs.

Overlapping Address Space

A VPN is a private network, so each VPN manages its own address range independently. This range is called the address space.

The address spaces of different VPNs may partially overlap. For example, if both VPN1 and VPN2 use the 10.110.10.0/24 subnet, their address spaces overlap.

VPN Instance

In MPLS VPN, routes of different VPNs are isolated by VPN instances.

A PE creates and maintains a dedicated VPN instance for each directly connected site. The VPN instance contains the VPN membership and routing rules of the corresponding site. If the users of a site belong to more than one VPN, its VPN instance contains the information of all those VPNs.

To ensure VPN data independence and security, each VPN instance on a PE has its own independent routing table and LFIB (Label Forwarding Information Base).

Specifically, a VPN instance contains the LFIB, the IP routing table, the interfaces bound to the VPN instance, and management information (including the RD, route filtering policy, member interface list, and so on).

VPN-IPv4 Address

Traditional BGP cannot correctly handle VPN routes with overlapping address spaces. For example, if VPN1 and VPN2 both use the 10.110.10.0/24 subnet and each advertises a route to this subnet, BGP selects only one of the two routes, and the route to the other VPN is lost.

PE routers use MP-BGP to advertise VPN routes to each other and solve this problem with the VPN-IPv4 address family.

A VPN-IPv4 address is 12 bytes long: an 8-byte RD (Route Distinguisher) and a 4-byte IPv4 address prefix.

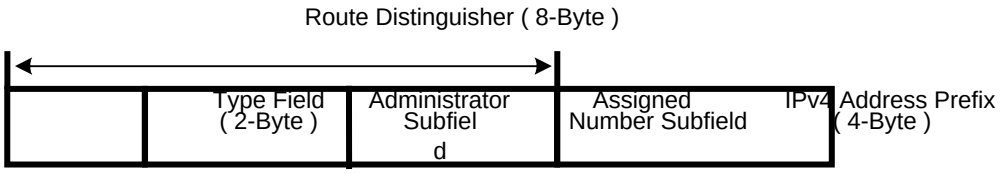


Figure 13-36 VPN-IPv4 Address Structure

After receiving ordinary IPv4 routes from a CE, a PE must advertise these private VPN routes to the remote PE. The private network routes are kept independent by the RD added to them.

An SP can allocate globally unique RDs independently. Therefore, even if VPNs in different SP networks use the same IPv4 address space, the PE routers can advertise distinct routes for them.

It is recommended to allocate a dedicated RD to each VPN instance on a PE, so that all routes to the same CE use the same RD. A VPN-IPv4 address whose RD is 0 is a globally unique IPv4 address.

Adding an RD to an IPv4 prefix makes the prefix globally unique; this is the purpose of the RD.

An RD can be based on an ASN, in which case it combines an ASN and an arbitrary number; or it can be based on an IP address, in which case it combines an IP address and an arbitrary number.

There are two RD formats, distinguished by a 2-byte Type field:

- If Type is 0, the Administrator subfield is 2 bytes and the Assigned Number subfield is 4 bytes. The format is a 16-bit ASN followed by a 32-bit user-defined number, for example, 100:1.
- If Type is 1, the Administrator subfield is 4 bytes and the Assigned Number subfield is 2 bytes. The format is a 32-bit IPv4 address followed by a 16-bit user-defined number, for example, 172.1.1.1:1.

To ensure the global uniqueness of RDs, do not set the Administrator subfield to a private ASN or a private IP address.

VPN Target Attribute

BGP/MPLS VPN uses a 32-bit BGP extended community attribute, VPN Target (also called Route Target), to control the advertisement of VPN routing information.

The VPN instances on PE routers use two types of VPN Target attributes:

- Export Target attribute: Before advertising the VPN-IPv4 routes learned from its directly connected sites to other PEs, the local PE sets the Export Target attribute for these routes.
- Import Target attribute: When a PE receives VPN-IPv4 routes advertised by other PE routers, it checks their Export Target attributes and adds the routes to the corresponding VPN routing table only if their Export Target attributes match the Import Target attributes of its VPN instances.

In other words, the VPN Target attribute defines which sites can accept a VPN-IPv4 route and from which sites a PE router can receive routes.

Like RDs, VPN Targets have two formats:

- 16-bit ASN:32-bit user-defined number, for example, 100:1
- 32-bit IPv4 address:16-bit user-defined number, for example, 172.1.1.1:1

MP-BGP

MP-BGP (Multiprotocol Extensions for BGP-4) carries VPN information and routes between PE routers. MP-BGP is backward compatible and supports both the traditional IPv4 address family and other address families (such as the VPN-IPv4 address family). It ensures that private VPN routes are advertised only within the VPN and enables communication between MPLS VPN members.

Routing Policy

In addition to controlling VPN route advertisement with the import and export extended communities, you can use import or export routing policies to control the import and advertisement of VPN routes more precisely.

The import routing policy filters the routes that can be imported into a VPN instance according to their VPN Target attribute; it can reject routes specified by the communities in the import list. The export routing policy can prevent the advertisement of routes specified by the communities in the export list.

After creating VPN instances, you can choose whether to configure import or export routing policies.

Tunneling Policy

A tunneling policy selects the tunnels used for packets of specified VPN instances.

The tunneling policy is optional and can be configured after VPN instances are created. By default, an LSP is selected as the tunnel without load sharing (the load-sharing number is 1). This policy takes effect only within one AS.

13.3.1.3 Forwarding BGP/MPLS VPN Packets

In basic L3VPN applications (excluding multi-AS VPN), VPN packets are forwarded with two layers of labels:

- The first-layer (outer) label is switched within the backbone network and indicates an LSP from the local PE to the remote PE. With this label, VPN packets reach the remote PE along the LSP.
- The second-layer (inner) label is used when a packet travels from the remote PE to the CE. It indicates the site, or more specifically the CE, to which the packet is sent, so the remote PE can find the correct interface to forward the packet based on this label.

In special cases, two sites of the same VPN connect to the same PE; then the only question is how to reach the remote CE.

The following figure shows an example of VPN packet forwarding:

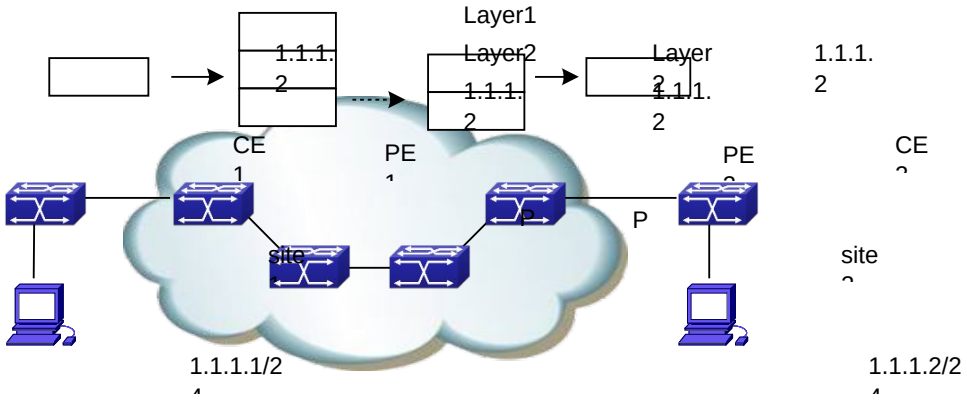


Figure 13-37 Forwarding VPN Packets

Site1 sends an IP packet with destination address 1.1.1.2; CE1 sends it to PE1.

PE1 looks up the VPN instance entries according to the receiving interface and the destination address. If a match is found, PE1 adds two labels (inner and outer) to the packet and forwards it.

The MPLS network forwards the packet to PE2 according to its outer label. The outer label is removed at the hop before PE2, leaving only the inner label.

PE2 looks up the VPN instance entries according to the inner label and the destination address, determines the egress interface, and forwards the packet to CE2.

CE2 forwards the packet to its destination through the normal IP forwarding process.

13.3.1.4 BGP/MPLS VPN Networking Solutions

In BGP/MPLS VPN networks, the advertisement and reception of VPN routes between sites are controlled by the VPN Target attribute. VPN Export Targets and Import Targets are configured independently, and each can have multiple values, enabling flexible VPN access control and various VPN networking solutions.

Basic VPN

In the most basic case, all users of a VPN form a closed user group: traffic can be forwarded between them, but no user in the VPN can communicate with users outside it.

In this networking, each VPN is assigned a dedicated VPN Target as both its Export Target and Import Target; this VPN Target must not be used by other VPNs.

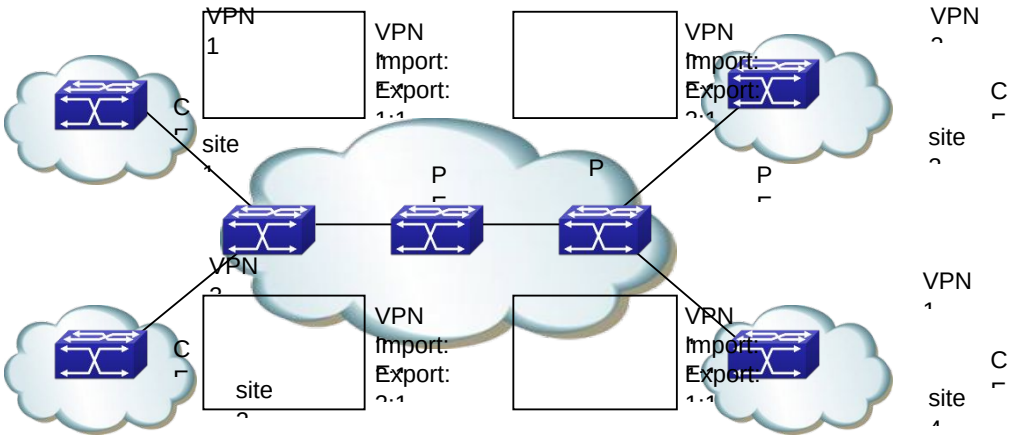


Figure 13-12 Basic VPN Networking Solution

In the figure above, the PE assigns VPN Target 100:1 to VPN1 and 200:1 to VPN2. The sites of VPN1 can communicate with each other, and so can the two sites of VPN2, but sites in VPN1 cannot communicate with sites in VPN2.

Hub&Spoke VPN

To control communication between users through a central access control device in the VPN, use the Hub&Spoke networking solution. The central device can then monitor and filter the communication between devices at both ends.

This networking requires two VPN Targets: one for "Hub" and one for "Spoke".

All sites follow these rules when configuring VPN Targets for the VPN instances on PEs:

- ☞ Spoke-PE: The Export Target is "Spoke" and the Import Target is "Hub".
- ☞ Hub-PE: Two interfaces or subinterfaces are required. One receives routes from Spoke-PEs, and the Import Target of its VPN instance is "Spoke"; the other advertises routes to Spoke-PEs, and the Export Target of its VPN instance is "Hub".

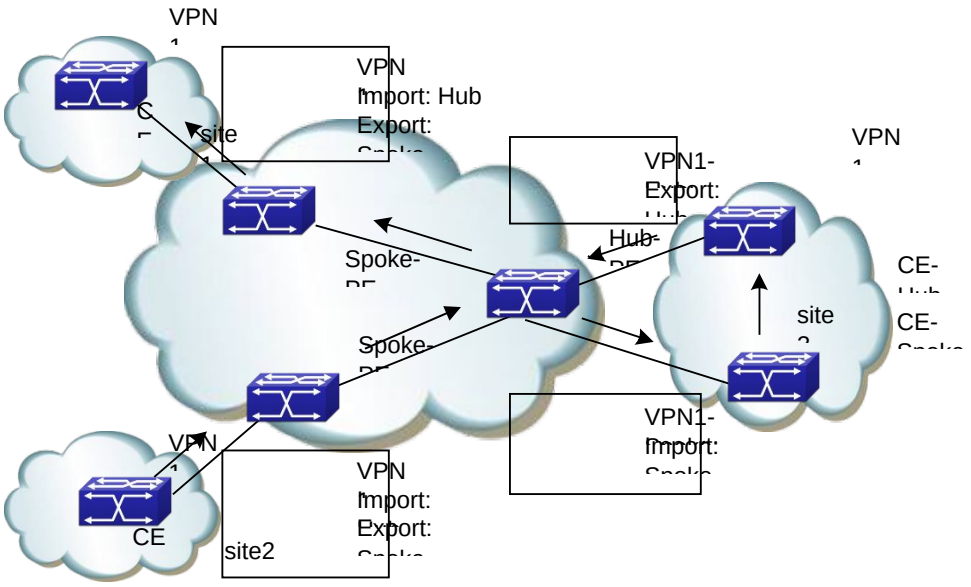


Figure 13-13 Hub&Spoke Networking Solution

In the figure above, Spoke sites communicate with each other through the Hub site (the arrows show the route advertisement process from site2 to site1):

- The Hub-PE can receive the VPN-IPv4 routes advertised by all Spoke-PEs.
- All Spoke-PEs can receive the VPN-IPv4 routes advertised by the Hub-PE.
- Because the Hub-PE advertises the routes learned from Spoke-PEs to other Spoke-PEs, Spoke sites can communicate with each other through the Hub site.
- The Import Target attribute of any Spoke-PE differs from the Export Target attributes of the other Spoke-PEs. Therefore, no two Spoke-PEs can advertise VPN-IPv4 routes to each other or communicate directly.

Extranet VPN

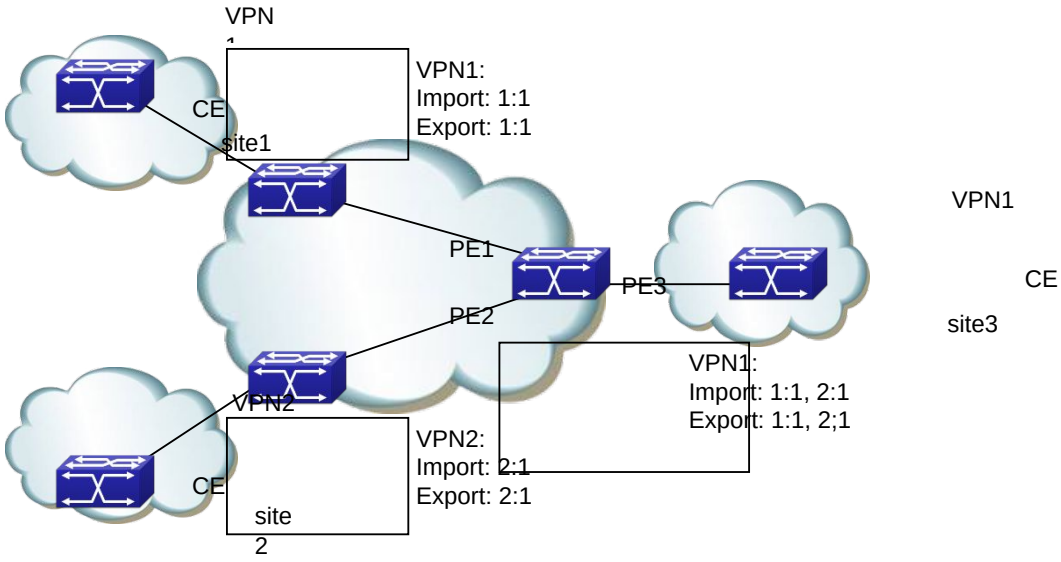


Figure 13-14 Extranet Networking Solution

If a VPN user wants to provide resources of some of its sites to users outside the VPN, use the Extranet networking solution.

In this networking, if a VPN needs to access the shared site, its Export Target must be included in the Import Targets of the shared site's VPN instance, and its Import Target must be included in the Export Targets of the shared site's VPN instance.

In the figure above, site3 of VPN1 can be accessed by both VPN1 and VPN2:

- ☞ PE3 can receive the VPN-IPv4 routes advertised by PE1 and PE2.
- ☞ PE1 and PE2 can receive the VPN-IPv4 routes advertised by PE3.
- ☞ Therefore, site1 and site3 of VPN1 can communicate with each other, and so can site2 of VPN2 and site3 of VPN1.

PE3 does not advertise the VPN-IPv4 routes from PE1 to PE2, or the VPN-IPv4 routes from PE2 to PE1 (routes learned from an IBGP neighbor are not sent to other IBGP neighbors), so site1 of VPN1 and site2 of VPN2 cannot communicate with each other.

13.3.1.5 BGP/MPLS VPN Route Advertisement

In basic BGP/MPLS VPN networks, VPN route advertisement involves only CEs and PEs, because P routers maintain only backbone routes and need no VPN routing information. A PE router maintains only the routes of the VPNs directly connected to it, not all VPN routes, so a BGP/MPLS VPN network scales easily.

To create a reachable route from the local CE to the remote CE and advertise private VPN routing information across the backbone network, the VPN route advertisement process has three parts: from the local CE to the ingress PE, from the ingress PE to the egress PE, and from the egress PE to the remote CE.

The three parts are described below:

Route information exchange from the local CE to the ingress PE

After establishing an adjacency with its directly connected PE, the CE sends its local VPN routes to the PE.

The CE can use static routes, RIP, OSPF, IS-IS, or EBGP to send routes to the PE; all are sent as standard IPv4 routes.

Route information exchange from the ingress PE to the egress PE

The PE adds the RD and VPN Target attributes to the VPN routes learned from the CE and stores these VPN-IPv4 routes in the VPN instance created for the CE.

The ingress PE advertises the VPN-IPv4 routes to the egress PE through MP-BGP. The egress PE determines whether to add each route to the routing table of a VPN instance by comparing the route's Export Target attribute with the Import Targets of the VPN instances it maintains.

PEs ensure reachability between each other through an IGP.

Route information exchange from the egress PE to the remote CE

As from the local CE to the ingress PE, the remote CE can learn VPN routes from the egress PE in several ways, including static routes, RIP, OSPF, IS-IS, and EBGP.

13.3.1.6 Multi-AS VPN Introduction

In real networks, multiple sites of a user's VPN may connect to SPs with different ASNs, or to different ASs of the same SP. Such a VPN that spans multiple autonomous systems is called a multi-AS VPN. RFC 2547 provides three multi-AS VPN solutions:

- ☞ VRF-to-VRF: ASBRs use VRF interfaces to establish EBGP neighbors and manage VPN routes. This is also called Inter-Provider Option A.
- ☞ EBGP redistribution of labeled VPN-IPv4 routes: ASBRs use MP-EBGP to advertise labeled VPN-IPv4 routes. This is also called Inter-Provider Option B.

- ☞ Multihop EBGP redistribution of labeled VPN-IPv4 routes: PEs use multihop MP-EBGP to advertise labeled VPN-IPv4 routes. This is also called Inter-Provider Option C.

The switch currently supports the first solution: the VRF-to-VRF multi-AS VPN solution.

Multi-AS VPN solution

As shown in the following figure, in this mode the PE routers of the two ASs connect directly to each other and act as the ASBRs of their own ASs. These PE routers (ASBRs) connect to each other through VRF interfaces, import all RTs that the local system needs the other end to learn, export all RTs that the local system wants to obtain from the other end, and establish EBGP connections through the VRF interfaces. As a result, the CEs they serve can communicate with or be isolated from each other as if they were in the same AS, and each of the two PE routers treats the other as its own CE. Within each AS, packets are forwarded as VPN packets with two layers of labels; between ASBRs, they are forwarded as ordinary IP packets.

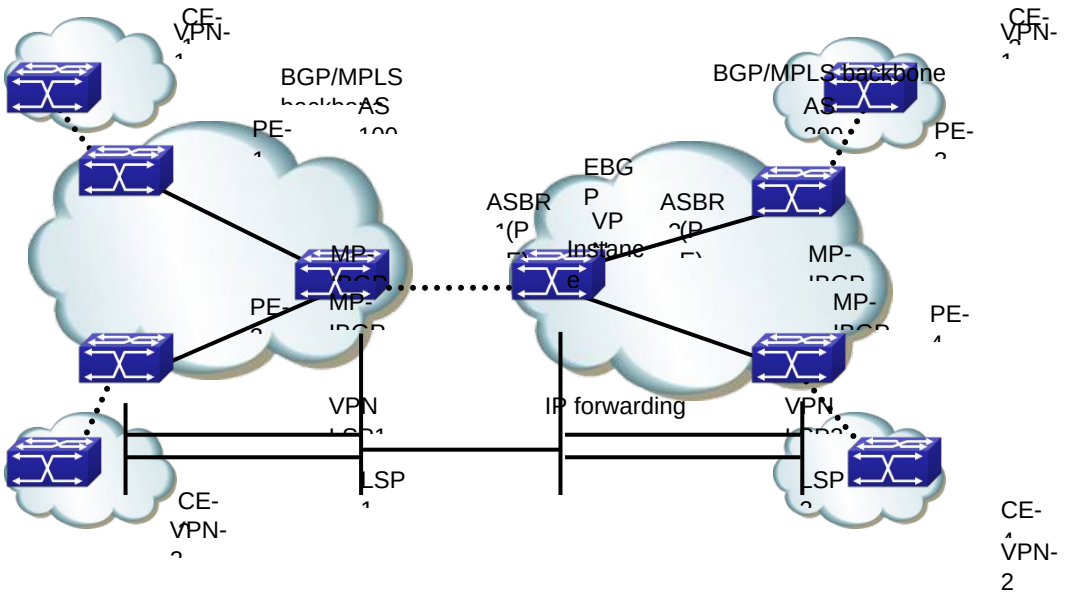


Figure 13-38 Multi-AS VPN Networking

- ☞ The advantage of this multi-AS VPN mode is that it is easy to implement: no special configuration is required between the two PEs acting as ASBRs.
- ☞ The disadvantage is poor scalability: the PEs acting as ASBRs must manage all VPN routes and create a VPN instance for each VPN, which results in too many VPN-IPv4 routes on the PEs.

13.3.2 BGP MPLS VPN Configuration

BGP MPLS VPN configuration task sequence:

Enable MPLS globally (required)

Configure VPN instances (required)

Create a VPN instance and enter VPN instance view

Configure the RD of the VPN instance

Configure the RT of the VPN instance

Associate the VPN instance with an interface

Configure basic MPLS VPN (required)

Use EBGP between PE and CE

Configure the remote PE as a public network VPNv4 neighbor

Enter BGP-VPN instance view

Configure the CE as a VPN private network neighbor

Advertise local private network routes

Use OSPF between PE and CE

Configure the remote PE as a public network VPNv4 neighbor

Create an OSPF instance between PE and CE and enter Router OSPF view

Enable OSPF on the subnet between PE and CE

Redistribute BGP routes

Enter BGP-VPN instance view

Redistribute OSPF routes

Advertise local private network routes

Use RIP between PE and CE

Configure the remote PE as a public network VPNv4 neighbor

Enter RIP VPN instance view

Enable RIP on the subnet between PE and CE

Redistribute BGP routes

Enter BGP-VPN instance view

Redistribute RIP routes

Advertise local private network routes

Use static routes between PE and CE

Configure the remote PE as a public network VPNv4 neighbor

Configure static VPN routes

Enter BGP-VPN instance view

Redistribute static routes

Advertise local private network routes

Enable MPLS (required)

Command	Explanation
Global Mode	
mpls enable **no mpls enable**	Required Enables MPLS. The no command disables MPLS.

Configure VPN instances (required)

Create a VPN instance and enter VPN instance view

Configure the RD of the VPN instance

Configure the RT of the VPN instance

Associate the VPN instance with an interface

Command	Explanation
Global Mode	
[no] ip vrf <*vrf-name*>	Required Creates a VPN instance. No VPN instance is created by default.
VRF Mode	

rd <*ASN:nn*_or_*IP-address:nn*>	Required Configures the RD of the VPN instance. No RD is configured by default.
[no] route-target {import | export | both} <*rt-value*>	Required Configures the RT of the VPN instance.
Interface Mode	
[no] ip vrf forwarding < ***vrf-name*** >	Required Associates the VPN instance with the interface.
[no] ip address <*ip-address*> <*mask*>	Required Configures the private network IP address of the PE interface directly connected to the CE.

3 **Configure basic MPLS VPN (required)**

(1) Use EBGP between PE and CE

Configure the remote PE as a public network VPNv4 neighbor

Enter BGP-VPN instance view

Configure the CE as a VPN private network neighbor

Advertise local private network routes

Command	Explanation
BGP Router Mode	
neighbor <*ip-address*> remote-as <*as-num*>	Required Configures the remote PE as a public network VPNv4 neighbor. It is recommended to use loopback interfaces to establish BGP neighbors between public network PEs.
neighbor <*ip-address*> update-source <*as-num*>	Specifies the local loopback interface used to establish the neighbor.
BGP-VPNv4 Mode	
address-family vpnv4 [unicast]	Required Creates the BGP VPNv4 address family. No VPNv4 address family is created by default.
[no] neighbor <*ip-address*> active	Optional Activates neighbors in VPNv4. All neighbors in VPNv4 view are inactive by default.
BGP Router Mode	
[no] address-family ipv4 {unicast| multicast|vrf <*vrf-nam*>}	Optional Creates the BGP IPv4 address family and enters BGP-VPN instance view. No IPv4 address family is created by default.
BGP-VPN Instance Mode	
[no] neighbor <*ip-address*> remote-as <*as-num*>	Optional Configures the CE as a VPN private network neighbor. No private network neighbor is configured by default.
[no] neighbor <*ip-address*> active	Optional Activates neighbors in VPNv4. All neighbors in VPNv4 view are active by default.
[no] redistribute {connected|ospf|rip| static}	Optional Redistributes directly connected routes and routes of other protocols. No routes are redistributed by default.

Use OSPF between PE and CE

Configure the remote PE as a public network VPNv4 neighbor

Create an OSPF instance between PE and CE and enter Router OSPF view

Enable OSPF on the subnet between PE and CE

Redistribute BGP routes

Enter BGP-VPN instance view

Redistribute OSPF routes

Advertise local private network routes

Command	Explanation
BGP Router Mode	
neighbor <*ip-address*> remote-as <*as-num*>	Required Configures the remote PE as a public network VPNv4 neighbor. It is recommended to use loopback interfaces to establish BGP neighbors between public network PEs.
neighbor <*ip-address*> update-source <*as-num*>	Specifies the local loopback interface used to establish the neighbor.
BGP-VPNv4 Mode	
address-family vpnv4 [unicast]	Required Creates the BGP VPNv4 address family. No VPNv4 address family is created by default.
[no] neighbor <*ip-address*> active	Optional Activates neighbors in VPNv4. All neighbors in VPNv4 view are active by default.
Global Mode	
[no] router ospf [<*process_id*> [<*vrf-nam*>]]	Optional Creates an OSPF instance between PE and CE and enters Router OSPF view.
OSPF VPN Instance Mode	
[no] network {<network> <mask> | <network>/<prefix>} area <area_id>	Optional Enables OSPF on the subnet between PE and CE. OSPF is not enabled on any subnet by default.
[no] redistribute { bgp | connected | static| rip | kernel} [metric-type {1 | 2}] [tag <tag>] [metric <cost_value>] [router-map <WORD>]	Optional Redistributes BGP routes. No routes are redistributed by default.
BGP Router Mode	
[no] address-family ipv4 {unicast| multicast|vrf <*vrf-nam*>}	Optional Creates the BGP IPv4 VRF address family and enters BGP-VPN instance view. It is not created by default.
BGP-VPN Instance Mode	
[no] redistribute {connected | ospf | rip | static}	Optional Redistributes directly connected routes and routes of other protocols. No routes are redistributed by default.

Use RIP between PE and CE

Configure the remote PE as a public network VPNv4 neighbor

Enter RIP VPN instance view

Enable RIP on the subnet between PE and CE

Redistribute BGP routes

Enter BGP-VPN instance view

Redistribute RIP routes

Advertise local private network routes

Command	Explanation
BGP Router Mode	
neighbor <*ip-address*> remote-as <*as-num*>	Required Configures the remote PE as a public network VPNv4 neighbor. It is recommended to use loopback interfaces to establish BGP neighbors between public network PEs.
neighbor <*ip-address*> update-source <*as-num*>	Specifies the local loopback interface used to establish the neighbor.
BGP-VPNv4 Mode	
address-family vpnv4 [unicast]	Required Creates the BGP VPNv4 address family. No VPNv4 address family is created by default.
[no] neighbor <*ip-address*> active	Optional Activates neighbors in VPNv4. All neighbors in VPNv4 view are active by default.
RIP Router Mode	
[no] address-family ipv4 vrf <*vrf-name*>	Optional Creates the RIP IPv4 address family and enters RIP VPN instance view.
RIP VPN Instance Mode	
[no] network {*A.B.C.D/M*|*ifname*|*vlan <id>*|loopback <*1-1024*> }	Optional Enables RIP between PE and CE.
[no] redistribute { kernel | connected | static | ospf | isis | bgp} [metric <value>] [route-map<word>]	Optional Redistributes BGP routes. No routes are redistributed by default.
BGP Router Mode	
[no] address-family ipv4 {unicast| multicast|vrf <*vrf-nam*>}	Optional Creates the BGP IPv4 VRF address family and enters BGP-VPN instance view. It is not created by default.
BGP-VPN Instance Mode	
[no] redistribute {connected | ospf | rip | static}	Optional Redistributes directly connected routes and routes of other protocols. No routes are redistributed by default.

Use static routes between PE and CE

Configure the remote PE as a public network VPNv4 neighbor

Configure static VPN routes

Enter BGP-VPN instance view

Redistribute static routes

Advertise local private network routes

Command	Explanation
BGP Router Mode	

neighbor <*ip-address*> remote-as <*as-num*>	Required Configures the remote PE as a public network VPNv4 neighbor. It is recommended to use loopback interfaces to establish BGP neighbors between public network PEs.
neighbor <*ip-address*> update-source <*as-num*>	Specifies the local loopback interface used to establish the neighbor.
BGP-VPNv4 Mode	
address-family vpnv4 [unicast]	Required Creates the BGP VPNv4 address family. No VPNv4 address family is created by default.
[no] neighbor <*ip-address*> active	Optional Activates neighbors in VPNv4. All neighbors in VPNv4 view are active by default.
Global Mode	
[no] ip route vrf <vrf-name> {<ip-prefix> <mask>|<ip-prefix/<prefix-length>} {<gateway-address>|null0}	Optional Configures static VPN routes between PE and CE manually.
BGP Router Mode	
[no] address-family ipv4 {unicast| multicast|vrf <*vrf-nam*>}	Optional Creates the BGP IPv4 VRF address family and enters BGP-VPN instance view. It is not created by default.
BGP-VPN Instance Mode	
[no] redistribute {connected|ospf|rip| static}	Optional Redistributes static routes, directly connected routes, and routes of other protocols. No routes are redistributed by default.

13.3.3 BGP MPLS VPN Typical Examples

13.3.3.1 Creating a BGP MPLS VPN with EBGP Between PE and CE

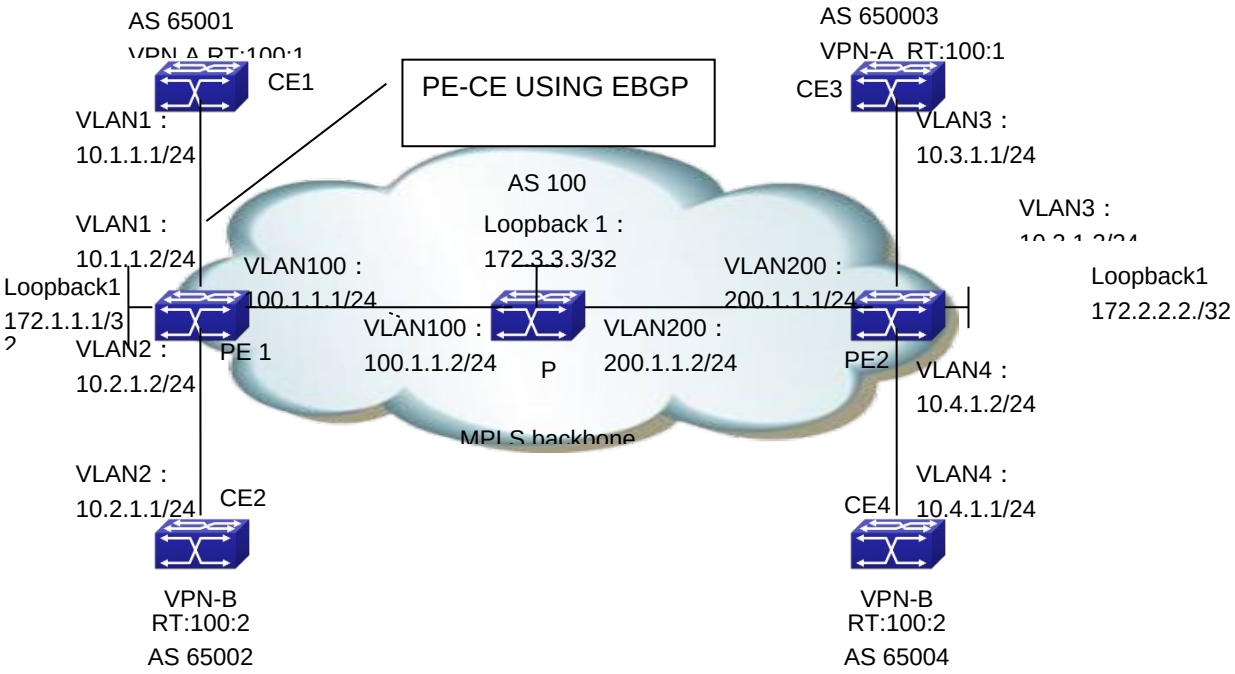


Figure 13-16 Creating a BGP MPLS VPN with EBGP Between PE and CE

CE1 configuration (the configurations of CE2 to CE4 are similar):

```
CE1#config
CE1(config)# interface vlan 1
CE1(config-if-Vlan1)#ip address 10.1.1.1 255.255.255.0
CE1(config-if-Vlan1)#exit
CE1(config)# router bgp 65001
CE1(config-router)#neighbor 10.1.1.2 remote-as 100
CE1(config-router)#redistribute connect
CE1(config-router)#exit
```

MPLS BGP configuration on switch PE1:

Configure VPN instances.

```
PE1#config
PE1(config)#ip vrf vpna
PE1(config-vrf)#rd 100:1
PE1(config-vrf)#route-target both 100:1
PE1(config)#ip vrf vpnb
PE1(config-vrf)#rd 100:2
PE1(config-vrf)#route-target both 100:2
```

Bind the interfaces to the VPN instances.

```
PE1(config)# interface vlan 1
PE1(config-if-Vlan1)# ip vrf forwarding vpna
PE1(config-if-Vlan1)#ip address 10.1.1.2 255.255.255.0
PE1(config-if-Vlan1)#exit
PE1(config)# interface vlan 2
PE1(config-if-Vlan2)# ip vrf forwarding vpnb
PE1(config-if-Vlan2)#ip address 10.2.1.2 255.255.255.0
PE1(config-if-Vlan2)#exit
```

Enable MPLS and LDP globally.

```
PE1(config)#mpls enable
PE1(config)#router ldp
PE1(config-router)#exit
```

Configure the interfaces and enable LDP.

```
PE1(config)# interface loopback 1
PE1(config-if-Loopback1)# ip address 172.1.1.1 255.255.255.255
PE1(config-if-Loopback1)# exit
PE1(config)# interface vlan 100
PE1(config-if-Vlan100)#ip address 100.1.1.1 255.255.255.0
PE1(config-if-Vlan100)#label-switching
PE1(config-if-Vlan100) #ldp enable
PE1(config-if-Vlan100)#exit
```

Enable OSPF to advertise internal network routes.

```
PE1(config)#router ospf
PE1(config-router)# ospf router-id 172.1.1.1
PE1(config-router)# network 0.0.0.0/0 area 0
PE1(config-router)# redistribute connected
```

Configure BGP.

```
PE1(config)# router bgp 100
PE1(config-router)#neighbor 172.2.2.2 remote-as 100
PE1(config-router)#neighbor 172.2.2.2 update-source 172.1.1.1
PE1(config-router)#address-family vpnv4
PE1(config-router-af)#neighbor 172.2.2.2 activate
PE1(config-router-af)#exit
PE1(config-router)# address-family ipv4 vrf vpna
PE1(config-router-af)#neighbor 10.1.1.1 remote-as 65001
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#exit
PE1(config-router)# address-family ipv4 vrf vpnb\
PE1(config-router-af)#neighbor 10.2.1.1 remote-as 65002
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#exit
PE1(config-router)#exit
```

Configuration of router P:

Enable MPLS globally and configure LDP on the related interfaces.

```
P#config
P(config)#mpls enable
P(config)#router ldp
P(config-router)#exit
P(config)# interface loopback 1
P(config-if-Loopback1)# ip address 172.3.3.3 255.255.255.255
P(config-if-Loopback1)# exit
P(config)#interface vlan 100
P(config-if-Vlan100)#ip address 100.1.1.2 255.255.255.0
P(config-if-Vlan100)#label-switching
P(config-if-Vlan100)#ldp enable
P(config-if-Vlan100)#exit
P(config)#interface vlan200
P(config-if-Vlan200)#ip address 200.1.1.2 255.255.255.0
P(config-if-Vlan200)#label-switching
P(config-if-Vlan200)#ldp enable
P(config-if-Vlan200)#exit
```

Configure OSPF.

```
P(config)#router ospf
P(config-router)# ospf router-id 172.3.3.3
P(config-router)# network 0.0.0.0/0 area 0
```

```
P(config-router)# redistribute connected
```

Configuration of switch PE2:

Configure VPN instances.

```
PE2#config
PE2(config)#ip vrf vpna
PE2(config-vrf)#rd 100:1
PE2(config-vrf)#route-target both 100:1
PE2(config)#ip vrf vpnb
PE2(config-vrf)#rd 100:2
PE2(config-vrf)#route-target both 100:2
```

Bind the interfaces to the VPN instances.

```
PE2(config)# interface vlan 3
PE2(config-if-Vlan3)# ip vrf forwarding vpna
PE2(config-if-Vlan3)#ip address 10.3.1.2 255.255.255.0
PE2(config-if-Vlan3)#exit
PE2(config)# interface vlan 4
PE2(config-if-Vlan4)# ip vrf forwarding vpnb
PE2(config-if-Vlan4)#ip address 10.4.1.2 255.255.255.0
PE2(config-if-Vlan4)#exit
```

Enable MPLS and LDP globally.

```
PE2(config)#mpls enable
PE1(config)#router ldp
PE1(config-router)#exit
```

Configure the interfaces and enable LDP.

```
PE2(config)# interface loopback 1
PE2(config-if-Loopback1)# ip address 172.2.2.2 255.255.255.255
PE2(config-if-Loopback1)# exit
PE2(config)# interface vlan 200
PE2(config-if-Vlan200)#ip address 200.1.1.1 255.255.255.0
PE2(config-if-Vlan200)#label-switching
PE2(config-if-Vlan200) #ldp enable
PE2(config-if-Vlan200)#exit
```

Enable OSPF to advertise internal network routes.

```
PE2(config)#router ospf
PE2(config-router)# ospf router-id 172.2.2.2
PE2(config-router)# network 0.0.0.0/0 area 0
PE2(config-router)# redistribute connected
```

Configure BGP.

```
PE2(config)# router bgp 100
PE2(config-router)#neighbor 172.1.1.1 remote-as 100
PE2(config-router)#neighbor 172.1.1.1 update-source 172.2.2.2
PE2(config-router)#address-family vpnv4
PE2(config-router-af)#neighbor 172.1.1.1 activate
PE2(config-router-af)#exit
PE2(config-router)# address-family ipv4 vrf vpna
PE2(config-router-af)#neighbor 10.3.1.1 remote-as 65003
PE2(config-router-af)#redistribute connected
PE2(config-router-af)#exit
PE2(config-router)# address-family ipv4 vrf vpnb
PE2(config-router-af)#neighbor 10.4.1.1 remote-as 65004
PE2(config-router-af)#redistribute connected
PE2(config-router-af)#exit
PE2(config-router)#exit
```

13.3.3.2 Creating a BGP MPLS VPN with OSPF Between PE and CE

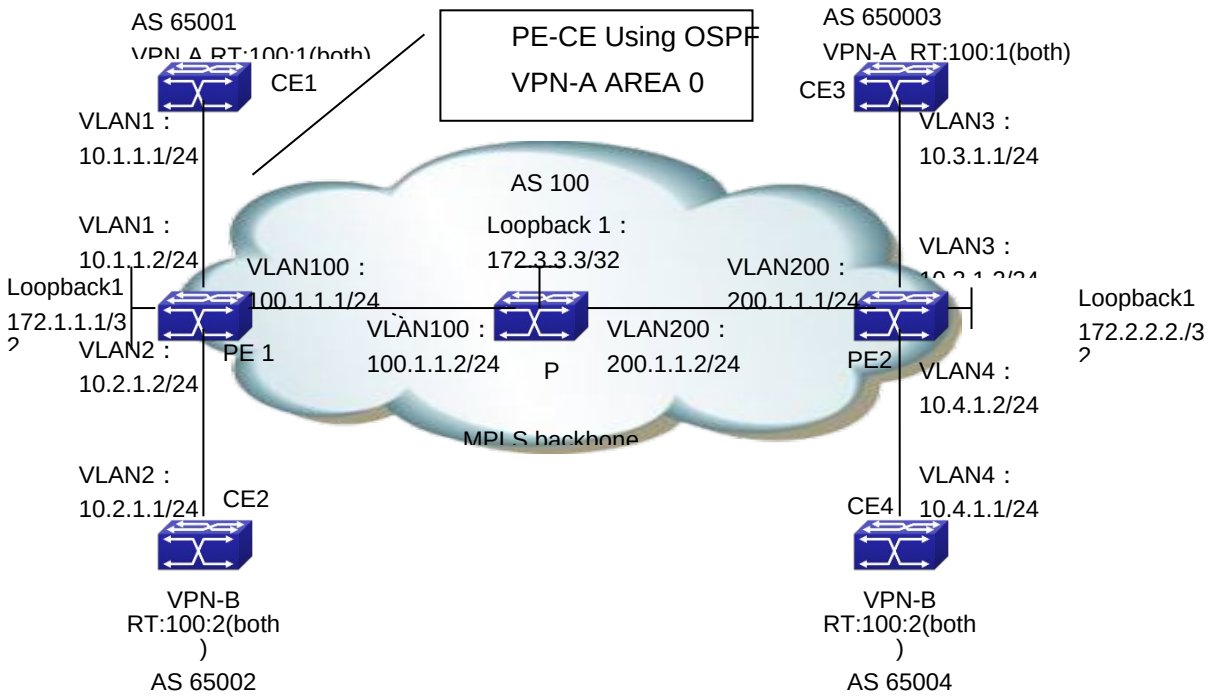


Figure 13-17 Creating a BGP MPLS VPN with OSPF Between PE and CE

CE1 configuration (the configurations of CE2 to CE4 are similar):

```
CE1#config
CE1(config)# interface vlan 1
CE1(config-if-Vlan1)#ip address 10.1.1.1 255.255.255.0
CE1(config-if-Vlan1)#exit
CE1(config)# router ospf
CE1(config-router)#network 0.0.0.0/0 area 0
CE1(config-router)#redistribute connect
CE1(config-router)#exit
```

MPLS BGP configuration on switch PE1 (the configuration of PE2 is similar):

Configure VPN instances.

```
PE1#config
PE1(config)#ip vrf vpna
PE1(config-vrf)#rd 100:1
PE1(config-vrf)#route-target both 100:1
PE1(config)#ip vrf vpnb
PE1(config-vrf)#rd 100:2
PE1(config-vrf)#route-target both 100:2
```

Bind the interfaces to the VPN instances.

```
PE1(config)# interface vlan 1
PE1(config-if-Vlan1)# ip vrf forwarding vpna
PE1(config-if-Vlan1)#ip address 10.1.1.2 255.255.255.0
PE1(config-if-Vlan1)#exit
PE1(config)# interface vlan 2
PE1(config-if-Vlan2)# ip vrf forwarding vpnb
PE1(config-if-Vlan1)#ip address 10.2.1.2 255.255.255.0
PE1(config-if-Vlan1)#exit
```

Enable MPLS and LDP globally.

```
PE1(config)#mpls enable
```

```
PE1(config)#router ldp
PE1(config-router)#exit
```

Configure the interfaces and enable LDP.

```
PE1(config)# interface loopback 1
PE1(config-if-Loopback1)# ip address 172.1.1.1 255.255.255.255
PE1(config-if-Loopback1)# exit
PE1(config)# interface vlan 100
PE1(config-if-Vlan100)#ip address 100.1.1.1 255.255.255.0
PE1(config-if-Vlan100)# label-switching
PE1(config-if-Vlan100) #ldp enable
PE1(config-if-Vlan100)#exit
```

Enable OSPF to advertise internal network routes.

```
PE1(config)#router ospf
PE1(config-router)# ospf router-id 172.1.1.1
PE1(config-router)# network 0.0.0.0/0 area 0
PE1(config-router)# redistribute connected
PE1(config-router)#exit
```

Enable OSPF in the VRFs to advertise private network routes.

```
PE1(config)#router ospf 1 vpna
PE1(config-router)# network 0.0.0.0/0 area 0
PE1(config-router)#redistribute connected
PE1(config-router)#redistribute bgp
PE1(config-router)#exit
PE1(config)#router ospf 1 vpnb
PE1(config-router)# network 0.0.0.0/0 area 0
PE1(config-router)#redistribute connected
PE1(config-router)#redistribute bgp
PE1(config-router)#exit
```

Configure BGP.

```
PE1(config)# router bgp 100
PE1(config-router)#neighbor 172.2.2.2 remote-as 100
PE1(config-router)#neighbor 172.2.2.2 update-source 172.1.1.1
PE1(config-router)#address-family vpnv4
PE1(config-router-af)#neighbor 172.2.2.2 activate
PE1(config-router-af)#exit
PE1(config-router)# address-family ipv4 vrf vpna
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute ospf
PE1(config-router-af)#exit
PE1(config-router)# address-family ipv4 vrf vpnb
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute ospf
PE1(config-router-af)#exit
PE1(config-router)#exit
```

Configuration of router P:

Enable MPLS globally and configure LDP on the related interfaces.

```
P#config
P(config)#mpls enable
P(config)#router ldp
P(config-router)#exit
P(config)# interface loopback 1
P(config-if-Loopback1)# ip address 172.3.3.3 255.255.255.255
P(config-if-Loopback1)# exit
P(config)#interface vlan 100
P(config-if-Vlan100)#ip address 100.1.1.2 255.255.255.0
P(config-if-Vlan100)#label-switching
P(config-if-Vlan100)#ldp enable
```

```
P(config-if-Vlan100)#exit
P(config)#interface vlan200
P(config-if-Vlan200)#ip address 200.1.1.2 255.255.255.0
P(config-if-Vlan200)#label-switching
P(config-if-Vlan200)#ldp enable
P(config-if-Vlan200)#exit
```

Configure OSPF.

```
P(config)#router ospf
P(config-router)# ospf router-id 172.3.3.3
P(config-router)# network 0.0.0.0/0 area 0
P(config-router)# redirect connected
```

13.3.3.3 Creating a BGP MPLS VPN with RIP Between PE and CE

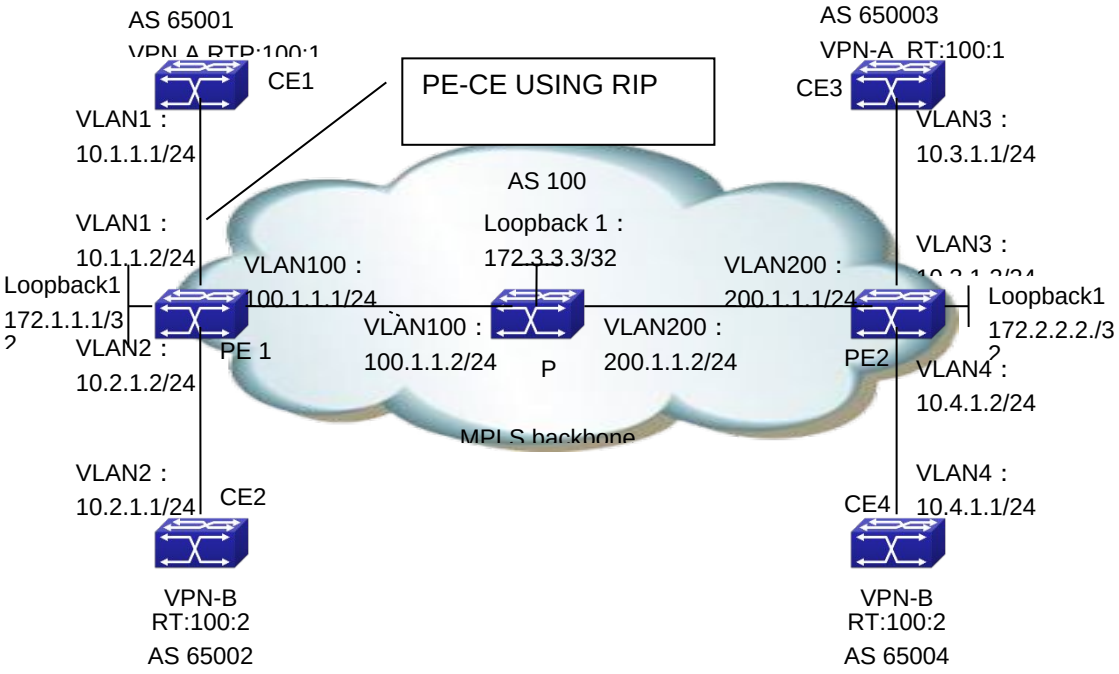


Figure 13-18 Creating a BGP MPLS VPN with RIP Between PE and CE

CE1 configuration (the configurations of CE2 to CE4 are similar):

```
CE1#config
CE1(config)# interface vlan 1
CE1(config-if-Vlan1)#ip address 10.1.1.1 255.255.255.0
CE1(config-if-Vlan1)#exit
CE1(config)# router rip
CE1(config-router)#network 0.0.0.0/0
CE1(config-router)#redistribute connect
CE1(config-router)#exit
```

MPLS BGP configuration on switch PE1 (the configuration of PE2 is similar):

Configure VPN instances.

```
PE1#config
PE1(config)#ip vrf vpna
PE1(config-vrf)#rd 100:1
PE1(config-vrf)#route-target both 100:1
PE1(config)#ip vrf vpnb
PE1(config-vrf)#rd 100:2
PE1(config-vrf)#route-target both 100:2
```

Bind the interfaces to the VPN instances.

```
PE1(config)# interface vlan 1
```

```
PE1(config-if-Vlan1)# ip vrf forwarding vpna
PE1(config-if-Vlan1)#ip address 10.1.1.2 255.255.255.0
PE1(config-if-Vlan1)#exit
PE1(config)# interface vlan 2
PE1(config-if-Vlan2)# ip vrf forwarding vpnb
PE1(config-if-Vlan1)#ip address 10.2.1.2 255.255.255.0
PE1(config-if-Vlan1)#exit
```

Enable MPLS and LDP globally.

```
PE1(config)#mpls enable
PE1(config)#router ldp
PE1(config-router)#exit
```

Configure the interfaces and enable LDP.

```
PE1(config)# interface loopback 1
PE1(config-if-Loopback1)# ip address 172.1.1.1 255.255.255.255
PE1(config-if-Loopback1)# exit
PE1(config)# interface vlan 100
PE1(config-if-Vlan100)#ip address 100.1.1.1 255.255.255.0
PE1(config-if-Vlan100)#label-switching
PE1(config-if-Vlan100) #ldp enable
PE1(config-if-Vlan100)#exit
```

Enable OSPF to advertise internal network routes.

```
PE1(config)#router ospf
PE1(config-router)# ospf router-id 172.1.1.1
PE1(config-router)# network 0.0.0.0/0 area 0
PE1(config-router)# redistribute connected
PE1(config-router)#exit
```

Enable RIP in the VRFs to advertise private network routes.

```
PE1(config)#router rip
PE1(config-router)#address-family ipv4 vrf vpna
PE1(config-router-af)#network 0.0.0.0/0
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute bgp
PE1(config-router-af)#exit
PE1(config-router)#address-family ipv4 vrf vpnb
PE1(config-router-af)#network 0.0.0.0/0
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute bgp
PE1(config-router-af)#exit
PE1(config-router)#exit
```

Configure BGP.

```
PE1(config)# router bgp 100
PE1(config-router)#neighbor 172.2.2.2 remote-as 100
PE1(config-router)#neighbor 172.2.2.2 update-source 172.1.1.1
PE1(config-router)#address-family vpnv4
PE1(config-router-af)#neighbor 172.2.2.2 activate
PE1(config-router-af)#exit
PE1(config-router)# address-family ipv4 vrf vpna
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute ospf
PE1(config-router-af)#exit
PE1(config-router)# address-family ipv4 vrf vpnb
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute ospf
PE1(config-router-af)#exit
PE1(config-router)#exit
```

Configuration of switch P:

Enable MPLS globally and configure LDP on the related interfaces.

```
P#config
P(config)#mpls enable
P(config)#router ldp
P(config-router)#exit
P(config)# interface loopback 1
P(config-if-Loopback1)# ip address 172.3.3.3 255.255.255.255
P(config-if-Loopback1)# exit
P(config)#interface vlan 100
P(config-if-Vlan100)#ip address 100.1.1.2 255.255.255.0
P(config-if-Vlan100)#label-switching
P(config-if-Vlan100)#ldp enable
P(config-if-Vlan100)#exit
P(config)#interface vlan200
P(config-if-Vlan200)#ip address 200.1.1.2 255.255.255.0
P(config-if-Vlan200)#label-switching
P(config-if-Vlan200)#ldp enable
P(config-if-Vlan200)#exit
```

Configure OSPF.

```
P(config)#router ospf
P(config-router)# ospf router-id 172.3.3.3
P(config-router)# network 0.0.0.0/0 area 0
P(config-router)# redistribute connected
```

13.3.3.4 Creating a BGP MPLS VPN with Static Routes Between PE and CE

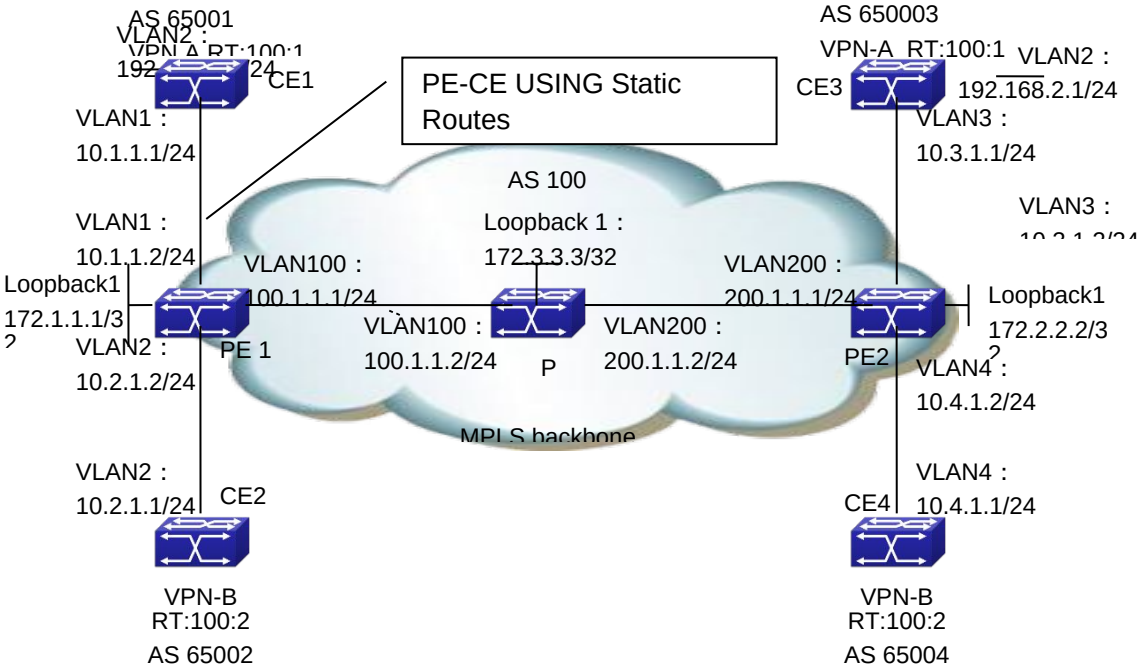


Figure 13-19 Creating a BGP MPLS VPN with Static Routes Between PE and CE

CE1 configuration (the configurations of CE2 to CE4 are similar):

```
CE1#config
CE1(config)# interface vlan 1
CE1(config-if-Vlan1)#ip address 10.1.1.1 255.255.255.0
CE1(config-if-Vlan1)#exit
CE1(config)# interface loopback 1
CE1(config-if-Vlan1)#ip address 192.168.1.1 255.255.255.0
CE1(config-if-Vlan1)# exit
CE1(config)# ip route vrf vpna 192.168.2.1/24 10.1.1.2
```

MPLS BGP configuration on switch PE1 (the configuration of PE2 is similar):

Configure VPN instances.

```
PE1#config
PE1(config)#ip vrf vpna
PE1(config-vrf)#rd 100:1
PE1(config-vrf)#route-target both 100:1
PE1(config)#ip vrf vpnb
PE1(config-vrf)#rd 100:2
PE1(config-vrf)#route-target both 100:2
```

Bind the interfaces to the VPN instances.

```
PE1(config)# interface vlan 1
PE1(config-if-Vlan1)# ip vrf forwarding vpna
PE1(config-if-Vlan1)#ip address 10.1.1.2 255.255.255.0
PE1(config-if-Vlan1)#exit
PE1(config)# interface vlan 2
PE1(config-if-Vlan2)# ip vrf forwarding vpnb
PE1(config-if-Vlan1)#ip address 10.2.1.2 255.255.255.0
PE1(config-if-Vlan1)#exit
```

Enable MPLS and LDP globally.

```
PE1(config)#mpls enable
PE1(config)#router ldp
PE1(config-router)#exit
```

Configure the interfaces and enable LDP.

```
PE1(config)# interface loopback 1
PE1(config-if-Loopback1)# ip address 172.1.1.1 255.255.255.255
PE1(config-if-Loopback1)# exit
PE1(config)# interface vlan 100
PE1(config-if-Vlan100)#ip address 100.1.1.1 255.255.255.0
PE1(config-if-Vlan100) #ldp enable
PE1(config-if-Vlan100)#exit
```

Enable OSPF to advertise internal network routes.

```
PE1(config)#router ospf
PE1(config-router)# ospf router-id 172.1.1.1
PE1(config-router)# network 0.0.0.0/0 area 0
PE1(config-router)# redistribute connected
PE1(config-router)#exit
```

Configure static private network routes.

```
PE1(config)# ip route vrf vpna 192.168.1.1/24 10.1.1.2
PE1(config)# ip route vrf vpnb 192.168.2.1/24 10.1.1.2
PE1(config-router)#address-family ipv4 vrf vpna
PE1(config-router-af)#network 0.0.0.0/0
PE1(config-router-af)#redistribute connected
PE1(config-router)#exit
```

Configure BGP.

```
PE1(config)# router bgp 100
PE1(config-router)#neighbor 172.2.2.2 remote-as 100
PE1(config-router)#neighbor 172.2.2.2 update-source 172.1.1.1
PE1(config-router)#address-family vpnv4
PE1(config-router-af)#neighbor 172.2.2.2 activate
PE1(config-router-af)#exit
PE1(config-router)# address-family ipv4 vrf vpna
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute static
PE1(config-router-af)#exit
PE1(config-router)# address-family ipv4 vrf vpnb
PE1(config-router-af)#redistribute connected
PE1(config-router-af)# redistribute static
```

```
PE1(config-router-af)#exit
PE1(config-router)#exit
```

Configuration of switch P:

Enable MPLS globally and configure LDP on the related interfaces.

```
P#config
P(config)#mpls enable
P(config)#router ldp
P(config-router)#exit
P(config)# interface loopback 1
P(config-if-Loopback1)# ip address 172.3.3.3 255.255.255.255
P(config-if-Loopback1)# exit
P(config)#interface vlan 100
P(config-if-Vlan100)#ip address 100.1.1.2 255.255.255.0
P(config-if-Vlan100)#ldp enable
P(config-if-Vlan100)#exit
P(config)#interface vlan200
P(config-if-Vlan200)#ip address 200.1.1.2 255.255.255.0
P(config-if-Vlan100)#ldp enable
P(config-if-Vlan200)#exit
```

Configure OSPF.

```
P(config)#router ospf
P(config-router)# ospf router-id 172.3.3.3
P(config-router)# network 0.0.0.0/0 area 0
P(config-router)# redistribute connected
```

13.3.4 MPLS BGP VPN Troubleshooting

When MPLS BGP VPN is configured and used, problems such as incorrect physical connections and configuration errors may cause it to fail. To avoid them, note the following:

- First, make sure that OSPF neighbors are correctly established among PE1, P, and PE2, that routes (including the loopback interface routes) are correctly advertised, and that the BGP neighbor relationship between the PEs is correctly established.
- Second, make sure that LDP is enabled globally on PE1, P, and PE2 and correctly enabled on the active interfaces. Check whether LDP sessions are correctly established on PE1, P, and PE2.
- Next, make sure that the PE-CE route advertisement method used for the VPN and its configuration are correct. Check whether the CE advertises the related private network routes to the remote PE. Note that a CE does not need a VRF instance. If EBGP is used to advertise private network routes, CE1 and CE2 must use different BGP ASNs; otherwise, BGP loop detection filters the corresponding private network routes.
- Then, make sure that the BGP VPN instances on the PEs are correctly configured. When OSPF or RIP is used to create and advertise PE-CE routes, redistribute BGP routes into OSPF or RIP, and redistribute the corresponding OSPF or RIP routes into the BGP VPN instances. If the configuration is correct, running "show ip bgp vpnv4 all" on PE1 displays the routing information of CE1 and CE2, and running "show mpls vrf-table" on a PE shows that labels are assigned to the corresponding private network routes and that their state is UP. If the Oper status of the corresponding private network routes in the vrf-table is not UP, use "show mpls ftn-table" to check whether an FTN entry is created for the corresponding FEC.
- Finally, if all the above steps are correct, use "show ip route" on CE1 and CE2 to check the routing information in the VPN. Creating a VPN with static routes is not recommended unless you are very familiar with BGP MPLS VPN.

- In addition, if a CE cannot see any remote CE device after the correct configuration is saved and the device is rebooted, wait patiently: establishing OSPF, LDP, and BGP connections and advertising routes takes time.

13.4 Public Network Access of MPLS VPN

13.4.1 Public Network Access Introduction

Public network access of a VPN is the ability of VPN sites to access the public Internet. RFC 4364 defines the basic protocol rules, including the following methods for a VPN to access the Internet:

- Non-VRF Internet Access Mode
- VRF Internet Access Mode 1
- VRF Internet Access Mode 3

13.4.1.1 Non-VRF Internet Access Mode

As shown in the next figure, in Non-VRF Internet Access Mode, PE routers communicate with Internet gateways through non-VRF interfaces, and the Internet access traffic of VPN sites is forwarded according to the global route table of the PE routers. A CE router that can access the Internet has two connections to the PE: one to the public network interface of the PE (the public network connection) and one to the private network interface of the PE (the private network connection). The global route table of a PE router can contain all or part of the Internet routes, or only a default route pointing to the Internet gateway. CE routers learn Internet routes through the public network connection. Through the same connection, they advertise to the PE the routes of the globally registered IP subnets in the VPN site; the PE advertises these routes to the Internet gateway, which advertises them to the Internet. The Internet access traffic of VPN sites is also sent and received through the public network connection. The private network connection between the CE and the PE is used by the CE to learn routes and to advertise the private network routes in the VPN. VPN sites also communicate with each other through private network connections, and this traffic is forwarded according to the VRF route table of the PE routers. In this mode, the global route table and the VRF route table of the PE routers are completely isolated, and VPN routes and Internet routes are distributed completely independently.

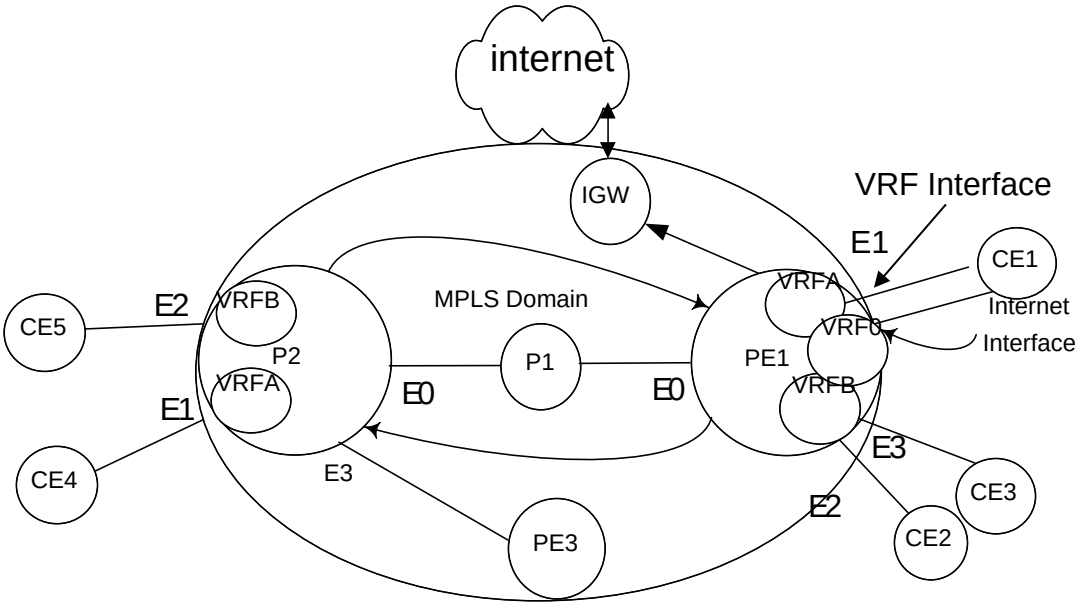


Fig 10-20 Non-VRF Internet Access Mode

13.4.1.2 VRF Internet Access Mode 1

As shown in the next figure, in VRF Internet Access Mode 1, PE routers communicate with Internet gateways through non-VRF interfaces. Both the Internet access traffic of VPN sites and the traffic between VPN sites are sent and received through the private network connections between the CEs and the PEs. The PE routers contain all or part of the Internet routes, or only a default route pointing to the Internet gateway. When an IP packet from a VPN to the Internet reaches a VRF interface of a PE, a failed lookup in the VRF route table triggers a lookup in the global route table. If a match is found, the packet is forwarded to the Internet gateway and then to the Internet through the gateway. To allow Internet hosts to access VPN sites, a special static route must be configured in the global route table of the PE. Its destination is the globally registered IP subnet of the VPN site, its egress interface is the private network interface pointing to the VPN site, and its next hop is the CE router. The PE advertises this static route to the Internet gateway, which advertises it to the Internet. When a packet from the Internet to the VPN reaches the public network interface of the PE and matches this static route in the global route table of the PE, it is forwarded to the next hop through the private network interface. In this mode, the global route table and the VRF route table of the PE routers are not completely isolated, because the global route table contains some VPN routes.

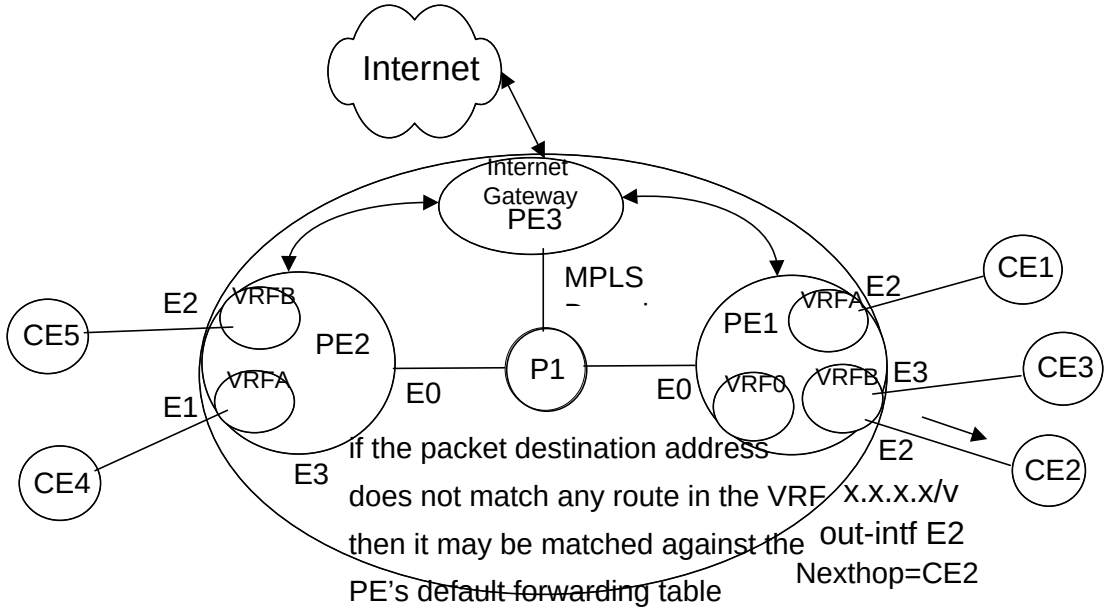


Fig 10-21 VRF Internet Access Mode 1

13.4.1.3 VRF Internet Access Mode 3

As shown in the next figure, in VRF Internet Access Mode 3, VPN sites access the Internet through the private network connections between the PEs and the CEs. The VRF route tables of the PE routers contain Internet routes, which are learned through the PE router connected to the Internet gateway (the Internet PE). The Internet PE creates an Internet VRF and connects to the Internet gateway through an interface bound to the Internet VRF. The Internet gateway can then advertise Internet routes to the VRF route table of the Internet PE. These routes are then advertised to the VRFs of the other PE routers as VPNv4 routes. The PE routers connected to VPN sites that need Internet access advertise the corresponding VRF routes (only routes whose destinations are globally registered IP subnets in the VPN) to the Internet PE as VPNv4 routes. These routes are added to the Internet VRF and then advertised to the Internet by the Internet gateway. The import and export policies of these routes depend on the route-target configuration of MBGP and the VRF. Note that in this mode, addresses and routes must not overlap between VPN sites that can access the Internet.

This mode is not recommended for public network access, because a large number of Internet routes are imported into the PEs.

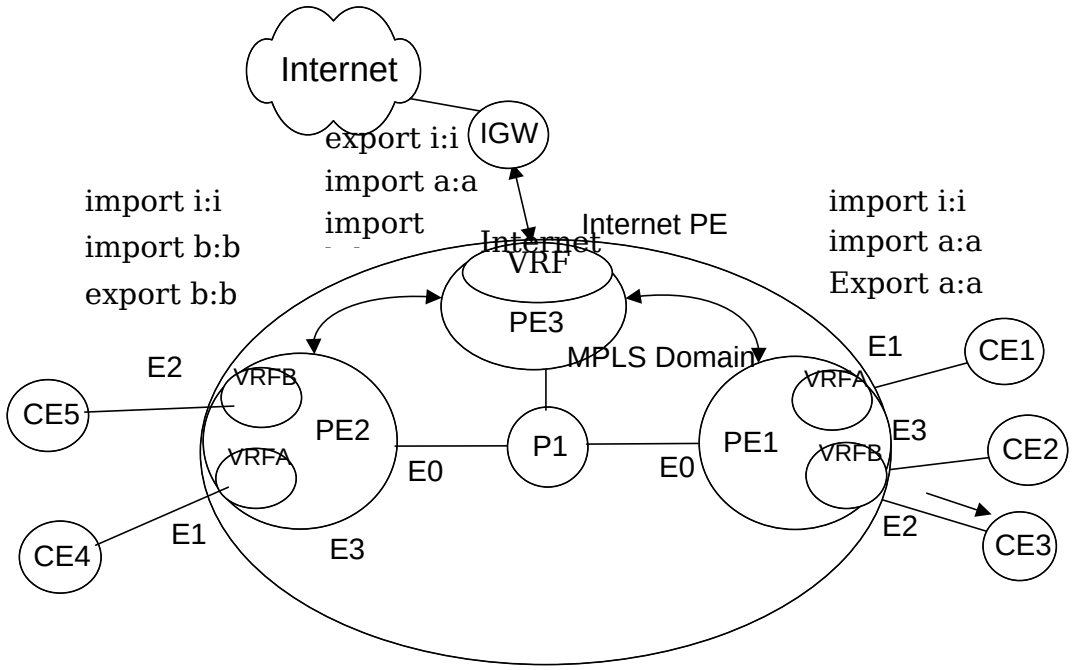


Fig 10_x001E_22 VRF Internet Access Mode 3

13.4.2 Public Network Access Configuration

Public network access configuration task sequence:

Configure Non-VRF Internet Access Mode

Configure regular L3VPN

Add a public network connection between the CE and the PE; the connecting interface is a non-VRF interface.

Filter routes on the CE, and advertise public network routes to the PE through the public network connection.

Configure a filter policy on the public network interface to filter packets whose source and destination addresses are private network addresses.

Configure default routes

The IGW imports the default route into BGP.

The PE advertises the default route to the CE through the public network connection.

The CE advertises the default route to the PE through the private network connection, and the PE advertises it to the other CEs.

Configure static routes

Configure a static route pointing to the Internet on CE1.

Configure a static route pointing to the public network interface of the CE on PE1.

Configure non-VRF Internet Access Mode

This configuration requires no commands other than those in the configuration sequence. For details about the commands, refer to the configuration instructions of the corresponding functions.

13.4.3 Public Network Access Typical Examples

13.4.3.1 Non-VRF Internet Access Mode

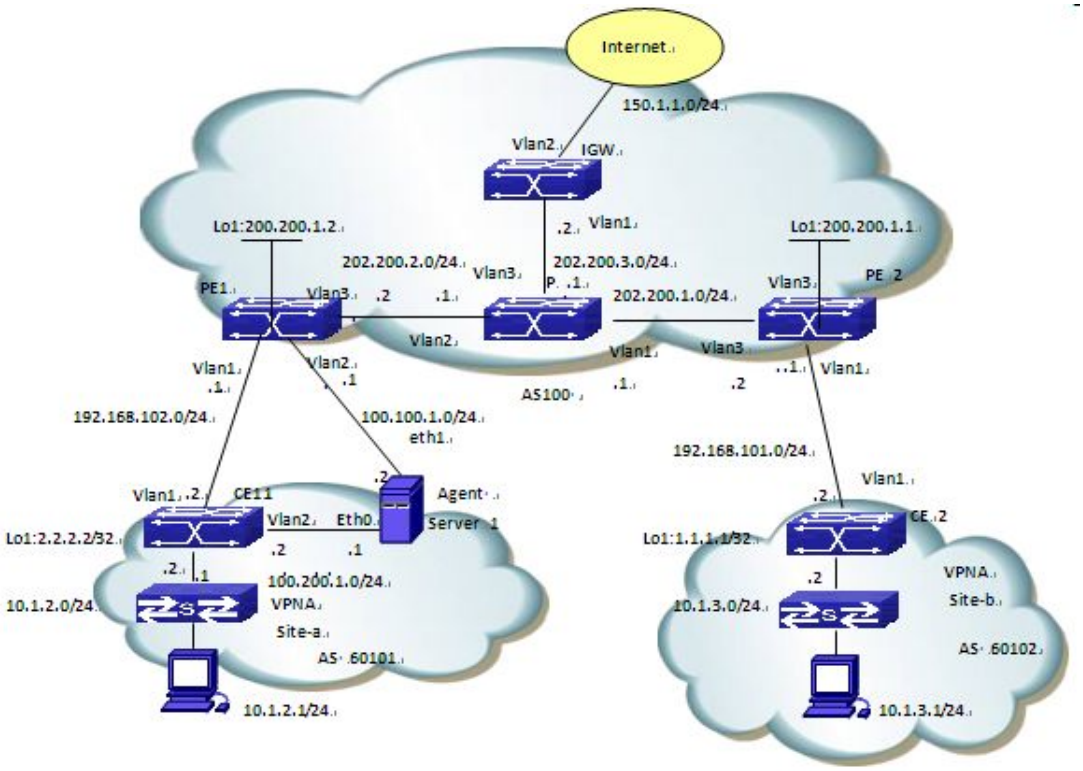


Fig 10-23 Non-VRF Internet Access Mode

The configuration of CE1 is as follows:

```
CE1#config
CE1(config)#access-list 1 deny 100.100.1.0 0.0.0.255
CE1(config)#access-list 1 deny 100.200.1.0 0.0.0.255
CE1(config)#access-list 1 permit any-source
CE1(config)#access-list 2 permit 10.1.1.0 0.0.0.255
CE1(config)#access-list 2 permit 10.1.2.0 0.0.0.255
CE1(config)#access-list 2 deny any-source
CE1(config)# interface vlan 1
CE1(config-if-Vlan1)#ip address 192.168.102.2 255.255.255.0
CE1(config-if-Vlan1)#exit
CE1(config)# interface vlan 2
CE1(config-if-Vlan1)#ip address 100.200.1.2 255.255.255.0
CE1(config-if-Vlan1)#exit
CE1(config)# interface vlan 3
CE1(config-if-Vlan1)#ip address 10.1.2.2 255.255.255.0
CE1(config-if-Vlan1)#exit
CE1(config)# interface loopback 1
CE1(config-if-Vlan1)#ip address 2.2.2.2 255.255.255.255
CE1(config-if-Vlan1)# exit
CE1(config)#router bgp 60102
CE1(config-router)#network 120.1.1.0/24
CE1(config-router)#network 120.1.2.0/24
CE1(config-router)#network 10.1.2.0/24
CE1(config-router)#redistribute connected
CE1(config-router)#neighbor 100.100.1.1 remote-as 100
CE1(config-router)#neighbor 100.100.1.1 distribute-list 2 out
CE1(config-router)#neighbor 192.168.102.1 remote-as 100
CE1(config-router)#neighbor 192.168.102.1 default-originate
CE1(config-router)#neighbor 192.168.102.1 distribute-list 1 out
```

```
CE1(config-router)#exit
CE1(config)# ip route 100.100.1.1 255.255.255.0 100.200.1.1
CE1(config)# ip route 0.0.0.0/0 100.200.1.1
CE1(config)# exit
```

The configuration of PE1 is as follows:

```
PE1#config
PE1(config)#access-list 100 deny ip 10.1.2.0 0.0.0.255 any-destination
PE1(config)#access-list 100 deny ip 10.1.2.0 0.0.0.255 any-destination
PE1(config)#access-list 100 deny ip 10.1.3.0 0.0.0.255 any-destination
PE1(config)#access-list 100 deny ip anysource 200.200.1.0 0.0.0.255
PE1(config)#access-list 100 deny ip anysource 202.200.0.0 0.0.255.255
PE1(config)#firewall enable
PE1(config-vrf)#ip vrf VRF-A
PE1(config-vrf)#rd 100:10
PE1(config-vrf)#route-target both 100:10
PE1(config-vrf)#exit
PE1(config)#interface vlan1
PE1(config-if-Vlan1)#ip vrf forwarding VRF-A
PE1(config-if-Vlan1)#ip address 192.168.102.1 255.255.255.0
PE1(config-if-Vlan1)#exit
PE1(config)#interface vlan2
PE1(config-if-Vlan2)#ip address 100.100.1.1 255.255.255.0
PE1(config-if-Vlan2)#ip access-group 1 in
PE1(config-if-Vlan2)#exit
PE1(config)# interface vlan3
PE1(config-if-Vlan3)#label-switching
PE1(config-if-Vlan3)#ldp enable
PE1(config-if-Vlan3)#ip address 202.200.2.2 255.255.255.0
PE1(config-if-Vlan3)#exit
PE1(config)#interface Loopback1
PE1(config)#ip address 200.200.1.2 255.255.255.255
PE1(config)#router ospf
PE1(config-router)#network 200.200.1.2/32 area 0
PE1(config-router)#network 202.200.2.0/24 area 0
PE1(config-router)#exit
PE1(config)#router bgp 100
PE1(config-router)#neighbor 100.200.1.2 remote-as 60102
PE1(config-router)#neighbor 200.200.1.1 remote-as 100
PE1(config-router)#neighbor 202.200.3.2 remote-as 100
PE1(config-router)#neighbor 202.200.3.2 next-hop-self
PE1(config-router)#address-family vpnv4 unicast
PE1(config-router-af)#neighbor 200.200.1.1 activate
PE1(config-router-af)#exit-address-family
PE1(config-router)#address-family ipv4 vrf VRF-A
PE1(config-router-af)#neighbor 192.168.102.2 remote-as 60102
PE1(config-router-af)#no neighbor 192.168.102.2 send-community extended
PE1(config-router-af)#exit-address-family
PE1(config-router)#exit
PE1(config)# router ldp
PE1(config-router)#ip route 100.200.1.2 255.255.255.0 100.100.1.2
```

The configuration of P is as follows:

```
P#config
P(config)#interface Vlan1
P(config-if-Vlan1)#label-switching
P(config-if-Vlan1)#ldp enable
P(config-if-Vlan1)#ip address 202.200.1.1 255.255.255.0
P(config-if-Vlan1)#exit
P(config)#interface Vlan2
P(config-if-Vlan2)#label-switching
P(config-if-Vlan2)#ldp enable
P(config-if-Vlan2)#ip address 202.200.2.1 255.255.255.0
```

```
P(config-if-Vlan2)#exit
P(config)#interface Vlan3
P(config-if-Vlan3)#ip address 202.200.3.1 255.255.255.0
P(config-if-Vlan3)#exit
P(config)#router ospf
P(config-router)#network 202.200.1.0/24 area 0
P(config-router)#network 202.200.2.0/24 area 0
P(config-router)#network 202.200.3.0/24 area 0
P(config-router)#exit
P(config)#router ldp
```

The configuration of PE2 is as follows:

```
PE2#config
PE2(config)#ip vrf VRF-A
PE2(config-vrf)#rd 100:10
PE2(config-vrf)#route-target both 100:10
PE2(config-vrf)#exit
PE2(config)#interface Vlan1
PE2(config-if-Vlan1)#ip vrf forwarding VRF-A
PE2(config-if-Vlan1)#ip address 192.168.101.1 255.255.255.0
PE2(config-if-Vlan1)#exit
PE2(config)#interface Vlan2
PE2(config-if-Vlan2)#label-switching
PE2(config-if-Vlan2)#ldp enable
PE2(config-if-Vlan2)#ip address 202.200.1.2 255.255.255.0
PE2(config-if-Vlan2)#exit
PE2(config)#interface Loopback1
PE2(config-if-loopback1)#ip address 200.200.1.1 255.255.255.255
PE2(config-if-loopback1)#exit
PE2(config)#router ospf
PE2(config-router)#network 200.200.1.1/32 area 0
PE2(config-router)#network 202.200.1.0/24 area 0
PE2(config-router)#exit
PE2(config)#router bgp 100
PE2(config-router)#address-family vpnv4 unicast
PE2(config-router-af)#neighbor 200.200.1.1 activate
PE2(config-router-af)#exit-address-family
PE2(config-router)#address-family ipv4 vrf VRF-A
PE2(config-router-af)#neighbor 192.168.101.2 remote-as 60101
PE2(config-router-af)#no neighbor 192.168.101.2 send-community extended
PE2(config-router-af)#exit-address-family
PE2(config-router)#exit
PE2(config)#router ldp
```

The configuration of CE2 is as follows:

```
CE2#config
CE2(config)#interface vlan 1
CE2(config-if-Vlan1)#ip address 192.168.101.2 255.255.255.0
CE2(config-if-Vlan1)#exit
CE2(config)#interface Loopback1
CE2(config-if-Loopback1)#ip address 1.1.1.1 255.255.255.255
CE2(config-if-Loopback1)#exit
CE2(config)#router bgp 60101
CE2(config-router)#network 10.1.3.0/24
CE2(config-router)#neighbor 192.168.101.1 remote-as 100
```

The configuration of IGW is as follows:

```
IGW#config
IGW(config)#interface Vlan1
IGW(config-if-Vlan1)#ip address 202.200.3.2 255.255.255.0
IGW(config-if-Vlan1)#exit
IGW(config)#interface Vlan2
IGW(config-if-Vlan2#ip address 150.1.1.1 255.255.255.0
```

```
IGW(config-if-Vlan2#exit
IGW(config)#router ospf
IGW(config-router)#network 202.200.3.0 0.0.0.255 area 0
IGW(config-router)#exit
IGW(config)#router bgp 100
IGW(config-router)#neighbor 202.200.2.2 remote-as 100
IGW(config-router)#neighbor 202.200.2.2 default-originate
```

13.4.3.2 VRF Internet Access Mode 1

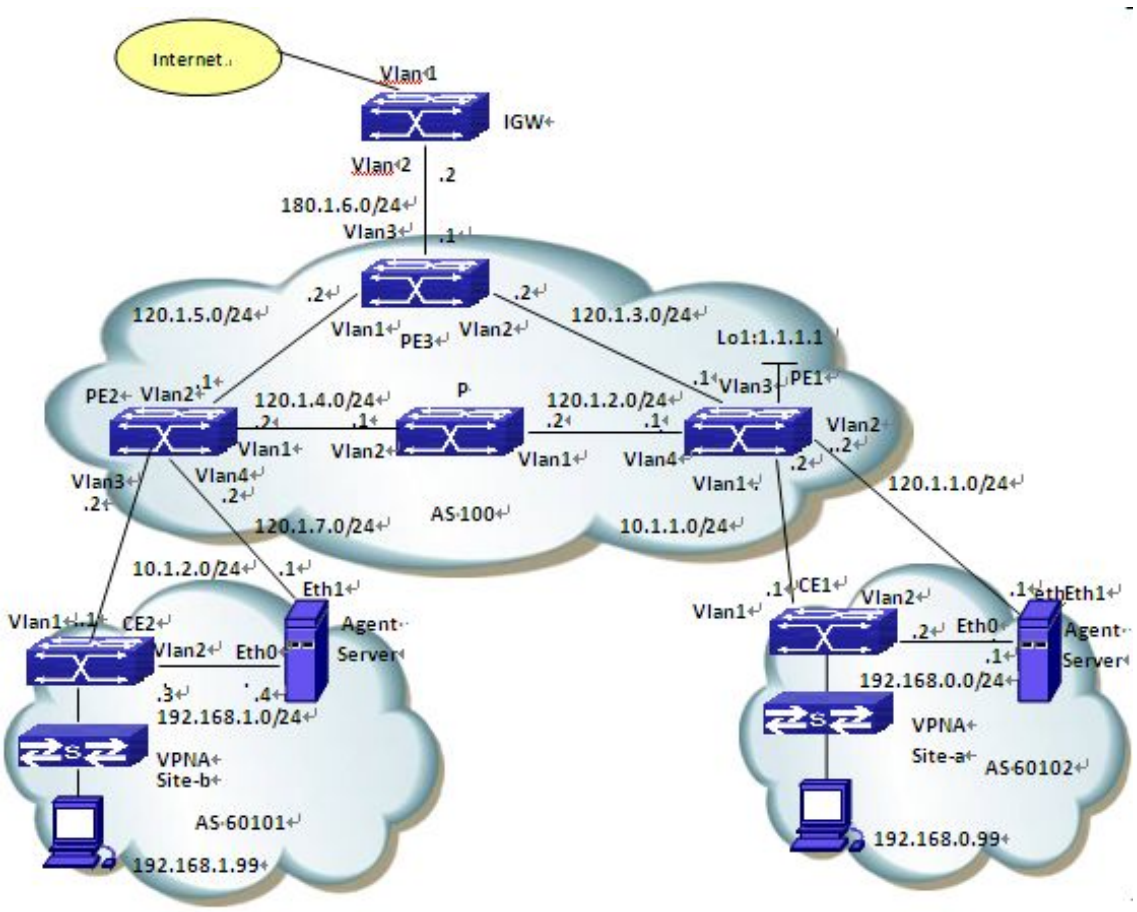


Fig 10-24 VRF Internet Access Mode 1

Site-a and site-b belong to VPNA. Their users can communicate with each other, and all of them need to access the Internet. Configure a proxy server in site-a and in site-b to perform NAT when users access the Internet with private network addresses.

The configuration of CE1 is as follows:

```
CE1#config
CE1(config)#interface Vlan1
CE1(config-if-Vlan1)#ip address 10.1.1.1 255.255.255.0
CE1(config-if-Vlan1)#exit
CE1(config)#interface Vlan2
CE1(config-if-Vlan2)#ip address 192.168.0.2 255.255.255.0
CE1(config-if-Vlan2)#exit
CE1(config)#interface loopback1
CE1(config-if-Loopback1)#ip address 11.11.11.11 255.255.255.255
CE1(config-if-Loopback1)#exit
CE1(config)#ip route 0.0.0.0/0 192.168.0.1
CE1(config)#router bgp 60101
CE1(config-router)#neighbor 10.1.1.2 remote-as 100
CE1(config-router)#network 192.168.0.0/24
```

The configuration of PE1 is as follows:

```
PE1#config
PE1(config)#ip vrf VPNA
PE1(config-vrf)#rd 100:10
PE1(config-vrf)#route-target both 100:10
PE1(config-vrf)#exit
PE1(config)#interface Vlan1
PE1(config-if-Vlan1)#ip vrf forwarding VPNA
PE1(config-if-Vlan1)#ip address 10.1.1.1 255.255.255.0
PE1(config-if-Vlan1)#exit
PE1(config)#interface Vlan2
PE1(config-if-Vlan2)#ip vrf forwarding VPNA fallback global
PE1(config-if-Vlan2)#exit
PE1(config)#interface Vlan3
PE1(config-if-Vlan3)#ip address 120.1.3.1 255.255.255.0
PE1(config-if-Vlan2)#exit
PE1(config)#interface Vlan4
PE1(config-if-Vlan4)#label-switching
PE1(config-if-Vlan4)#ldp enable
PE1(config-if-Vlan4)#ip address 202.200.1.2 255.255.255.0
PE1(config-if-Vlan4)#exit
PE1(config)#interface loopback1
PE1(config-if-Loopback1)#ip address 1.1.1.1 255.255.255.255
PE1(config-if-Loopback1)#exit
PE1(config)#router ospf
PE1(config-router)#redistribute static
PE1(config-router)#network 1.1.1.1/32 area 0
PE1(config-router)#network 120.1.2.0/24 area 0
PE1(config-router)#network 120.1.3.0/24 area 0
PE1(config-router)#exit
PE1(config)#router bgp 100
PE1(config-router)#neighbor 2.2.2.2 remote-as 100
PE1(config-router)#neighbor 2.2.2.2 update-source 1.1.1.1
PE1(config-router)#address-family vpnv4 unicast
PE1(config-router-af)#neighbor 2.2.2.2 activate
PE1(config-router-af)#exit-address-family
PE1(config-router)#address-family ipv4 vrf VPNA
PE1(config-router-af)#network 120.1.1.0/24
PE1(config-router-af)#neighbor 10.1.1.1 remote-as 60101
PE1(config-router-af)#no neighbor 10.1.1.1 send-community extended
PE1(config-router-af)#exit-address-family
PE1(config-router)#exit
PE1(config)#router ldp
PE1(config-router)#exit
PE1(config)#ip route 120.1.1.0/24 vlan 2 120.1.1.1
```

The configuration of P is as follows:

```
P#config
P(config)#interface Vlan1
P(config-if-Vlan1)#label-switching
P(config-if-Vlan1)#ldp enable
P(config-if-Vlan1)#ip address 120.1.2.2 255.255.255.0
P(config-if-Vlan1)#exit
P(config)#interface Vlan2
P(config-if-Vlan2)#label-switching
P(config-if-Vlan2)#ldp enable
P(config-if-Vlan2)#ip address 120.1.4.1 255.255.255.0
P(config-if-Vlan2)#exit
P(config)#router ospf
P(config-router)#network 0.0.0.0/0 area 0
P(config-router)#exit
P(config)#router ldp
```

The configuration of PE2 is as follows:

```
PE2#config
PE2(config)#ip vrf VPNA
PE2(config-vrf)#rd 100:10
PE2(config-vrf)#route-target both 100:10
PE2(config-vrf)#exit
PE2(config)#interface Vlan1
PE2(config-if-Vlan1)#label-switching
PE2(config-if-Vlan1)#ldp enable
PE2(config-if-Vlan1)#ip address 120.1.4.2 255.255.255.0
PE2(config-if-Vlan1)#exit
PE2(config)#interface Vlan2
PE2(config-if-Vlan2)#ip address 120.1.5.1 255.255.255.0
PE2(config-if-Vlan2)#exit
PE2(config)#interface Vlan3
PE2(config-if-Vlan3)#ip vrf forwarding VPNA
PE2(config-if-Vlan1)#ip address 10.1.2.2 255.255.255.0
PE2(config-if-Vlan1)#exit
PE2(config)#interface Vlan4
PE2(config-if-Vlan4)#ip vrf forwarding VPNA fallback global
PE2(config-if-Vlan4)#ip address 120.1.7.2 255.255.255.0
PE2(config-if-Vlan4)#exit
PE2(config)#interface Loopback1
PE2(config-if-Loopback1)#ip address 2.2.2.2 255.255.255.255
PE2(config-if-Loopback)#exit
PE2(config)#router ospf
PE2(config-router)#redistribute static
PE2(config-router)#network 2.2.2.2/32 area 0
PE2(config-router)#network 120.1.4.0/24 area 0
PE2(config-router)#network 120.1.5.0/24 area 0
PE2(config-router)#exit
PE2(config)#router bgp 100
PE2(config-router)#neighbor 1.1.1.1 remote-as 100
PE2(config-router)#neighbor 1.1.1.1 update-source 2.2.2.2
PE2(config-router)#address-family vpnv4 unicast
PE2(config-router-af)#neighbor 1.1.1.1 activate
PE2(config-router-af)#exit-address-family
PE2(config-router)#address-family ipv4 vrf VPNA
PE2(config-router-af)#network 120.1.7.0/24
PE2(config-router-af)#neighbor 10.1.2.1 remote-as 60102
PE2(config-router-af)#no neighbor 10.2.1.1 send-community extended
PE2(config-router-af)#exit-address-family
PE2(config-router)#exit
PE2(config)#router ldp
PE2(config-router)#exit
PE2(config)#ip route 120.1.7.0/24 vlan 4 120.1.7.1
```

The configuration of PE3 is as follows:

```
PE3#config
PE3(config)#interface Loopback1
PE3(config-if-Loopback1)#ip address 3.3.3.3 255.255.255.255
PE3(config-if-Loopback1)#exit
PE3(config-if-Vlan1)#interface Vlan1
PE3(config-if-Vlan1)#ip address 120.1.5.2 255.255.255.0
PE3(config-if-Vlan1)#exit
PE3(config)#interface Vlan2
PE3(config-if-Vlan2)#ip address 120.1.3.2 255.255.255.0
PE3(config-if-Vlan2)#exit
PE3(config)#interface Vlan3
PE3(config-if-Vlan3)#ip address 180.1.6.1 255.255.255.0
PE3(config-if-Vlan3)#exit
PE3(config)#router ospf 1
PE3(config-router)#default-information originate
```

```
PE3(config-router)# network 0.0.0.0/0 area 0
PE3(config-router)#exit
PE3(config)#router bgp 100
PE3(config-router)#network 120.1.1.0 mask 255.255.255.0
PE3(config-router)#network 120.1.7.0 mask 255.255.255.0
PE3(config-router)#neighbor 180.1.6.2 remote-as 200
PE3(config-router)#exit
PE3(config)#ip route 0.0.0.0/0 180.1.6.2
```

The configuration of CE2 is as follows:

```
CE2#config
CE2(config)#interface Vlan1
CE2(config-if-Vlan1)#ip address 10.1.2.1 255.255.255.0
CE2(config-if-Vlan1)#exit
CE2(config)#interface Vlan2
CE2(config-if-Vlan2)#ip address 192.168.1.3 255.255.255.0
CE2(config-if-Vlan2)#exit
CE2(config-if-Loopback1)#interface Loopback1
CE2(config-if- Loopback1)#ip address 22.22.22.22 255.255.255.255
CE2(config-if- Loopback1)#exit
CE2(config)#ip route 0.0.0.0/0 192.168.1.4
CE2(config)#router bgp 60101
CE2(config-router)#neighbor 10.1.2.2 remote-as 100
CE2(config-router)#network 192.168.1.0/24
CE2(config-router)#exit
```

The configuration of IGW is as follows:

```
IGW#config
IGW(config)#interface Vlan1
IGW(config-if-Vlan1)#ip address 180.1.5.2 255.255.255.0
IGW(config-if-Vlan1)#exit
IGW(config)#interface Vlan2
IGW(config-if-Vlan2)#ip address 180.1.6.2 255.255.255.0
IGW(config-if-Vlan2)#exit
IGW(config)#router bgp 200
IGW(config-router)#neighbor 180.1.6.1 remote-as 100
IGW(config-router)#exit
```

13.4.4 Public Network Access Troubleshooting

When Public Network Access is configured and used, problems such as incorrect physical connections or configuration errors may cause it to fail. Observe the following points to avoid them:

- First, make sure that the regular MPLS BGP VPN works correctly and that hosts in the private network can communicate with each other. If communication in the VPN fails, refer to MPLS BGP VPN troubleshooting.
- Second, check whether the public network access mode in use is non-VRF or VRF, because their configurations differ considerably.
- In non-VRF mode, configure a filter policy on the non-VRF PE-CE interface to prevent private network routes and traffic from entering the PE through the public network interface; otherwise, there may be security risks. Also make sure that the advertisement of default routes and the NAT configuration on the IGW are correct.
- In VRF mode, use the "ip vrf forwarding vrf_name fallback global" command when configuring the private network interface, so that the global route table is looked up a second time if the private network route lookup fails. Also make sure that the advertisement of default routes and the NAT configuration on the IGW are correct.
- Finally, if all of the above is correct, the CE can access the Internet. Whichever of the above networking modes is used, the other CEs access the Internet by forwarding traffic to the PE through the VPN, and traffic from the Internet is also forwarded through the PE.

13.5 VPLS and VPWS

13.5.1 VPLS Overview

IP networks are now deployed all over the world, and providers focus on using the existing IP networks to offer low-cost private networks. MPLS VPN is a technology that offers VPN services over IP networks and is easy to provision and simple to configure. It has two types: MPLS L3VPN and MPLS L2VPN. Because MPLS L3VPN is involved in the internal route management of users, management is complex for providers. Traditional MPLS L2VPN, in the form of VLL, offers a point-to-point L2VPN service that connects two sites as if they were directly linked. However, it cannot provide multipoint switching for providers. VPLS is developed from traditional MPLS L2VPN and can implement multipoint-to-multipoint VPN networks. VPLS therefore provides a better solution for providers.

VPLS (Virtual Private LAN Service) is a service that offers a LAN over an MPLS network; more precisely, it is an Ethernet-based L2VPN technology. VPLS combines the advantages of Ethernet and MPLS, and allows user networks at different locations to communicate as if they were directly connected. VPLS also enables users to extend their LAN to a MAN or even a WAN.

VPWS is a special case of VPLS; conversely, VPLS is an extension of VPWS. VPLS offers a multipoint-to-multipoint VPN, whereas VPWS offers only a point-to-point L2VPN solution. To the user CE, a VPLS network therefore looks like a switch, and a VPWS network looks like an Ethernet cable.

VPLS can use LDP signaling or BGP signaling to establish PWs. LDP-based signaling establishes virtual circuits through a point-to-point LDP session between two PEs. Because LDP was standardized earlier and is simple to deploy, many products support it. For compatibility, the switch uses LDP-based signaling.

13.5.1.1 Basic Concepts of VPLS

- **VPLS (Virtual Private LAN Service)**

VPLS is a service that offers a LAN over an MPLS network. It allows users at dispersed locations to access the network at the same time as if they were directly connected. VPLS also enables users to extend their LAN to a MAN or even a WAN.

- **VC (Virtual Circuit)**

A unidirectional logical connection between two points. A pair of VCs in opposite directions forms a PW.

- **S-TAG (Service Tag), S-VID (Service VLAN ID)**

Tags assigned by the provider network to identify users.

- **P-TAG (Provider Tag), P-VID (Provider VLAN ID)**

The VLAN tag expected by the peer.

- **VFI (Virtual Forwarding Instance)**

Each VFI provides a separate VPLS service and performs Ethernet bridging. A VFI maps the physical access links of VPLS to PWs and, in the reverse direction, maps PWs to the physical access links.

- **UPE (User-facing Provider Edge)**

An aggregation device through which users access the VPN.

- **NPE (Network Provider Edge)**

A core PE device at the edge of the core domain of a VPLS network. It provides transparent VPLS transmission across the core network.

- **MTU (Maximum Transmission Unit)**

The maximum transmission unit.

- **QinQ (802.1Q in 802.1Q)**

A tunneling protocol based on 802.1Q encapsulation that can offer point-to-multipoint L2VPN service. It encapsulates the private network VLAN tag of a user in a public network tag, so packets with two tags traverse the provider backbone network. This provides users with a simpler Layer 2 VPN tunnel.

- **PW Signaling**

PW signaling is used to establish and maintain PWs and is the basis of VPLS. It can also be used to discover the peer PE devices of a VFI automatically. PW signaling currently includes LDP and BGP. The switch supports LDP only.

- **VPWS (Virtual Private Wire Service)**

A point-to-point L2VPN service. Packets received on one end (AC or PW) are forwarded directly to the other end (PW or AC). Unlike VPLS, VPWS does not need to learn or look up MAC addresses and involves no broadcast or multicast, so it is more efficient.

13.5.1.2 Basic VPLS Network Model

The VPLS network model has five parts: CE, PE, P, AC, and PW. They are described below.

- **CE (Customer Edge)**

An edge device on the user side that connects directly to the provider network. It can be a router, a switch, or a host.

- **PE (Provider Edge)**

An edge device in the provider network that connects to the CE and provides VPN access. It maps and forwards packets between the public network tunnel and the private network.

- **P (Provider)**

A backbone router in the provider network that does not connect to any CE. It needs only basic MPLS forwarding capability and does not maintain VPN information.

- **AC (Attachment Circuit)**

In L2VPN, the CE accesses the PE through an AC. The AC carries frames between the CE and the PE and can be a physical or logical link.

- **PW (Pseudo Wire)**

Simply put, a PW is a VC plus a tunnel; the tunnel can be an LSP, GRE, or CR-LSP. In VPLS, a PW acts like a direct tunnel between ACs that transparently carries the Layer 2 data of users.

RFC 4762 defines two models: the full-mesh PW model and the hierarchical (layered) VPLS model, as shown in the figures below.

13.5.1.3 Advantages of VPLS

Advantages of VPLS:

- VPLS provides an Ethernet interface to users, simplifies the LAN/WAN boundary, and supports fast and flexible service deployment.
- VPLS leaves the control and maintenance of routing policies to users, which simplifies network management for providers.
- All CEs in a VPLS service belong to one subnet, which simplifies IP address configuration.
- The VPLS service does not participate in IP addressing or routing.

13.5.1.4 Basic VPLS Network Model with Full-Mesh PE Connections

All PEs are logically connected to each other. They learn MAC addresses and forward packets among multiple points. The MPLS network provides tunnels that carry packets between VPN sites.

As in L3VPN, P devices forward MPLS packets but do not learn or exchange MAC addresses. To allow MAC addresses to overlap between VPNs, the forwarding tables are independent.

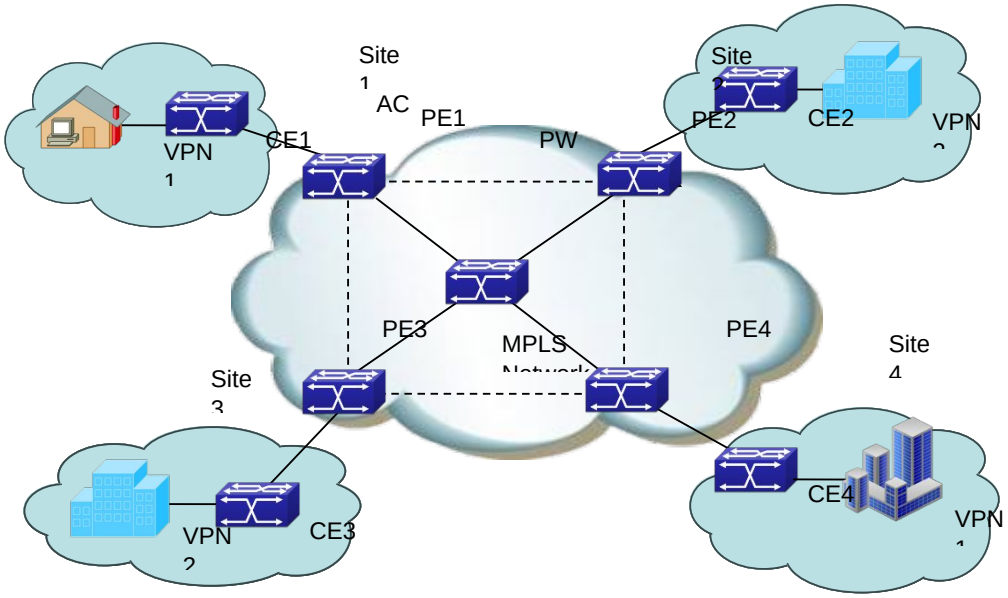


Figure 13_x001E_25 VPLS model of fully connection PE

The full-mesh VPLS model avoids loops through split-horizon forwarding: a packet that a PE receives from a PW is not forwarded to the other PWs of the same VFI. In other words, any two PEs communicate directly through a PW and do not forward packets through a third PE. This is why the PWs of a VFI must be fully meshed.

13.5.1.5 Hierarchical VPLS Model

All NPEs are logically connected to each other. A UPE establishes a virtual connection with the nearest NPE and exchanges packets with VPN sites through the NPE, so the network topology becomes hierarchical. UPEs are mainly used for VPN access and have lower performance requirements, whereas NPEs are mainly used to aggregate traffic and have higher performance requirements. For a more robust network, a backup link can be added between the UPE and the NPE. The virtual connection between the UPE and the NPE can be established through QinQ or LDP.

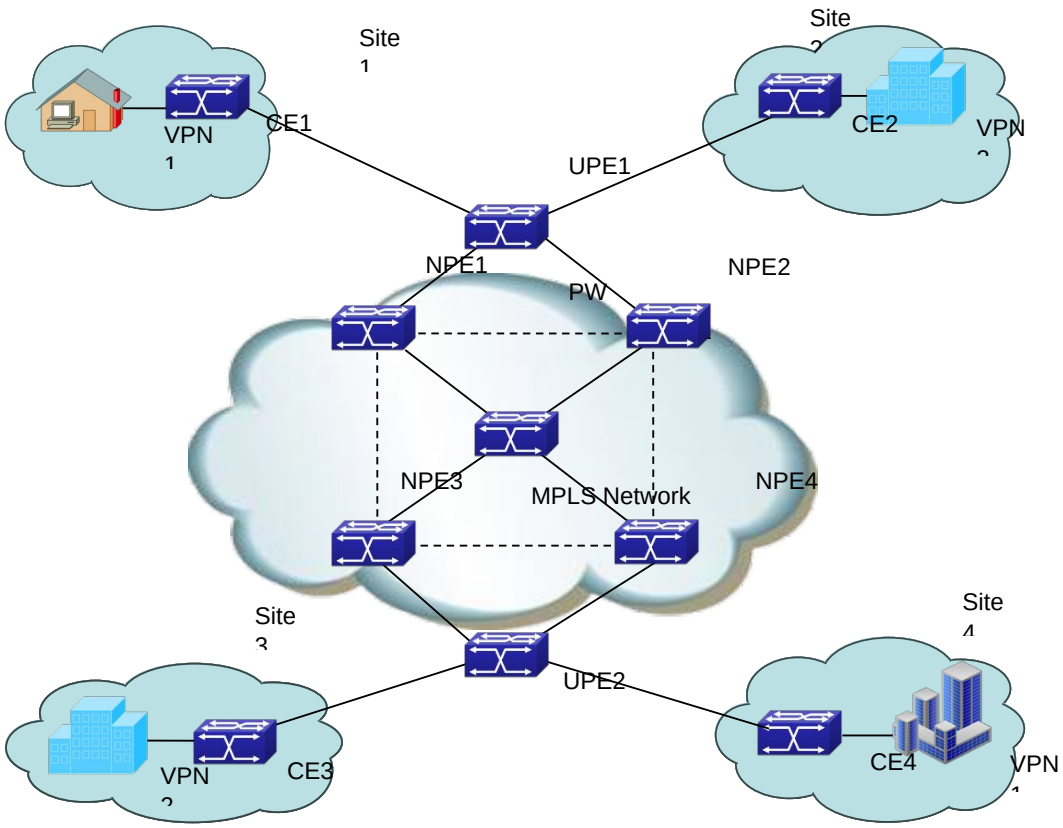


Figure 13-206 Hierarchical VPLS Model

Depending on the connection between the UPE and the NPE, H-VPLS access is divided into the LSP access method and the QinQ access method.

- **LSP Access Method**

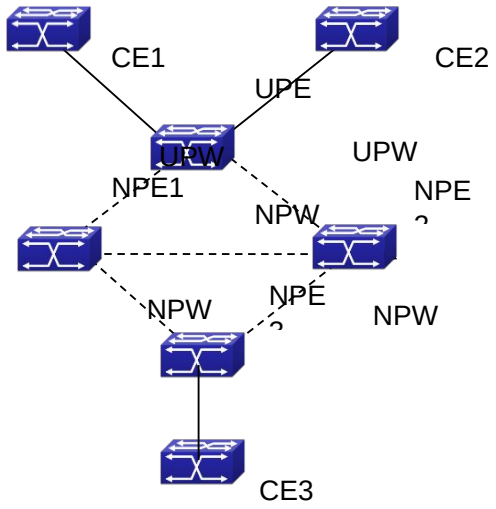


Figure 13-217 LSP Access Method

As shown in the figure, the UPE acts as an aggregation device and connects to NPE1 and NPE2 through LSPs. The UPE must establish a virtual U-PW with NPE1 and with NPE2 (a U-PW requires a new VFI with the peers configured on the UPE and the NPE, and the PW IDs on the two devices must be the same). The UPE does not establish virtual connections with any other devices.

The data forwarding process in LSP access mode is as follows:

(1) The UPE sends packets from the CE to NPE1 with the VC label of the corresponding U-PW (this label is assigned by NPE1 and serves as the multiplexing label that identifies the PW).

(2) When NPE1 receives a packet, it examines the VC label to determine the VFI, pushes the corresponding VC label based on the destination MAC address, and forwards the packet.

(3) When NPE1 receives a packet from an N-PW, it adds the corresponding VC label and forwards the packet to the UPE, which forwards it to the CE.

When CE1 and CE2 exchange data as local CEs, the UPE forwards the packets between them because it has a bridging function. However, for packets with an unknown destination MAC address and for broadcast packets, the UPE broadcasts them to CE2 through the bridge and also forwards them to NPE1 through the U-PW. NPE1 then replicates the packets and forwards them to the CEs on the remote side.

- **QinQ Access Method**

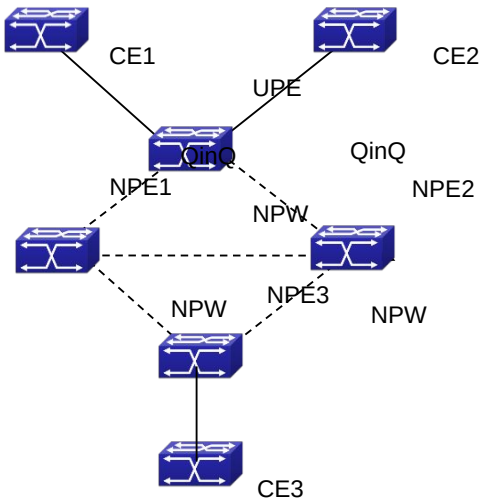


Figure 13-228 QinQ Access Method

As shown in the figure, the UPE is a standard bridging device that establishes a point-to-point Ethernet QinQ connection with the PE (that is, QinQ is applied to the CE-facing interface, and VLAN access mode is applied on PE1). Packets received by the UPE are tagged with an outer VLAN tag. When they reach PE1, the outer VLAN tag is interpreted under VLAN access mode as the provider VLAN tag, that is, the service delimiter. Based on this delimiter, packets are mapped to the relevant VFI, which then determines how to forward them (unicast or multicast).

The data forwarding process in QinQ access mode is as follows:

(1) QinQ is enabled on the CE access port. The UPE adds the multiplexing tag to packets and transparently transmits them to PE1 through the QinQ tunnel between the UPE and PE1.

(2) When PE1 receives a packet, it examines the VLAN tag to determine the VFI, pushes the relevant PW label based on the destination MAC address, and forwards the packet.

(3) When PE1 receives a packet from a PW, it determines the VFI based on the PW label, adds a VLAN tag based on the destination MAC address, and forwards the packet through the QinQ tunnel to the UPE, which forwards it to the CE.

When CE1 and CE2 exchange data as local CEs, the UPE forwards the packets between them because it has a bridging function. However, for packets with an unknown destination MAC address and for broadcast packets, the UPE broadcasts them to CE2 through the bridge and also forwards them to PE1 through QinQ. PE1 then replicates the packets and forwards them to the CEs on the remote side.

- **Loop Avoidance in the H-VPLS Model**

Loop avoidance in the hierarchical VPLS model differs from that in the full-mesh model. H-VPLS requires full-mesh connections only between NPEs, not between UPEs and NPEs. Therefore, packets received from a PW are not forwarded to PWs connected to other NPEs, but they can be forwarded to PWs connected to UPEs. In addition, packets received from a PW connected to a UPE can be forwarded to PWs connected to other NPEs.

13.5.1.6 Packet Forwarding of VPLS

In the VPLS model, packet encapsulation differs on the AC and on the PW.

On the AC, there are two access methods, VLAN access and Ethernet access, described below:

- **VLAN Access**

The Ethernet frame header sent from the CE to the PE or from the PE to the CE carries a VLAN tag, which is a delimiter added by the provider device to distinguish users. This tag is called the S-TAG.

- **Ethernet Access**

The Ethernet frame header sent from the CE to the PE or from the PE to the CE carries no delimiter. If the header contains a VLAN tag, the tag is an inner user tag and is meaningless to the PE. This tag is called the U-TAG.

On the PW, there are two packet encapsulation modes, Raw mode and Tagged mode, described below:

- **Raw Mode**

A frame carried on the PW cannot contain an S-TAG. For packets from the CE: if a packet with a delimiter is received, the delimiter is removed, and the PW and tunnel labels are pushed before forwarding; if a packet without a delimiter is received, the PW and tunnel labels are pushed directly before forwarding. For packets sent from the PE down to the CE, a delimiter is added or not according to the configuration, but existing tags cannot be rewritten or removed.

- **Tagged Mode**

A frame carried on the PW must contain an S-TAG. For packets from the CE: if a packet with a delimiter is received, the delimiter is compared with the VLAN tag expected by the peer. If they are the same, the S-TAG is kept; otherwise, the S-TAG is rewritten to the expected VLAN tag. If the peer does not send an expected VLAN tag, a null tag (tag value 0) is used. Finally, the PW and tunnel labels are pushed before forwarding. If a packet without a delimiter is received, the expected VLAN tag or a null tag is added, and then the PW and tunnel labels are pushed before forwarding. For packets sent from the PE down to the CE, delimiters are rewritten, removed, or kept according to the configuration before forwarding.

Depending on the AC access mode and the PW packet encapsulation mode, VPLS packet forwarding is divided into the following four types.

- **Forwarding in Ethernet Access with Raw Mode**

As shown in the figure, the AC uses Ethernet access mode and the PW uses Raw encapsulation mode. Packets are forwarded from CE1 to CE4 as follows:

1. CE1 sends packets with a U-TAG to PE1.

2. PE1 selects the appropriate PW based on the destination MAC address and adds a VC label to the packets.

3. To forward the packets through the MPLS tunnel in the public network, PE1 adds a public network tunnel label and forwards the packets to PE4.

4. When PE4 receives the packets, it identifies the relevant VFI based on the VC label and sends the packets with the U-TAG to CE4.

- **Forwarding in Ethernet Access with Tagged Mode**

As shown in the figure, the AC uses Ethernet access mode and the PW uses Tagged encapsulation mode. The process is similar to Ethernet access with Raw mode. The difference is that frames on the PW must carry an S-TAG. When PE1 receives packets without an S-TAG, it first adds the expected VLAN tag or a null tag and then pushes two layers of MPLS labels before forwarding. When PE4 receives the packets, it removes the two label layers and the S-TAG before forwarding the packets to CE4.

Forwarding in VLAN Access with Raw Mode

As shown in the figure, the AC uses VLAN access mode and the PW uses Raw encapsulation mode. Packets are forwarded from CE1 to CE4 as follows:

1. Packets sent by CE1 must carry an S-TAG. If packets from CE1 carry no VLAN tag or an unmatched VLAN tag, they are processed by normal Layer 2 forwarding.

2. When PE1 receives the packets, it removes the S-TAG, adds two layers of MPLS labels, and forwards the packets to PE4 through the public network MPLS tunnel.

3. PE4 removes the two layers of MPLS labels, adds the S-TAG, and forwards the packets to CE4.

Forwarding in VLAN Access with Tagged Mode

As shown in the figure, the AC uses VLAN access mode and the PW uses Tagged encapsulation mode. The process is similar to VLAN access with Raw mode. The difference is that frames on the PW carry an S-TAG. When PE1 receives packets, it compares the S-TAG with the expected VLAN tag sent by the peer PE. If they are the same, the S-TAG is kept; otherwise, the S-TAG is rewritten to the expected VLAN tag. If there is no expected VLAN tag, a null tag (tag value 0) is used.

13.5.2 VPLS Configuration

VPLS configuration task sequence:

1. Enable a routing protocol globally (required)

2. Configure basic MPLS (required)

(1) Enable MPLS globally

(2) Enable label switching on interfaces

3. Configure LDP sessions (required)

(1) Enter LDP configuration mode in Global Mode, which enables LDP

(2) Configure remote LDP neighbors with the targeted-peer command

(3) Enable LDP on Layer 3 interfaces

4. Create a PW template (optional)

(1) Create a pw-class and enter PW template configuration mode

(2) Set the transport mode of the pw-class

5. Configure a VFI (required)

(1) Create a VFI, specify the VFI ID, and enter VFI configuration mode

(2) Configure the endpoints of remote PWs (usually the same as the LDP targeted peers)

(3) Specify a pw-class for each peer

(4) Set the PW encapsulation mode of the VFI

6. Configure the user access mode and bind a VFI (required for VPLS)

(1) Enter Port Mode

(2) Bind the port to a VFI

7. Bind a PW to a port (required for VPWS)

(1) Create an L2VC and configure the peer PE of VPWS

(2) Enter Port Mode

(3) Bind the port to the specified PW and set the access mode

1. Enable a routing protocol globally (required)

Command	Description
Global Mode	
router ospf	Enters OSPF Router Mode.
OSPF Router Mode	
network 0.0.0.0/0 area 0	Specifies the network segment of the interfaces on which OSPF is enabled. By default, OSPF is disabled on interfaces.

2. Enable MPLS globally (required)

Command	Description
Global Mode	
mpls enable **no mpls enable**	Enables MPLS. The no command disables MPLS.
Interface Mode (Layer 3 interface)	
label-switching enable **no label-switching**	Enables label switching. The no command disables label switching. By default, label switching is disabled.

3. Configure LDP sessions (required)

(1) Enter LDP configuration mode in Global Mode, which enables LDP

(2) Configure remote LDP neighbors with the targeted-peer command

(3) Enable LDP on Layer 3 interfaces

Command	Description
Global Mode	
router ldp **no router ldp**	Enables LDP. The no command disables LDP.
targeted-peer 1.1.1.1	Configures a remote LDP neighbor. By default, no remote LDP neighbor is configured.
Interface Mode	
ldp {enable | disable}	Enables LDP on the interface. The no command disables LDP on the interface.

4. Create a PW template (optional)

(1) Create a pw-class and enter PW template configuration mode

(2) Set the transport mode of the pw-class

Command	Description
Global Mode	
pw-class <pw-class-name> **no pw-class <pw-class-name>**	Creates a pw-class (PW template). The no command deletes the pw-class. By default, no PW template exists.
PW Template Mode	
transport-mode {ethernet | vlan}	Sets the packet encapsulation mode of the PW template: ethernet corresponds to Raw mode, and vlan corresponds to Tagged mode.

5. Configure a VFI (required)

(1) Create a VFI, specify the VFI ID, and enter VFI configuration mode

(2) Configure the endpoints of remote PWs (usually the same as the LDP targeted peers)

(3) Specify a pw-class for each peer

(4) Set the PW encapsulation mode of the VFI

Command	Description
Global Mode	
vfi <vfi-name> <vfi-id> **no vfi <vfi-name>**	Creates a VFI and specifies the VFI ID. The no command deletes the VFI. By default, no VFI exists.
VFI Mode	
peer ip-address [pw-id pw-id] [no-split-horizon] [pw-class class-name]	Configures a peer PE of the VPLS instance, including whether split horizon is enabled (enabled by default) and the PW template to use.
transport-mode {ethernet | vlan}	Sets the packet encapsulation mode of the PW.

6. Configure the user access mode and bind a VFI (required for VPLS)

(1) Enter Port Mode

(2) Bind the port to a VFI

Command	Description
Port Mode	
xconnect vfi vfi-id [mode {ethernet | vlan [svid svid]}]	Binds the port to a VFI and sets the AC access mode. By default, a port is not bound to any VFI.

7. Bind a PW to a port (required for VPWS)

(1) Create an L2VC and configure the peer PE of VPWS

(2) Enter Port Mode

(3) Bind the port to the specified PW and set the access mode

Command	Description
Global Mode	
l2-vc ip-address pw-id pw-id [group group-id] [pw-class class-name]	Creates an L2VC and configures the peer PE of VPWS. By default, no peer PE is configured.
Port Mode	
xconnect l2-vc pw-id <pw-id> [mode {ethernet | vlan [svid <svid>]}]	Binds the port to the specified PW and sets the access mode. By default, a port is not bound to any PW.

13.5.3 Typical Examples of VPLS

13.5.3.1 Full-Mesh VPLS Configuration

13.5.3.1.1 Network Requirements

- CE1 and CE2 are at two different sites but belong to the same VPN1.
- CE1 and CE2 access the PEs through port Eth1/0/1.
- CE1 accesses PE1 in Ethernet mode, and CE2 accesses PE2 in VLAN mode with Svid 200.
- The packet encapsulation mode between PE1 and PE2 is Raw (Ethernet) mode, and the mode between PE3 and PE1 (PE2) is Tagged (VLAN) mode.
- Configure VPLS to establish Layer 2 VPN1 so that CE1 and CE2 can communicate with each other at Layer 2.

13.5.3.1.2 Network Diagram

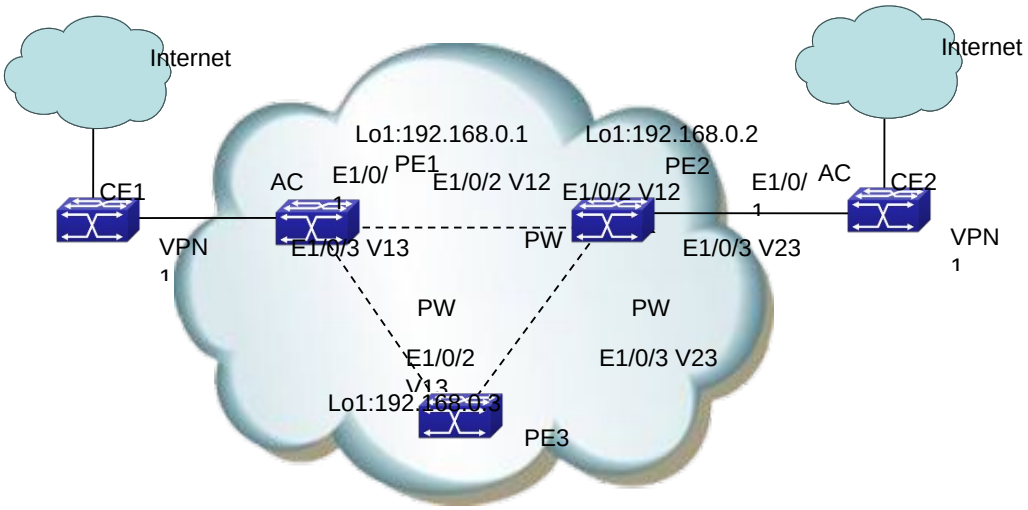


Figure 13-239 Full-Mesh VPLS Model for PEs

13.5.3.1.3 Configuration Steps

Configure the IP address and mask of each interface, including VLAN and loopback interfaces, as shown in the figure above. This configuration is omitted. The following example is for VPLS with LDP.

(1) PE1 configuration

#Create PW template

```
PE1(config)#pw-class c1
PE1(config-class) #transport-mode ethernet
PE1(config-class)#exit
PE1(config)#pw-class c2
PE1(config-class)#transport-mode vlan
PE1(config-class)#exit
```

#Configure routing

```
PE1(config)#router ospf
PE1(config-router)#router-id 192.168.0.1
PE1(config-router)#network 0.0.0.0/0 area 0
PE1(config-router)#exit
```

#Configure MPLS and label switching (in non-H-VPLS, the interface connected to the CE has label switching enabled)

```
PE1(config)#mpls enable
PE1(config)#int vlan 12
PE1(config-if-vlan12)#label-switching
PE1(config-if-vlan12)#exit
PE1(config)#int vlan 13
PE1(config-if-vlan13)#label-switching
PE1(config-if-vlan13)#exit
```

#Configure LDP

```
PE1(config)#router ldp
PE1(config-router)#router-id 192.168.0.1
PE1(config-router)#targeted-peer 192.168.0.2
PE1(config-router)#targeted-peer 192.168.0.3
PE1(config-router)#exit
PE1(config)#int vlan 12
PE1(config-if-vlan12)#ldp enable
PE1(config-if-vlan12)#exit
PE1(config)#int vlan 13
```

```
PE1(config-if-vlan13)#ldp enable
PE1(config-if-vlan13)#exit
```

#Configure instance V1 and the remote PEs in LDP

```
PE1(config)#vfi v1 100
PE1(config-vfi)#peer 192.168.0.2 pw-class c1
PE1(config-vfi)#peer 192.168.0.3 pw-class c2
PE1(config-vfi)#exit
```

#Bind the port to the VPLS instance in Ethernet mode

```
PE1(config-if-ethernet1/0/1)#xconnect vfi 100 mode ethernet
```

(2) PE2 configuration

#Create PW template

```
PE2(config)#pw-class c1
PE2(config-class)#transport-mode ethernet
PE2(config-class)#exit
PE2(config)#pw-class c2
PE2(config-class)#transport-mode vlan
PE2(config-class)#exit
```

#Configure routing

```
PE2(config)#router ospf
PE2(config-router)#router-id 192.168.0.2
PE2(config-router)#network 0.0.0.0/0 area 0
PE2(config-router)#exit
```

#Configure MPLS and label switching (in non-H-VPLS, the interface connected to the CE does not have label switching enabled)

```
PE2(config)#mpls enable
PE2(config)#int vlan 12
PE2(config-if-vlan12)#label-switching
PE2(config-if-vlan12)#exit
PE2(config)#int vlan 23
PE2(config-if-vlan23)#label-switching
PE2(config-if-vlan23)#exit
```

#Configure LDP

```
PE2(config)#router ldp
PE2(config-router)#router-id 192.168.0.2
PE2(config-router)#targeted-peer 192.168.0.1
PE2(config-router)#targeted-peer 192.168.0.3
PE2(config-router)#exit
PE2(config)#int vlan 12
PE2(config-if-vlan12)#ldp enable
PE2(config-if-vlan12)#exit
PE2(config)#int vlan 23
PE2(config-if-vlan23)#ldp enable
PE2(config-if-vlan23)#exit
```

#Configure instance V1 and the remote PEs in LDP

```
PE2(config)#vfi v1 100
PE2(config-vfi)#peer 192.168.0.1 pw-class c1
PE2(config-vfi)#peer 192.168.0.3 pw-class c2
PE2(config-vfi)#exit
```

#Bind the port to the VPLS instance in VLAN mode with Svid 200

```
PE2(config-if-etherne1/0/1)#xconnect vfi 100 mode vlan svid 200
```

(3) PE3 configuration

The configuration is similar to that of PE1 and PE2.

13.5.3.2 H-VPLS Access with LSP

13.5.3.2.1 Network Requirements

- CE1 and CE2 are at two sites but belong to the same VPN1.
- CE1 and CE2 access the PEs through port Eth1/0/1.
- N-PE1, N-PE2, and N-PE3 form a full-mesh VPLS network.
- U-PE is the user access device; it accesses N-PE1 through a hierarchical VPLS PW.
- CE1 accesses U-PE in Ethernet mode, and CE2 accesses N-PE3 in VLAN mode with Svid 200.
- The packet encapsulation mode between N-PEs is Raw (Ethernet) mode.
- U-PE and N-PE1 are connected through a PW.
- Configure VPLS to establish Layer 2 VPN1 so that CE1 and CE2 can communicate with each other.

13.5.3.2.2 Network Diagram

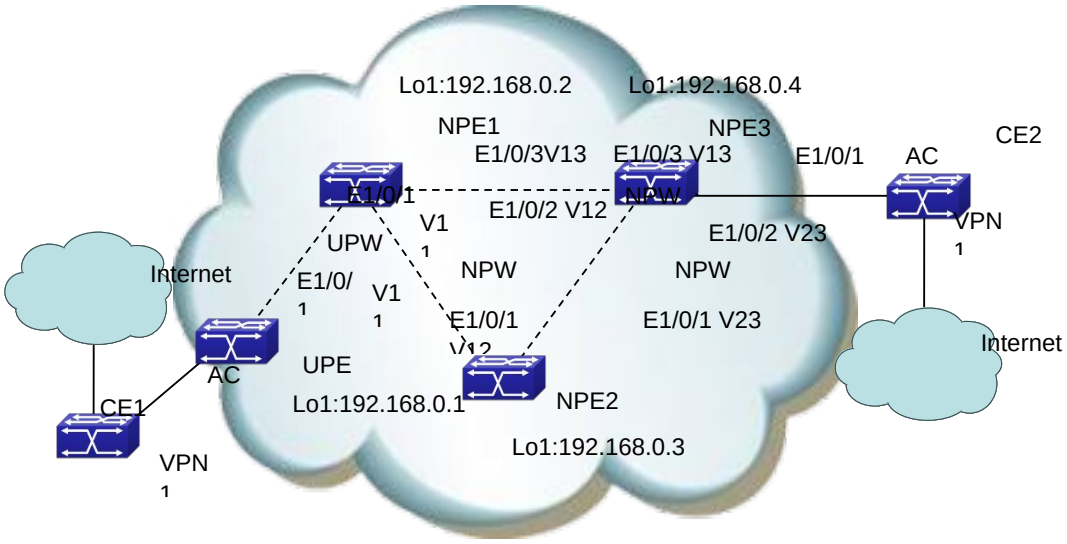


Figure 13_x001E_30 H_VPLS network in LSP access mode

13.5.3.2.3 Configuration Steps

Configure the IP address and mask of each interface, including VLAN and loopback interfaces, as shown in the figure above. This configuration is omitted. The following configuration applies only to H-VPLS with LDP.

(1) UPE configuration

#Create PW template

```
UPE(config)#pw-class c1
UPE(config-class)#transport-mode ethernet
UPE(config-class)#exit
```

#Configure routing

```
UPE(config)#router ospf
UPE(config-router)#router-id 192.168.0.1
UPE(config-router)#network 0.0.0.0/0 area 0
UPE(config-router)#exit
```

#Configure MPLS and label switching (the interface connected to the CE does not have label switching enabled)

```
UPE(config)#mpls enable
UPE(config)#int vlan 11
UPE(config-if-vlan11)#label-switching
```

```
UPE(config-if-vlan11)#exit
```

#Configure LDP

```
UPE(config)#router ldp
UPE(config-router)#router-id 192.168.0.1
UPE(config-router)#targeted-peer 192.168.0.2
UPE(config-router)#exit
UPE(config)#int vlan 11
UPE(config-if-vlan11)#ldp enable
UPE(config-if-vlan11)#exit
```

#Configure instance V1 and the remote PE in LDP

```
UPE(config)#vfi v1 100
UPE(config-vfi)#peer 192.168.0.2 pw-class c1
UPE(config-vfi)#exit
```

#Bind the port to the VPLS instance in Ethernet mode

```
UPE(config-if-ethernet1/0/1)#xconnect vfi 100 mode ethernet
```

(2) NPE1 configuration

#Create PW template

```
NPE1(config)#pw-class c1
NPE1(config-class)#transport-mode ethernet
NPE1(config-class)#exit
```

#Configure routing

```
NPE1(config)#router ospf
NPE1(config-router)#router-id 192.168.0.2
NPE1(config-router)#network 0.0.0.0/0 area 0
NPE1(config-router)#exit
```

#Configure MPLS and label switching

```
NPE1(config)#mpls enable
NPE1(config)#int vlan 11
NPE1(config-if-vlan11)#label-switching
NPE1(config-if-vlan11)#exit
NPE1(config)#int vlan 12
NPE1(config-if-vlan12)#label-switching
NPE1(config-if-vlan12)#exit
NPE1(config)#int vlan 13
NPE1(config-if-vlan13)#label-switching
NPE1(config-if-vlan13)#exit
```

#Configure LDP

```
NPE1(config)#router ldp
NPE1(config-router)#router-id 192.168.0.2
NPE1(config-router)#targeted-peer 192.168.0.1
NPE1(config-router)#targeted-peer 192.168.0.3
NPE1(config-router)#targeted-peer 192.168.0.4
NPE1(config-router)#exit
NPE1(config)#int vlan 11
NPE1(config-if-vlan11)#ldp enable
NPE1(config-if-vlan11)#exit
NPE1(config)#int vlan 12
NPE1(config-if-vlan12)#ldp enable
NPE1(config-if-vlan12)#exit
NPE1(config)#int vlan 13
NPE1(config-if-vlan13)#ldp enable
NPE1(config-if-vlan13)#exit
```

#Configure instance V1 and the remote PEs in LDP (split horizon must be disabled for the UPE peer)

```
NPE1(config)#vfi v1 100
```

```
NPE1(config-vfi)#peer 192.168.0.3 pw-class c1
NPE1(config-vfi)#peer 192.168.0.4 pw-class c1
NPE1(config-vfi)#peer 192.168.0.1 no-split-horizon pw-class c1
NPE1(config-vfi)#exit
```

(3) NPE2 configuration

The configuration is similar to that of NPE1, except that there is no UPE peer, so split horizon is not disabled.

(4) NPE3 configuration

#Create PW template

```
NPE3(config)#pw-class c1
NPE3(config-class)#transport-mode ethernet
NPE3(config-class)#exit
```

#Configure routing

```
NPE3(config)#router ospf
NPE3(config-router)#router-id 192.168.0.4
NPE3(config-router)#network 0.0.0.0/0 area 0
NPE3(config-router)#exit
```

#Configure MPLS and label switching

```
NPE3(config)#mpls enable
NPE3(config)#int vlan 13
NPE3(config-if-vlan11)#label-switching
NPE3(config-if-vlan11)#exit
NPE3(config)#int vlan 23
NPE3(config-if-vlan12)#label-switching
NPE3(config-if-vlan12)#exit
```

#Configure LDP

```
NPE3(config)#router ldp
NPE3(config-router)#router-id 192.168.0.4
NPE3(config-router)#targeted-peer 192.168.0.2
NPE3(config-router)#targeted-peer 192.168.0.3
NPE3(config-router)#exit
NPE3(config)#int vlan 13
NPE3(config-if-vlan11)#ldp enable
NPE3(config-if-vlan11)#exit
NPE3(config)#int vlan 23
NPE3(config-if-vlan12)#ldp enable
NPE3(config-if-vlan12)#exit
```

#Configure instance V1 and the remote PEs in LDP

```
NPE3(config)#vfi v1 100
NPE3(config-vfi)#peer 192.168.0.2 pw-class c1
NPE3(config-vfi)#peer 192.168.0.3 pw-class c1
NPE3(config-vfi)#exit
```

#Bind the port to the VPLS instance

```
N-PE3(config-if-ethernet1/0/1)#xconnect vfi 100 mode vlan svid 200
```

13.5.3.3 H-VPLS Access with QinQ

13.5.3.3.1 Network Requirements

- CE1 and CE2 are at two sites but belong to the same VPN1.
- CE1 and CE2 access the PEs through ports Eth1/0/2 and Eth1/0/1, respectively.
- U-PE is the user access device; it accesses N-PE1 through hierarchical QinQ with Svid 100.
- CE2 uses normal VLAN access mode with Svid 200.

- The packet encapsulation mode between N-PEs is Tagged mode.
- U-PE and N-PE1 are connected through QinQ.
- Configure VPLS to establish Layer 2 VPN1 so that CE1 and CE2 can communicate with each other.

13.5.3.3.2 Network Diagram

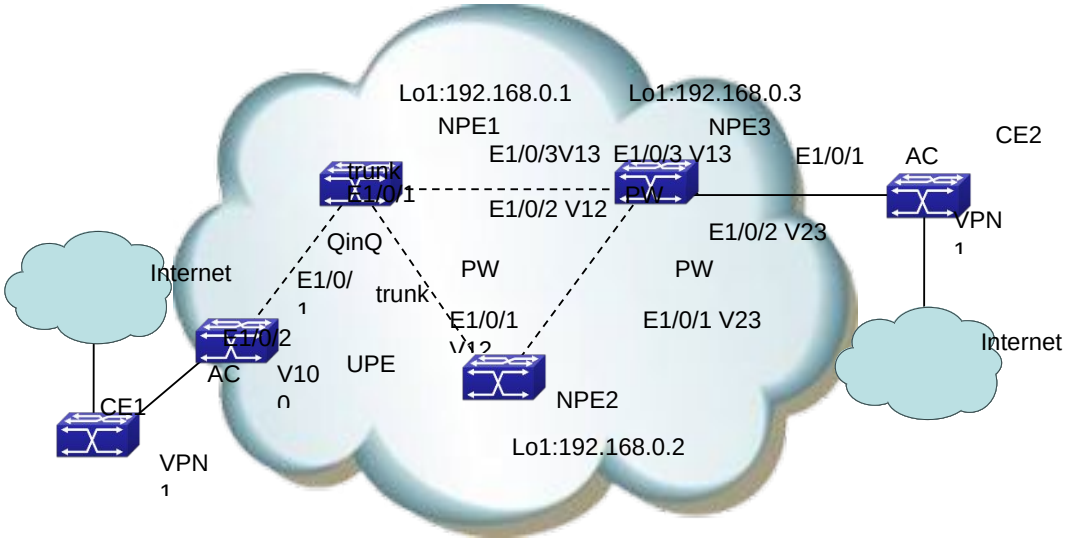


Figure 13_x001E_31 H_VPLS network in QinQ access mode

13.5.3.3.3 Configuration Steps

Configure the IP address and mask of each interface, including VLAN and loopback interfaces, as shown in the figure above. This configuration is omitted. The following configuration applies only to H-VPLS in QinQ mode.

(1) UPE configuration

#Enable QinQ on the port; the port belongs to VLAN 100

```
UPE(config-if-ethernet1/0/2)#switchport access vlan 100
UPE(config-if-ethernet1/0/2)#dot1q-tunnel enable
```

#Configure the port connected to NPE1 as a trunk port; by default, a trunk port allows all VLANs

```
UPE(config-if-ethernet1/0/1)#switchport mode trunk
```

(2) NPE1 configuration

#Create PW template

```
NPE1(config)#pw-class c1
NPE1(config-class)#transport-mode Vlan
NPE1(config-class)#exit
```

#Configure routing

```
NPE1(config)#router ospf
NPE1(config-router)#router-id 192.168.0.1
NPE1(config-router)#network 0.0.0.0/0 area 0
NPE1(config-router)#exit
```

#Configure MPLS and label switching

```
NPE1(config)#mpls enable
NPE1(config)#int vlan 12
NPE1(config-if-vlan12)#label-switching
NPE1(config-if-vlan12)#exit
NPE1(config)#int vlan 13
NPE1(config-if-vlan13)#label-switching
```

```
NPE1(config-if-vlan13)#exit
```

#Configure LDP

```
NPE1(config)#router ldp
NPE1(config-router)#router-id 192.168.0.1
NPE1(config-router)#targeted-peer 192.168.0.2
NPE1(config-router)#targeted-peer 192.168.0.3
NPE1(config-router)#exit
NPE1(config)#int vlan 12
NPE1(config-if-vlan12)#ldp enable
NPE1(config-if-vlan12)#exit
NPE1(config)#int vlan 13
NPE1(config-if-vlan13)#ldp enable
NPE1(config-if-vlan13)#exit
```

#Configure instance V1 and the remote PEs in LDP

```
NPE1(config)#vfi v1 100
NPE1(config-vfi)#peer 192.168.0.2 pw-class c1
NPE1(config-vfi)#peer 192.168.0.3 pw-class c1
NPE1(config-vfi)#exit
```

#Bind the port to the VPLS instance in VLAN mode for UPE QinQ access, with Svid 100

```
NPE1(config-if-ethernet1/0/2)#switchport mode trunk
NPE1 (config-if-ethernet1/0/2)#xconnect vfi 100 mode vlan svid 100
```

(3) NPE2 configuration

The configuration is similar to that of NPE1.

(4) NPE3 configuration

#Create PW template

```
NPE3(config)#pw-class c1
NPE3(config-class)#transport-mode Vlan
NPE3(config-class)#exit
```

#Configure routing

```
NPE3(config)#router ospf
NPE3(config-router)#router-id 192.168.0.3
NPE3(config-router)#network 0.0.0.0/0 area 0
NPE3(config-router)#exit
```

#Configure MPLS and label switching

```
NPE3(config)#mpls enable
NPE3(config)#int vlan 13
NPE3(config-if-vlan11)#label-switching
NPE3(config-if-vlan11)#exit
NPE3(config)#int vlan 23
NPE3(config-if-vlan12)#label-switching
NPE3(config-if-vlan12)#exit
```

#Configure LDP

```
NPE3(config)#router ldp
NPE3(config-router)#router-id 192.168.0.3
NPE3(config-router)#targeted-peer 192.168.0.1
NPE3(config-router)#targeted-peer 192.168.0.2
NPE3(config-router)#exit
NPE3(config)#int vlan 13
NPE3(config-if-vlan11)#ldp enable
NPE3(config-if-vlan11)#exit
NPE3(config)#int vlan 23
NPE3(config-if-vlan12)#ldp enable
NPE3(config-if-vlan12)#exit
```

#Configure instance V1 and the remote PEs in LDP

```
NPE3(config)#vfi v1 100
NPE3(config-vfi)#peer 192.168.0.1 pw-class c1
NPE3(config-vfi)#peer 192.168.0.2 pw-class c1
NPE3(config-vfi)#exit
```

#Bind the port to the VPLS instance

```
N-PE3(config-if-ethernet1/0/1)#xconnect vfi 100 mode vlan svid 200
```

13.5.3.4 VPWS Configuration

13.5.3.4.1 Network Requirements

- CE1 and CE2 are at two sites but belong to the same VPN1.
- CE1 and CE2 access the PEs through port Eth1/0/1.
- CE1 accesses PE1 in Ethernet mode, and CE2 accesses PE2 in VLAN mode with Svid 200.
- The packet encapsulation mode between PE1 and PE2 is Raw (Ethernet) mode.
- Configure VPWS to establish Layer 2 VPN1 so that CE1 and CE2 can communicate with each other.

13.5.3.4.2 Network Diagram

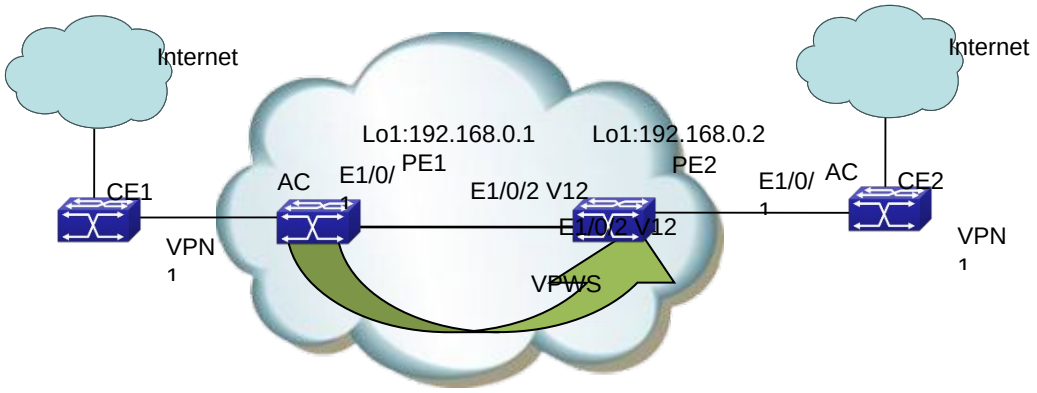


Figure 13_x001E_32 VPWS group network diagram

13.5.3.4.3 Configuration Steps

Configure the IP address and mask of each interface, including VLAN and loopback interfaces, as shown in the figure above. This configuration is omitted. The following configuration applies only to VPWS.

(1) PE1 configuration

#Create PW template

```
PE1(config)#pw-class c1
PE1(config-class)#transport-mode ethernet
PE1(config-class)#exit
```

#Configure routing

```
PE1(config)#router ospf
PE1(config-router)#router-id 192.168.0.1
PE1(config-router)#network 0.0.0.0/0 area 0
PE1(config-router)#exit
```

#Configure MPLS and label switching (in non-H-VPLS, the interface connected to the CE does not have label switching enabled)

```
PE1(config)#mpls enable
PE1(config)#int vlan 12
```

```
PE1(config-if-vlan12)#label-switching
PE1(config-if-vlan12)#exit
```

#Configure LDP

```
PE1(config)#router ldp
PE1(config-router)#router-id 192.168.0.1
PE1(config-router)#targeted-peer 192.168.0.2
PE1(config-router)#exit
```

#Configure the remote PE of VPWS

```
PE1(config)#l2-vc 192.168.0.2 pw-id 1 pw-class c1
```

#Bind the port to VPWS in Ethernet mode

```
PE1(config-if-ethernet1/0/1)#xconnect l2-vc pw-id 1 mode Ethernet
```

(2) PE2 configuration

#Create PW template

```
PE2(config)#pw-class c1
PE2(config-class)#transport-mode vlan
PE2(config-class)#exit
```

#Configure routing

```
PE2(config)#router ospf
PE2(config-router)#router-id 192.168.0.2
PE2(config-router)#network 0.0.0.0/0 area 0
PE2(config-router)#exit
```

#Configure MPLS and label switching (in non-H-VPLS, the interface connected to the CE does not have label switching enabled)

```
PE2(config)#mpls enable
PE2(config)#int vlan 12
PE2(config-if-vlan12)#label-switching
PE2(config-if-vlan12)#exit
```

#Configure LDP

```
PE2(config)#router ldp
PE2(config-router)#router-id 192.168.0.2
PE2(config-router)#targeted-peer 192.168.0.1
PE2(config-router)#exit
PE2(config)#int vlan 12
PE2(config-if-vlan12)#ldp enable
PE2(config-if-vlan12)#exit
```

#Configure the remote PE of VPWS

```
PE2(config)#l2-vc 192.168.0.1 pw-id 1 pw-class c1
```

#Bind the port to VPWS in VLAN mode with Svid 200

```
PE2(config-if-ethernet1/0/1)#xconnect l2-vc pw-id 1 mode vlan svid 200
```

13.5.4 VPLS Troubleshooting

When VPLS is configured and used, the L2VPN may not work correctly because of physical connection problems or configuration errors. Check the following:

- First, make sure that OSPF neighbors among PE1, P, and PE2 work correctly and that routes, including routes to loopback interfaces, are correct; that is, check that each PE has routes to the peer PE and to P.
- Second, make sure that PE1, P, and PE2 have MPLS and LDP enabled globally, and label switching and LDP enabled on the active interfaces. Also check that the LDP sessions on them are correct. Make sure that the LDP remote neighbor (the targeted peer address) is correct and

that the neighbor is established. The address of the LDP remote peer must be the same as the peer address in the VFI or L2VC.

- Then, to check whether the PWs on the PEs are established correctly, use the **show vpls peer xxx** command. The correct PW state is up; otherwise, check whether the PW encapsulation (Ethernet or VLAN), VFI ID, MTU, transport mode, and other parameters are the same on both peers.
- Check whether each VFI port is bound to the correct VFI and whether the access mode and Svid are correct. In VPLS mode, also check that the pw-id bound to the port is correct and that the pw-ids bound on the PEs are the same.
- Finally, if all of the above is correct, use the **show vfi mac-addresses-table** command to check the MAC addresses in the local VPN; each side should display the MAC address of the peer CE.
- If the MAC address of the peer CE is not displayed and traffic does not recover after the devices are restarted, wait for a while, because establishing OSPF, LDP, and PW connections takes time.
- **Note:** Do not enable 802.1X, STP, QinQ, or similar functions on a PE port after it is bound to a VFI, because this may affect the user network. Also, do not add this port to a Layer 3 interface used by the public network, because routing protocols (such as OSPF and RIP) and multicast protocols (such as IGMP and MLD) on that Layer 3 interface may leak provider information to the CE.

Chapter 14 Data Center Configuration

14.1 MC-LAG

14.1.1 Introduction to MC-LAG

MC-LAG (Multichassis Link Aggregation Group) is a mechanism that implements link aggregation across devices. As shown in Figure 14-10, two access switches perform link aggregation negotiation with the connected device as if they were a single device. This raises link reliability from the board level to the device level and forms a dual-active system.

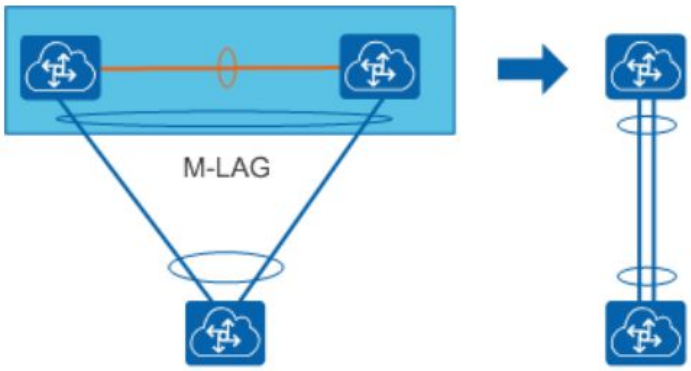


Figure 14-10 MC-LAG Schematic Diagram

MC-LAG is an extension of LACP. MC-LAG is not specified in IEEE 802.1AX; it is a proprietary protocol whose implementation varies by vendor. MC-LAG is compatible with LACP: externally, the two MC-LAG switches use the same LACP protocol information and behave like a single switch.

14.1.2 Typical Applications

14.1.2.1 Dual-Homing Access to an IP Network

The server is dual-homed to the network through MC-LAG, and the gateway is on the MC-LAG devices. Upstream traffic from the server is load-balanced across the two access devices through LACP. Downstream traffic is load-balanced across the two access devices through ECMP and then forwarded locally to the server.

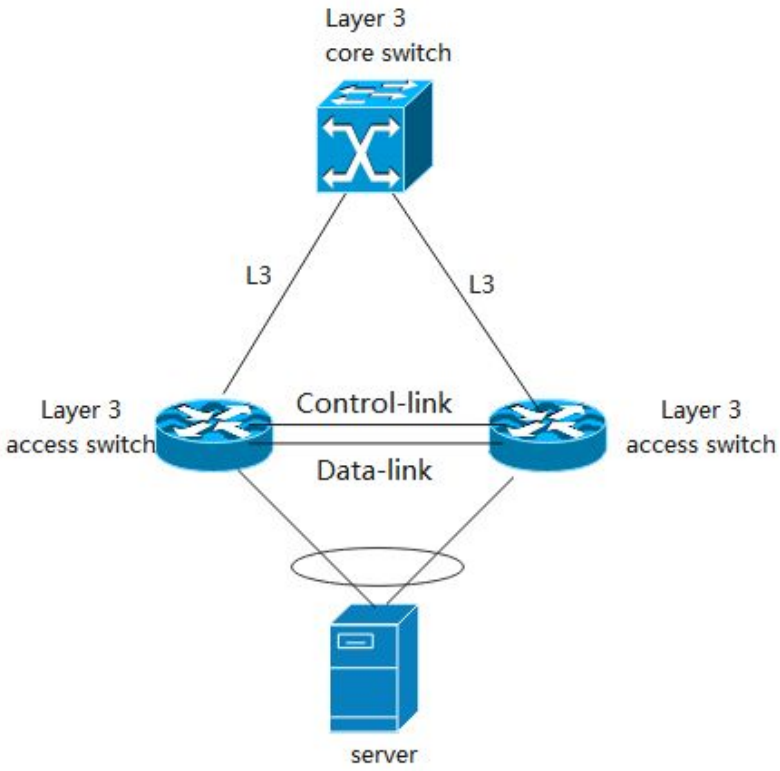


Figure 14-11 MC-LAG Access to an IP Network

14.1.2.2 Dual-Homing Access to a VXLAN Network

The server is dual-homed to the VXLAN network through MC-LAG, and the gateway is on the core device. Upstream traffic from the server is load-balanced across the two access devices through LACP. Downstream traffic is load-balanced across the two access devices through ECMP and then forwarded locally to the server.

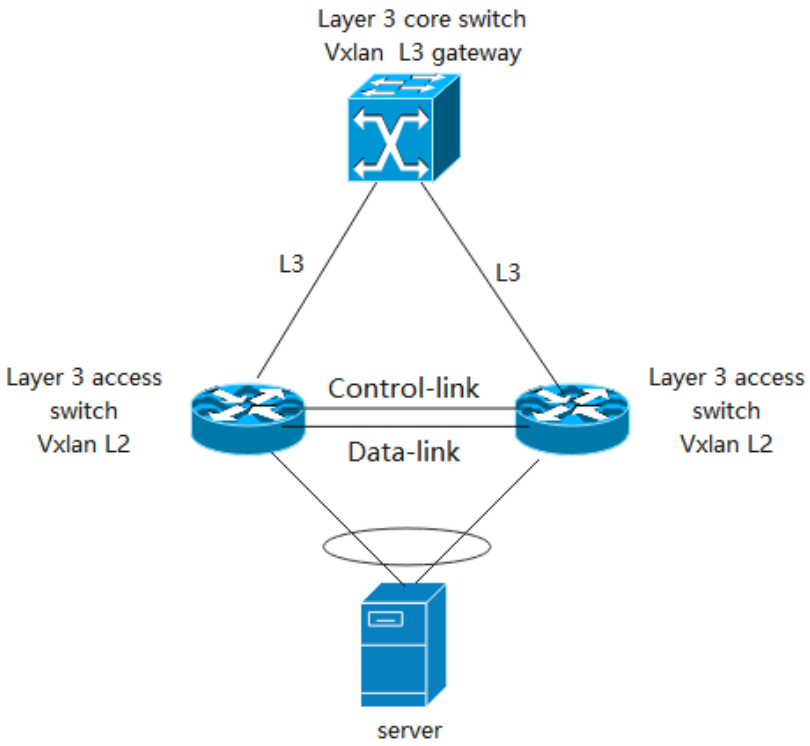


Figure 14-12 MC-LAG Access to a VXLAN Network

14.1.3 MC-LAG Configuration

MC-LAG configuration task list:

2. Configure the MC-LAG data link port
3. Add a port group to the MC-LAG group
4. Configure a VEG group
5. Enter MC-LAG Mode

(1) Configure the MC-LAG domain ID (required)

(2) Configure the MC-LAG domain priority (optional)

(3) Configure the IP address of the Layer 3 interface of the local MC-LAG control link (required)

(4) Configure the IP address of the Layer 3 interface of the peer MC-LAG control link (required)

(5) Associate a VEG group with MC-LAG (required)

(6) Enable MC-LAG (required)

6. Configure the gateway interface

(1) Configure the IP and MAC addresses of the gateway interface

(2) Bind the gateway interface to the VEG group

7. Configure the MAC address of the distributed gateway VTEP (optional)

1. Configure the MC-LAG data link port

Command	Explanation
Port Mode, Port-channel Mode	
[no] switchport mclag data link	Configures the port as the local MC-LAG data link port. The no command removes the local data link port configuration.

2. Add a port group to the MC-LAG group

Command	Explanation
Port-channel Mode	
[no] mclag group	Adds the port group to the MC-LAG group. The no command removes the port group from the MC-LAG group.

3. Configure a VEG group

Command	Explanation
Global Mode	
[no] virtual-equipment-group <ID>	Configures a VEG group. The no command deletes the VEG group.

4. Enter MC-LAG Mode

Command	Explanation
Global Mode	
[no] mclag	Enters MC-LAG Mode. The no command deletes the configuration under MC-LAG Mode and exits MC-LAG Mode.
MC-LAG Mode	
mclag domain-id <1-128> **no mclag domain-id**	Configures the MC-LAG domain ID. The no command deletes the domain ID configuration.
MC-LAG Mode	
mclag priority <1-256> **no mclag priority**	Configures the MC-LAG domain priority. The no command restores the default domain priority.
MC-LAG Mode	
mclag local-ip A.B.C.D **no mclag local-ip**	Configures the IP address of the Layer 3 interface of the local MC-LAG control link. The no command deletes the local IP address configuration.
MC-LAG Mode	
mclag peer-ip A.B.C.D **no mclag peer-ip**	Configures the IP address of the Layer 3 interface of the peer MC-LAG control link. The no command deletes the peer IP address configuration.
MC-LAG Mode	
[no] virtual-equipment-group <ID>	Associates the VEG group with MC-LAG. The no command disassociates the VEG group from MC-LAG.
MC-LAG Mode	
[no] mclag enable	Enables MC-LAG. The no command disables MC-LAG.

5. Configure the gateway interface

Command	Explanation
VLAN Interface Mode, NVI Interface Mode	
[no] ip address A.B.C.D A.B.C.D (secondary |) **[no] ipv6 address X:X::X:X/M** **mac-address XX-XX-XX-XX-XX-XX** **no mac-address**	Configures the IP and MAC addresses of the gateway interface. The no command deletes the IP and MAC addresses of the gateway interface.
VLAN Interface Mode, NVI Interface Mode	
[no] virtual-equipment-group <ID>	Binds the gateway interface to the VEG group. The no command unbinds the gateway interface from the VEG group.

6. Configure the MAC address of the distributed gateway VTEP (optional)

Command	Explanation
Global Mode	
evpn nve mac-address FF-FF-FF-FF-FF-FF **no evpn nve mac-address**	Configures the MAC address of the distributed gateway VTEP. The no command deletes the VTEP MAC address configuration.

14.1.4 Typical MC-LAG Applications

14.1.4.1 Dual-Homing Access to an IP Network

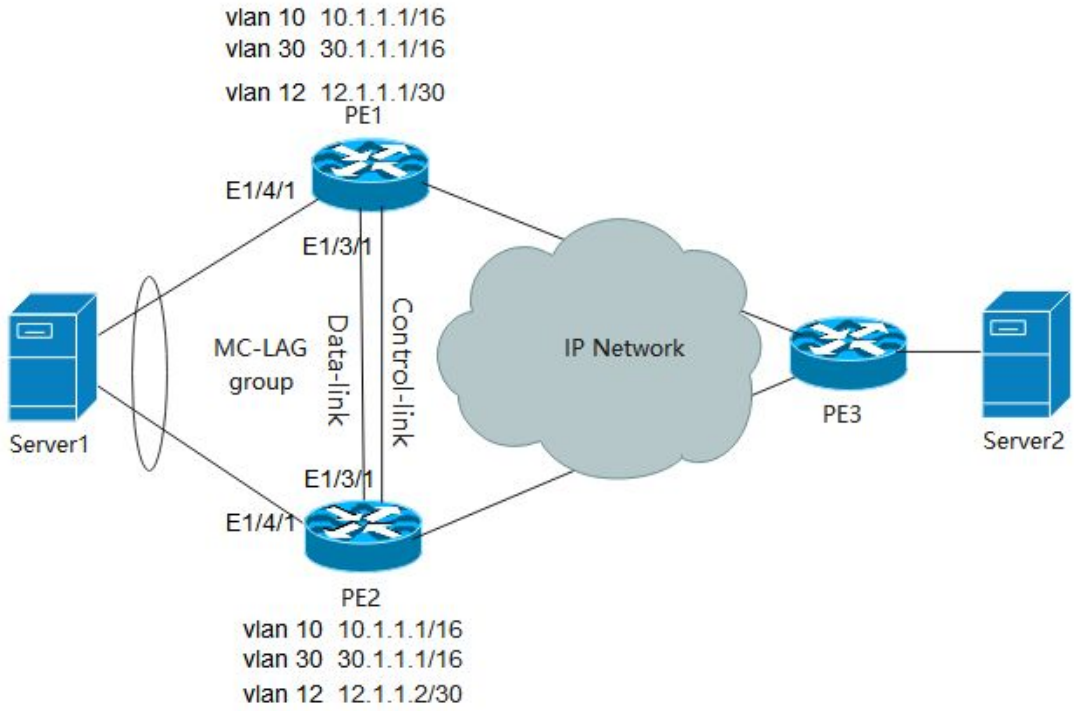


Figure 14-13 Example of MC-LAG Access to an IP Network

As shown in Figure 14-13, this example uses an IP network to illustrate the MC-LAG configuration.

PE1 configuration:

#Configure a VEG group

```
PE1(config)#virtual-equipment-group 1
```

#Configure the MC-LAG data link port

```
PE1(config)#interface ethernet 1/3/1
PE1(config-if-ethernet1/3/1)#switchport mode trunk
PE1(config-if-ethernet1/3/1)#switchport trunk allowed vlan 10;30
PE1(config-if-ethernet1/3/1)#switchport mclag data link
```

#Add the port group to the MC-LAG group

```
PE1(config)#interface ethernet 1/4/1
PE1(config-if-ethernet1/4/1)#switchport mode trunk
PE1(config-if-ethernet1/4/1)#switchport trunk allowed vlan 10;30
PE1(config-if-ethernet1/4/1)#port-group 1 mode active
PE1(config)#interface port-channel 1
PE1(config-if-port-channel1)#mclag group
```

#Enter MC-LAG Mode

```
PE1(config)#mclag
PE1(config-mclag)#mclag domain-id 128
PE1(config-mclag)#mclag local-ip 12.1.1.1
PE1(config-mclag)#mclag peer-ip 12.1.1.2
PE1(config-mclag)#virtual-equipment-group 1
PE1(config-mclag)#mclag enable
```

#Configure the gateway interfaces

```
PE1(config)#interface vlan 10
PE1(config-if-vlan10)#ip address 10.1.1.1 255.255.0.0
PE1(config-if-vlan10)#mac-address 00-00-10-00-00-10
PE1(config-if-vlan10)#virtual-equipment-group 1
PE1(config)#interface vlan 30
PE1(config-if-vlan30)#ip address 30.1.1.1 255.255.0.0
PE1(config-if-vlan30)#mac-address 00-00-30-00-00-30
PE1(config-if-vlan30)#virtual-equipment-group 1
```

PE2 configuration:

#Configure a VEG group

```
PE1(config)#virtual-equipment-group 1
```

#Configure the MC-LAG data link port

```
PE2(config)#interface ethernet 1/3/1
PE2(config-if-ethernet1/3/1)#switchport mode trunk
PE2(config-if-ethernet1/3/1)#switchport trunk allowed vlan 10;30
PE2(config-if-ethernet1/3/1)#switchport mclag data link
```

#Add the port group to the MC-LAG group

```
PE2(config)#interface ethernet 1/4/1
PE2(config-if-ethernet1/4/1)#switchport mode trunk
PE2(config-if-ethernet1/4/1)#switchport trunk allowed vlan 10;30
PE2(config-if-ethernet1/4/1)#port-group 1 mode active
PE2(config)#interface port-channel 1
PE2(config-if-port-channel1)#mclag group
```

#Enter MC-LAG Mode

```
PE2(config)#mclag
PE2(config-mclag)#mclag domain-id 128
PE2(config-mclag)#mclag local-ip 12.1.1.2
PE2(config-mclag)#mclag peer-ip 12.1.1.1
PE2(config-mclag)#virtual-equipment-group 1
PE2(config-mclag)#mclag enable
```

#Configure the gateway interfaces

```
PE2(config)#interface vlan 10
PE2(config-if-vlan10)#ip address 10.1.1.1 255.255.0.0
PE2(config-if-vlan10)#mac-address 00-00-10-00-00-10
PE2(config-if-vlan10)#virtual-equipment-group 1
PE2(config)#interface vlan 30
PE2(config-if-vlan30)#ip address 30.1.1.1 255.255.0.0
PE2(config-if-vlan30)#mac-address 00-00-30-00-00-30
PE2(config-if-vlan30)#virtual-equipment-group 1
```

14.1.4.2 Dual-Homing Access to a Centralized VXLAN Network

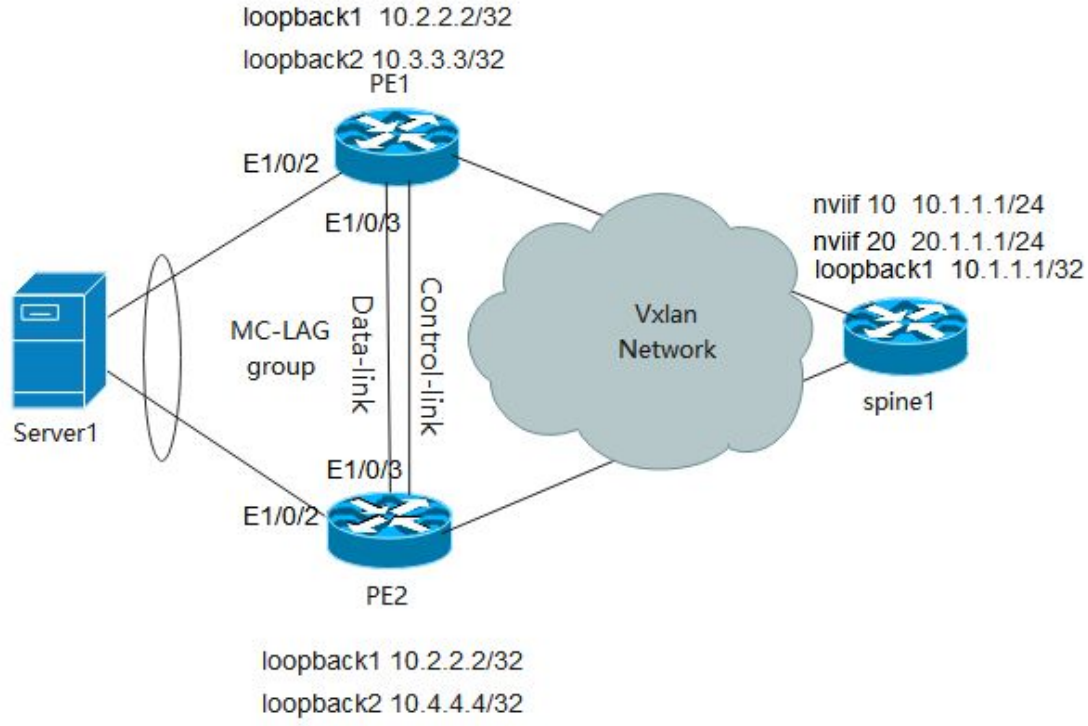


Figure 14-14 Example of MC-LAG Access to a Centralized VXLAN Network

This example shows only the VXLAN- and MC-LAG-related configuration of PE1, PE2, and Spine1. The underlay network configuration, which makes the routes of all underlay network segments reachable, is omitted.

PE1 configuration:

#Configure a VEG group

```
PE1(config)#virtual-equipment-group 1
```

#Configure the MC-LAG domain, control link, and data link, and associate the VEG group

```
PE1(config)#mclag
PE1(config-mclag)#mclag enable
PE1(config-mclag)#mclag domain id 1
PE1(config-mclag)#mclag peer ip address 10.4.4.4
PE1(config-mclag)#mclag local ip address 10.3.3.3
PE1(config-mclag)#virtual-equipment-group 1
PE1(config)#interface ethernet 1/0/3
PE1(config-if-ethernet1/0/3)#switchport mode trunk
PE1(config-if-ethernet1/0/3)#switchport mclag data link
```

#Configure the VXLAN NVI and NVE

```
PE1(config)#nvi 10
PE1(config-nvi)#vxlan-id 10
PE1(config)#nvi 20
PE1(config-nvi)#vxlan-id 20
PE1(config)#interface nve 1
PE1(config-if-nve1)#nve mode vxlan
PE1(config-if-nve1)#source 10.2.2.2
PE1(config-if-nve1)#destination 10.1.1.1
PE1(config-if-nve1)#join nvi 10
PE1(config-if-nve1)#join nvi 20
PE1(config-if-nve1)#description to-Spine
```

#Configure the MC-LAG group and VXLAN mapping

```
PE1(config)#interface ethernet 1/0/2
PE1(config-if-ethernet1/0/2)#port-group 10 mode active
PE1(config)#interface port-channel 10
PE1(config-if-port-channel10)#xconnect nvi 10 mode vlan svid 10
PE1(config-if-port-channel10)#xconnect nvi 20 mode vlan svid 20
PE1(config-if-port-channel10)#mclag group
```

PE2 configuration:

#Configure a VEG group

```
PE2(config)#virtual-equipment-group 1
```

#Configure the MC-LAG domain, control link, and data link, and associate the VEG group

```
PE2(config)#mclag
PE2(config-mclag)#mclag enable
PE2(config-mclag)#mclag domain id 1
PE2(config-mclag)#mclag peer ip address 10.3.3.3
PE2(config-mclag)#mclag local ip address 10.4.4.4
PE1(config-mclag)#virtual-equipment-group 1
PE2(config)#interface ethernet 1/0/3
PE2(config-if-ethernet1/0/3)#switchport mode trunk
PE2(config-if-ethernet1/0/3)#switchport mclag data link
```

#Configure the VXLAN NVI and NVE

```
PE2(config)#nvi 10
PE2(config-nvi)#vxlan-id 10
PE2(config)#nvi 20
PE2(config-nvi)#vxlan-id 20
PE2(config)#interface nve 1
PE2(config-if-nve1)#nve mode vxlan
PE2(config-if-nve1)#source 10.2.2.2
PE2(config-if-nve1)#destination 10.1.1.1
PE2(config-if-nve1)#join nvi 10
PE2(config-if-nve1)#join nvi 20
PE2(config-if-nve1)#description to-Spine
```

#Configure the MC-LAG group and VXLAN mapping

```
PE2(config)#interface ethernet 1/0/2
PE2(config-if-ethernet1/0/2)#port-group 10 mode active
PE2(config)#interface port-channel 10
PE2(config-if-port-channel10)#xconnect nvi 10 mode vlan svid 10
PE2(config-if-port-channel10)#xconnect nvi 20 mode vlan svid 20
PE2(config-if-port-channel10)#mclag group
```

Spine1 configuration:

#Configure the VXLAN NVI, NVE, and gateway

```
Spine1(config)#nvi 10
Spine1(config-nvi)#vxlan-id 10
Spine1(config)#nvi 20
Spine1(config-nvi)#vxlan-id 20
Spine1(config)#interface nve 1
Spine1(config-if-nve1)#nve mode vxlan
Spine1(config-if-nve1)#source 10.1.1.1
Spine1(config-if-nve1)#destination 10.2.2.2
Spine1(config-if-nve1)#join nvi 10
Spine1(config-if-nve1)#join nvi 20
Spine1(config)#interface nvi-interface 10
Spine1(config-if-nviIf10)#ip address 10.1.1.1 255.255.255.0
Spine1(config)#interface nvi-interface 20
Spine1(config-if-nviIf20)#ip address 20.1.1.1 255.255.255.0
```

14.1.4.3 Dual-Homing Access to a Distributed VXLAN Network

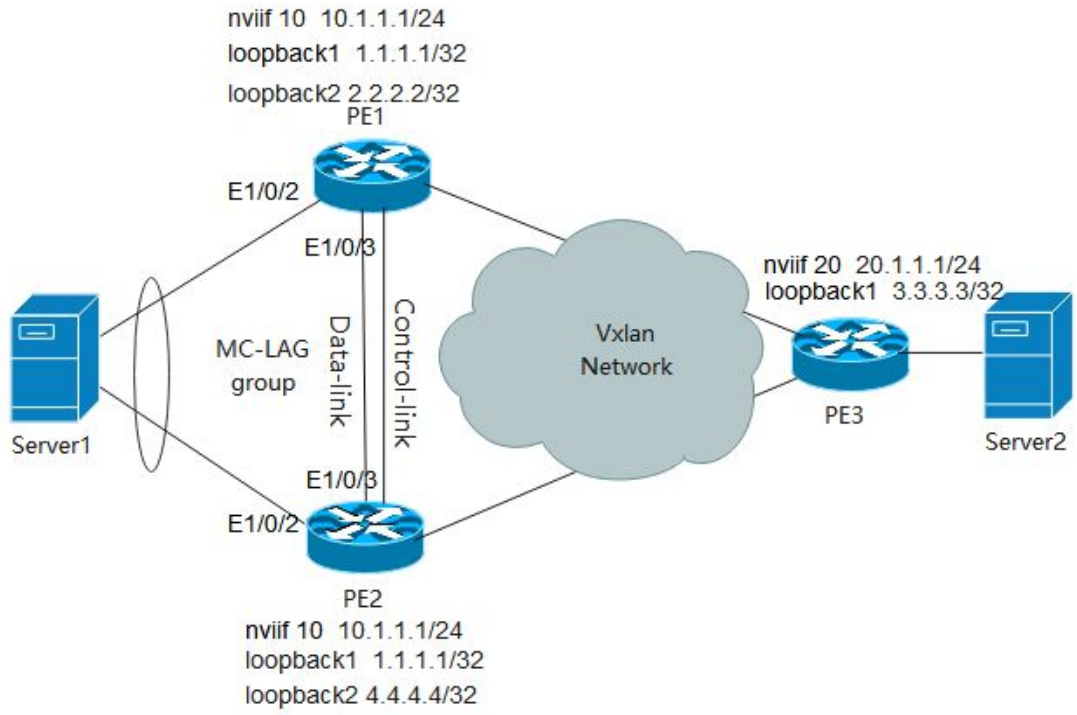


Figure 14-15 Example of MC-LAG Access to a Distributed VXLAN Network

This example uses the VXLAN distributed gateway scenario and shows only the VXLAN- and MC-LAG-related configuration of PE1, PE2, and PE3. The underlay network configuration, which makes the routes of all underlay network segments reachable, is omitted.

PE1 configuration:

#Configure a VEG group

```
PE1(config)#virtual-equipment-group 1
```

#Configure the MC-LAG domain, control link, and data link, and associate the VEG group

```
PE1(config)#mclag
PE1(config-mclag)#mclag enable
PE1(config-mclag)#mclag domain id 1
PE1(config-mclag)#mclag peer ip address 4.4.4.4
PE1(config-mclag)#mclag local ip address 2.2.2.2
PE1(config-mclag)#virtual-equipment-group 1
PE1(config)#interface ethernet 1/0/3
PE1(config-if-ethernet1/0/3)#switchport mode trunk
PE1(config-if-ethernet1/0/3)#switchport mclag data link
```

#Configure VXLAN, EVPN, the gateway, and the VRF

```
PE1(config)#evpn nve source-address 1.1.1.1
PE1(config)#evpn nve mac-address 00-03-0f-22-22-22
PE1(config)#nvi 10
PE1(config-nvi)#vxlan-id 10
PE1(config-nvi)#evpn
PE1(config-nvi-evpn)#rd 1:1
PE1(config-nvi-evpn)#route-target both 100:100
PE1(config-nvi-evpn)#enable
PE1(config)#ip vrf vpn1
PE1(config-vrf)#rd 10:1
PE1(config-vrf)#route-target both 100:1
PE1(config-vrf)#l3-vni 1000
PE1(config)#interface nvi-interface 10
PE1(config-if-nvi-interface10)#ip vrf forwarding vpn1
```

```
PE1(config-if-nvi-interface10)#mac-address 00-00-33-33-99-99
PE1(config-if-nvi-interface10)#distributed-gateway enable
PE1(config-if-nvi-interface10)#ip address 10.1.1.1 255.255.255.0
PE1(config-if-nvi-interface10)#virtual-equipment-group 1
```

#Configure BGP

```
PE1(config)#router bgp 100
PE1(config-router)#neighbor 3.3.3.3 remote-as 100
PE1(config-router)#neighbor 3.3.3.3 update-source 2.2.2.2
PE1(config-router)#neighbor 4.4.4.4 remote-as 100
PE1(config-router)#neighbor 4.4.4.4 update-source 2.2.2.2
PE1(config-router)#address-family l2vpn evpn
PE1(config-router-af)#neighbor 3.3.3.3 activate
PE1(config-router-af)#neighbor 4.4.4.4 activate
PE1(config-router-af)#exit-address-family
```

#Configure the MC-LAG group and VXLAN mapping

```
PE1(config)#interface ethernet 1/0/2
PE1(config-if-ethernet1/0/2)#port-group 10 mode active
PE1(config)#interface port-channel 10
PE1(config-if-port-channel10)#xconnect nvi 10 mode vlan svid 10
PE1(config-if-port-channel10)#mclag group
```

PE2 configuration:

#Configure a VEG group

```
PE2(config)#virtual-equipment-group 1
```

#Configure the MC-LAG domain, control link, and data link, and associate the VEG group

```
PE2(config)#mclag
PE2(config-mclag)#mclag enable
PE2(config-mclag)#mclag domain id 1
PE2(config-mclag)#mclag peer ip address 2.2.2.2
PE2(config-mclag)#mclag local ip address 4.4.4.4
PE2(config-mclag)#virtual-equipment-group 1
PE2(config)#interface ethernet 1/0/3
PE2(config-if-ethernet1/0/3)#switchport mode trunk
PE2(config-if-ethernet1/0/3)#switchport mclag data link
```

#Configure VXLAN, EVPN, the gateway, and the VRF

```
PE2(config)#evpn nve source-address 1.1.1.1
PE2(config)#evpn nve mac-address 00-03-0f-22-22-22
PE2(config)#nvi 10
PE2(config-nvi)#vxlan-id 10
PE2(config-nvi)#evpn
PE2(config-nvi-evpn)#rd 1:1
PE2(config-nvi-evpn)#route-target both 100:100
PE2(config-nvi-evpn)#enable
PE2(config)#ip vrf vpn1
PE2(config-vrf)#rd 10:1
PE2(config-vrf)#route-target both 100:1
PE2(config-vrf)#l3-vni 1000
PE2(config)#interface nvi-interface 10
PE2(config-if-nvi-interface10)#ip vrf forwarding vpn1
PE2(config-if-nvi-interface10)#mac-address 00-00-33-33-99-99
PE2(config-if-nvi-interface10)#distributed-gateway enable
PE2(config-if-nvi-interface10)#ip address 10.1.1.1 255.255.255.0
PE2(config-if-nvi-interface10)#virtual-equipment-group 1
```

#Configure BGP

```
PE2(config)#router bgp 100
PE2(config-router)#neighbor 3.3.3.3 remote-as 100
PE2(config-router)#neighbor 3.3.3.3 update-source 4.4.4.4
```

```
PE2(config-router)#neighbor 2.2.2.2 remote-as 100
PE2(config-router)#neighbor 2.2.2.2 update-source 4.4.4.4
PE2(config-router)#address-family l2vpn evpn
PE2(config-router-af)#neighbor 3.3.3.3 activate
PE2(config-router-af)#neighbor 2.2.2.2 activate
PE2(config-router-af)#exit-address-family
```

#Configure the MC-LAG group and VXLAN mapping

```
PE2(config)#interface ethernet 1/0/2
PE2(config-if-ethernet1/0/2)#port-group 10 mode active
PE2(config)#interface port-channel 10
PE2(config-if-port-channel10)#xconnect nvi 10 mode vlan svid 10
PE2(config-if-port-channel10)#mclag group
```

PE3 configuration:

```
PE3(config)#evpn nve source-address 3.3.3.3
PE3(config)#nvi 20
PE3(config-nvi)#vxlan-id 20
PE3(config-nvi)#evpn
PE3(config-nvi-evpn)#rd 2:1
PE3(config-nvi-evpn)#route-target both 100:100
PE3(config-nvi-evpn)#enable
PE3(config)#ip vrf vpn1
PE3(config-vrf)#rd 20:1
PE3(config-vrf)#route-target both 100:1
PE3(config-vrf)#l3-vni 1000
PE3(config)#interface nvi-interface 20
PE3(config-if-nvi-interface10)#ip vrf forwarding vpn1
PE3(config-if-nvi-interface10)#mac-address 00-00-33-44-99-99
PE3(config-if-nvi-interface10)#distributed-gateway enable
PE3(config-if-nvi-interface10)#ip address 20.1.1.1 255.255.255.0
```

#Configure BGP

```
PE3(config)#router bgp 100
PE3(config-router)#neighbor 4.4.4.4 remote-as 100
PE3(config-router)#neighbor 4.4.4.4 update-source 3.3.3.3
PE3(config-router)#neighbor 2.2.2.2 remote-as 100
PE3(config-router)#neighbor 2.2.2.2 update-source 3.3.3.3
PE3(config-router)#address-family l2vpn evpn
PE3(config-router-af)#neighbor 4.4.4.4 activate
PE3(config-router-af)#neighbor 2.2.2.2 activate
PE3(config-router-af)#exit-address-family
```

14.1.5 MC-LAG Troubleshooting

MC-LAG may fail to operate normally because of physical connection problems, configuration errors, or other causes. Observe the following points:

- ☞ First, make sure that the physical connections are correct.
- ☞ Second, make sure that the interface and the link protocol are up (run the **show interface** command).
- ☞ Then, make sure that all member ports in the same aggregation group have the same negotiation parameters.
- ☞ After the configuration is complete, run the **show mclag** command to display the MC-LAG configuration and status, and the **show mclag group** command to display the MC-LAG group configuration and status.

14.2 NETCONF

14.2.1 Introduction to NETCONF

14.2.1.1 NETCONF Overview

NETCONF (Network Configuration Protocol) is a network configuration and management protocol based on XML (Extensible Markup Language). It uses a simple RPC (Remote Procedure Call) based mechanism for communication between the client and the server.

NETCONF provides a mechanism for managing network devices. Through NETCONF, network devices can provide a complete set of standardized APIs (Application Programming Interfaces), and applications can use these APIs directly to deliver configurations to and obtain configurations from network devices. With this mechanism, you can add, modify, and delete device configurations and obtain the configuration and status information of network devices.

NETCONF has the following advantages:

- NETCONF defines messages in XML format and uses the RPC mechanism to modify configuration information. This simplifies configuration management and enables interoperability between devices from different vendors.
- It reduces network failures caused by manual configuration errors.
- It makes upgrading system software with configuration tools more efficient.
- It is highly extensible: vendors can define their own protocol operations to implement unique management functions.
- It provides security mechanisms such as authentication and authorization to secure message transmission.

14.2.1.2 Main Elements of NETCONF

The basic NETCONF network architecture consists of the following main elements: client, server, and configuration datastore.

1. Client

The client uses NETCONF to manage network devices:

1) It sends requests to the NETCONF server to query or modify one or more specific parameter values.

2) It receives alarms and events sent proactively by the NETCONF server to learn the current state of the managed device.

2. Server

The server maintains the data of the managed device and responds to client requests.

1) After receiving a client request, the server parses the data and returns a response to the client.

2) When a device fault or another event occurs, the server uses the notification mechanism to proactively notify the client of the device alarms and events and to report changes in the current state of the device.

3. Datastore

NETCONF defines one or more configuration datastores and allows configuration operations on them.

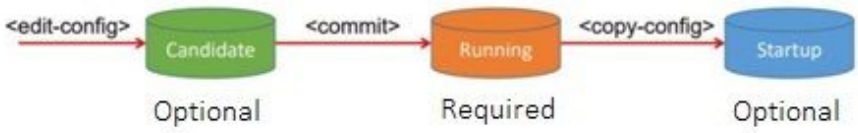


Figure 14-16 NETCONF Configuration Datastores

1) The running datastore holds the configuration currently running on the device; this datastore generally supports persistent operations.

2) NETCONF can support a candidate datastore, which acts as an additional scratch datastore. You can modify the candidate datastore repeatedly until you get the desired result, and then use the commit operation to apply the candidate configuration to the running datastore.

3) NETCONF can also overwrite the startup datastore with the running configuration by using the copy-config operation. <running/> holds the complete configuration currently active on the network device. A device has only one datastore of this type, and it always exists. The base model contains only the <running/> datastore. Additional datastores may be defined by capabilities and are available only on devices that advertise the corresponding capability, for example the <candidate/> datastore.

14.2.1.3 NETCONF Data

NETCONF distinguishes between configuration data and state data and provides different operations to add, delete, modify, and query data.

1) Configuration data: data used to configure network devices.

2) State data: data that reflects the state of the device.

A NETCONF session is the logical connection between the client and the server. NETCONF configures and obtains data through client/server sessions.

1) The information that the client obtains from the running server includes configuration data and state data.

2) The client can modify configuration data; by operating on configuration data, it moves the server to the state the user expects.

3) The client cannot modify state data. State data mainly relates to the running status and statistics of the server.

14.2.2 NETCONF Configuration

NETCONF configuration task list:

1. Enable NETCONF

2. Display NETCONF session information

3. Display information about users connected through NETCONF

14.2.2.1 Enable NETCONF

Command	Explanation
Global Mode	
netconf server enable **no netconf server enable**	Enables NETCONF. The no command disables NETCONF.

14.2.2.2 Display NETCONF Session Information

Command	Explanation
Admin Mode and Global Mode	
show netconf session	Displays NETCONF session information.

14.2.2.3 Display Information About Users Connected Through NETCONF

Command	Explanation
Admin Mode and Global Mode	
show netconf tcp	Displays information about users connected through NETCONF.

14.2.3 NETCONF Example

When network devices need to be managed centrally through a network management system with high security and scalability, NETCONF can be used for communication between the network management system and the devices.

As shown in the figure below, the network management system is deployed on the client. The server acts as an SSH server and accepts connections from the client, which acts as an SSH client, so that configuration can be managed through NETCONF. SSH is an application-layer security protocol that improves the security of NETCONF.

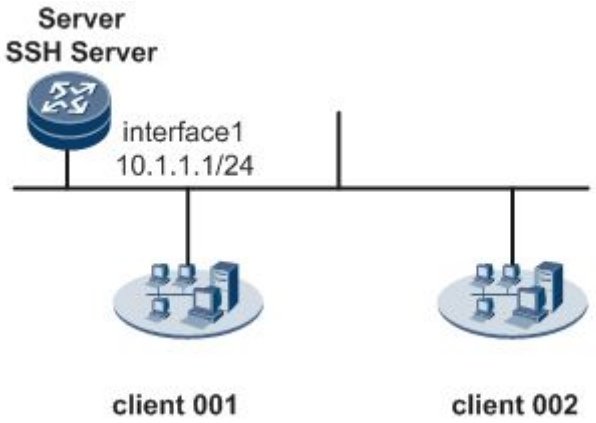


Figure 14-17 NETCONF Network Topology

Prerequisite: The client and the server are reachable over the network.

Steps:

（1）Configure a username and password on the server.

```
Switch(config)# username test privilege 15 password 0 test123
```

（2）Enable NETCONF on the server.

```
Switch(config)# netconf server enable
```

（3）Log in to the server from the client through the network management software.

If the login succeeds, the server prints Netconf: User *<user-name>* login successfully from *<user-address>:<user-port>*

（4）Verify the configuration.

```
Switch(config)# show netconf tcp
LocalAddress  LocalPort  ForeignAddress   ForeignPort  State    IF    VRF
0.0.0.0       830       0.0.0.0          *         LISTEN      0     0
```

```
172.17.100.34 830       172.30.8.120    57924      ESTABLISHED 3527  0
Switch(config)# Show netconf session
session_id:1
transport:netconf-ssh
username:test
source-host:172.30.8.120
login-time:2021-07-20T15:46:40+02:00
in-rpcs:0
in-bad-rpcs:0
out-rpc-errors:0
out-notifications:0
```

（5）Operate on data through the network management software.

```
① <get>/<get-bulk> getting information
```

The client obtains device VLAN interface information (sent XML):

```
<get>
  <filter type="subtree">
    <switch_vlan:interface-vlans
xmlns:switch_vlan="http://example.com/nc/switch_vlan"/>
  </filter>
</get>
```

Result (received XML):

```
<?xml version="1.0" encoding="utf-8"?>
<data xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <interface-vlans xmlns="http://example.com/nc/switch_vlan">
    <interface-vlan>
      <vlan-id>10</vlan-id>
      <ip-address>
        <ipv4-address>
          <address>10.1.1.1</address>
          <prefix>255.255.0.0</prefix>
        </ipv4-address>
        <ipv6-address>
          <address-str>2010:1::1/64</address-str>
        </ipv6-address>
      </ip-address>
    </interface-vlan>
    <interface-vlan>
      <vlan-id>12</vlan-id>
      <ip-address>
        <ipv4-address>
          <address>12.1.1.1</address>
          <prefix>255.255.255.0</prefix>
        </ipv4-address>
        <ipv6-address>
          <address-str>2012::1/64</address-str>
        </ipv6-address>
      </ip-address>
    </interface-vlan>
  </interface-vlans>
</data>
② <get-config>/<get-bulk-config> get configuration information
```

The client obtains device VLAN information (sent XML):

```
<get-config>
  <source>
    <running/>
  </source>
  <filter type="subtree">
    <switch_vlan:vlans
xmlns:switch_vlan="http://example.com/nc/switch_vlan"/>
```

```
  </filter>
</get-config>
```

Result (received XML):

```
<?xml version="1.0" encoding="utf-8"?>
<data xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <vlans xmlns="http://example.com/nc/switch_vlan">
    <vlan>
      <vlan-id>1</vlan-id>
    </vlan>
    <vlan>
      <vlan-id>10</vlan-id>
    </vlan>
    <vlan>
      <vlan-id>12</vlan-id>
    </vlan>
  </vlans>
</data>
③ <edit-config> edit to add specified module data
```

The client adds VLAN 10 (sent XML):

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <vlans xmlns="http://example.com/nc/switch_vlan">
        <vlan>
          <vlan-id>10</vlan-id>
        </vlan>
      </vlans>
    </config>
  </edit-config>
</rpc>
```

Success response (received XML):

```
<?xml version="1.0" encoding="utf-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="12">
<ok/>
</rpc-reply>
④ <edit-config> edit and delete the specified module data
```

The client deletes VLAN 10 (sent XML):

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <vlans xmlns="http://example.com/nc/switch_vlan">
        <vlan nc:operation="delete"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
          <vlan-id>10</vlan-id>
        </vlan>
      </vlans>
    </config>
  </edit-config>
</rpc>
```

Success response (received XML):

```
<?xml version="1.0" encoding="utf-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="12">
  <ok/>
</rpc-reply>
⑤ <rpc> get device status
```

The client sends XML:

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <switch_statis:memory-info
xmlns:switch_statis="http://example.com/nc/switch_statis">
    <switch_statis:get/>
  </switch_statis:memory-info>
</rpc>
```

The client receives the XML result:

```
<?xml version="1.0" encoding="utf-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="8">
  <memory-usage-list xmlns="http://example.com/nc/switch_statis">
    <memory-usage>
      <memory-usage-member-id>1</memory-usage-member-id>
      <memory-usage-slot-id>0</memory-usage-slot-id>
      <total-memory>16384</total-memory>
      <free-memory>13817556992</free-memory>
      <memory-usage-percent>19.57%</memory-usage-percent>
    </memory-usage>
  </memory-usage-list>
</rpc-reply>
```

14.2.4 NETCONF Troubleshooting

(1) To delete a single list entry through XML, add the delete attribute to the list entry header. You cannot delete a list entry by deleting the key value of the current list.

(2) To delete a single container entry or list entry through XML, you do not need to include all leaf nodes of the entry; add the delete operation to the container or list entry header to delete the whole container or list entry and all its data (when deleting a list entry, include the key node).

(3) When multiple YANG modules depend on each other, deliver the XML configuration in the same order as the corresponding command-line configuration.

(4) To configure the device through XML, the device must be in Admin Mode. To configure through XML in other modes, configure the **multi config access** command on the device.

14.3 Telemetry

14.3.1 Telemetry Overview

Telemetry is a remote data collection technology that collects data from physical or virtual devices at high speed. Using a push model, the device periodically and proactively sends information such as interface traffic statistics and CPU and memory data to the collector. Compared with the request-response interaction of the traditional pull model, telemetry provides faster, more real-time data collection.

As SDN networks grow in scale and carry more services, users demand more intelligent operation and maintenance (O&M) of SDN networks, including more precise monitoring data to detect and quickly mitigate traffic microbursts. At the same time, monitoring must have little impact on device functions and performance, so that device and network utilization improve.

Traditional network monitoring methods (such as SNMP get and CLI) have the following shortcomings, become less and less efficient, and cannot keep up with evolving user requirements:

1. They obtain device monitoring data in pull mode, which cannot monitor a large number of network nodes and limits network growth.

2. Their precision is at the minute level. Precision can be improved only by increasing the query frequency, which causes high CPU usage on network nodes and affects normal device operation.

3. Because of network transmission delay, the monitored node data is inaccurate.

Large-scale, high-performance network monitoring therefore requires a new monitoring method.

Telemetry meets these requirements. It enables intelligent O&M systems to manage more devices and to monitor data with higher precision and in closer to real time, with little impact on device functions and performance. It provides the essential big data foundation for quickly locating network problems and optimizing network quality, turns network quality analysis into big data analysis, and strongly supports intelligent O&M.

	Telemetry	SNMP get	SNMP Trap	CLI	SYSLOG
Working mode	Push mode	Pull mode	Push mode	Pull mode	Push mode
Precision	Subsecond	Minutes	Seconds	Minutes	Seconds
Data structure	Defined by YANG models	Defined by MIBs	Defined by MIBs	Unstructured	Unstructured

Multiple monitoring technologies can be deployed simultaneously in a network for comprehensive, multi-angle monitoring. Alternatively, deploy only the technology required to monitor a specific aspect in real time.

After the collected data is sent to the collector through gRPC (Google Remote Procedure Call), the administrator can display it graphically on the collector to understand the network status more clearly and locate network faults quickly. Administrators can also detect potential network problems and correct them in time to avoid network failures.

The typical telemetry network model, shown in the figure below, includes the following components:

Switch: acts as the static subscription client and the dynamic subscription server in the network.

Collector and analyzer: acts as the static subscription server and the dynamic subscription client in the network.

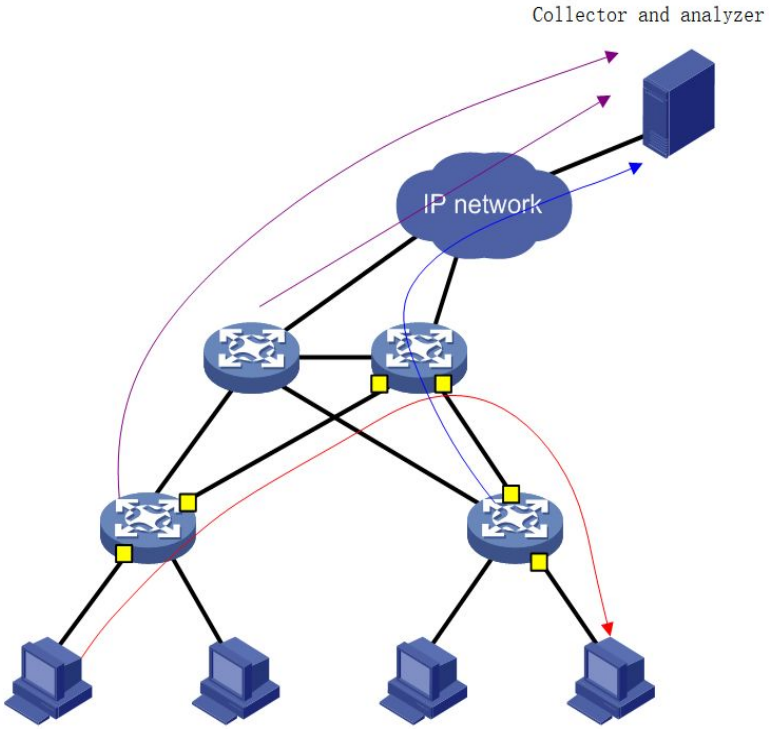


Figure 14-19 Typical Telemetry Network Model

14.3.2 Telemetry Configuration

Telemetry configuration task list:

8. Configure a destination group for static subscription (required)
9. Configure a sensor group for static subscription (required)
10. Configure a subscription group for static subscription (required)
11. Configure the gRPC server for dynamic subscription (required)

Enter Telemetry Mode (required)

Command	Explanation
Global Mode	
telemetry exit	Enters Telemetry Mode. Exits Telemetry Mode.

Configure a destination group (required)

Command	Explanation
Telemetry Mode	
destination-group WORD no destination-group WORD	Configures the destination group name and enters Destination-group Mode. The no command deletes the destination group and its configuration.
Destination-group Mode	
ipv4-address x.x.x.x port <1-65535> no ipv4-address x.x.x.x port <1-65535>	Configures the IP address and TCP port of the target collector in the destination group of a static telemetry subscription. The no command deletes the IP address and TCP port of the target collector from the destination group of a static telemetry subscription.

Configure a sensor group (required)

Command	Explanation
Telemetry Mode	
sensor-group WORD no sensor-group WORD	Configures the sensor group name and enters Sensor-group Mode. The no command deletes the sensor group and its configuration.
Sensor-group Mode	
sensor-path memoryInfo (| (op-type (eq | ge | gt | le | lt) op-value <1-100> (| ((and | or) op-type (eq | ge | gt | le | lt) op-value <1-100>)))) no sensor-path memoryInfo	Configures the memory sampling path and its conditions in the sensor group of a static telemetry subscription. The no command deletes the memory sampling path from the sensor group of a static telemetry subscription.
sensor-path cpuInfo (| (op-type (eq | ge | gt | le | lt) op-value <1-100> (| ((and | or) op-type (eq | ge | gt | le | lt) op-value <1-100>)))) no sensor-path cpuInfo	Configures the CPU sampling path and its conditions in the sensor group of a static telemetry subscription. The no command deletes the CPU sampling path from the sensor group of a static telemetry subscription.
sensor-path ifStatistics ((ethernet | port-channel) IFNAME|) no sensor-path ifStatistics ((ethernet | port-channel) IFNAME|)	Configures the interface statistics sampling path in the sensor group of a static telemetry subscription. The no command deletes the interface statistics sampling path from the sensor group of a static telemetry subscription.

Configure a subscription group (required)

Command	Explanation
Telemetry Mode	
subscription-group WORD no subscription-group WORD	Configures the name of the subscription group of a static telemetry subscription and enters Subscription-group Mode. The no command deletes the subscription group of a static telemetry subscription and its configuration.
Subscription-group Mode	
destination-group WORD no destination-group WORD	Configures the destination group referenced in the subscription group. The no command deletes the destination group reference from the subscription group.
sensor-group WORD sample-interval <10000-1800000> no sensor-group WORD	Configures the sensor group bound to the subscription group of a static telemetry subscription and its reporting interval. The no command deletes the sensor group binding and reporting interval from the subscription group of a static telemetry subscription.

Configure the gRPC server for dynamic subscription (required)

Command	Explanation
Telemetry Mode	

grpc server x.x.x.x port <10000-57999> no grpc server	Configures the address and TCP listening port of the gRPC server for dynamic telemetry subscription. The no command deletes the address and TCP listening port of the gRPC server for dynamic telemetry subscription.

Configure the TUN interface VLAN if a service port is used as the management port (optional)

Command	Explanation
Global Mode	
tun interface vlan <vlan-id | WORD> no tun interface vlan <vlan-id | WORD>	Enables the TUN function on the VLAN interface. The no command disables the TUN function on the VLAN interface.

Display the static and dynamic telemetry subscription configuration (optional)

Command	Explanation
Admin Mode	
show telemetry subscription	Displays all static telemetry subscription configuration.
show telemetry dynamic-subscription	Displays all dynamic telemetry subscription configuration.

14.3.3 Typical Telemetry Application

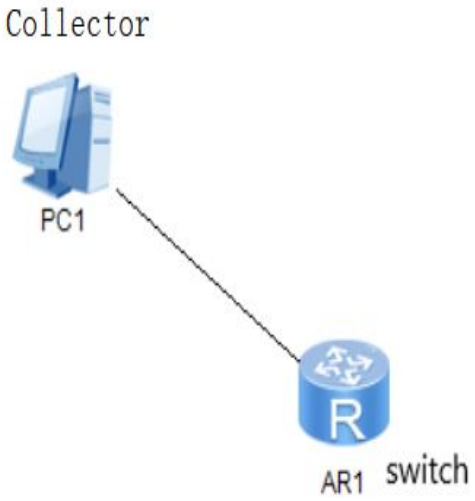


Figure 14-20 Typical Telemetry Network

Configuration steps:

1. Configure static subscription on the switch as shown in the figure above.

```
Switch(config)# telemetry
Switch(config-telemetry)# destination-group destination1
Switch(config-telemetry-dest-group-destination1)#  ipv4-address 10.20.2.1
   port 10001
Switch(config-telemetry-dest-group-destination1)#  exit
Switch(config-telemetry)#sensor-group sensor1
Switch(config-telemetry-sen-group-sensor1)#sensor-path cpuInfo    op-type
   ge op-value 1 or op-type ge op-value 19
```

```
Switch(config-telemetry-sen-group-sensor1)#sensor-path memoryInfo    op-type
   ge op-value 1 or op-type ge op-value 19
Switch(config-telemetry-sen-group-sensor1)#sensor-path ifStatistics
Switch(config-telemetry-sen-group-sensor1)# exit
Switch(config-telemetry)#subscription subscription1
Switch(config-telemetry-sub-group-subscription1)# sensor-group sensor1
   sample-interval 10000
Switch(config-telemetry-sub-group-subscription1)# destination-group
   destination1
Switch(config-telemetry-sub-group-subscription1)# exit
```

This completes the static subscription configuration. Before configuring it, make sure that the device and the collector can reach each other over the network.

2. Configure dynamic subscription on the switch as shown in the figure above.

```
Switch(config)#telemetry
Switch(config-telemetry)#grpc server 172.17.100.235 port 50002
```

This completes the dynamic subscription configuration. Before configuring it, make sure that the device and the collector can reach each other over the network.

14.4 VXLAN and EVPN

VXLAN (Virtual eXtensible Local Area Network) encapsulates Layer 2 frames in a Layer 3 protocol, extending Layer 2 networks across a Layer 3 network. It is used in data centers to let virtual machines migrate within an interconnected Layer 3 network without changing their IP and MAC addresses, ensuring service continuity. VXLAN uses a 24-bit network identifier, which allows up to 16M isolated virtual networks. This overcomes the 4K limit of the widely used VLAN and gives large-scale multi-tenant cloud environments enough virtual network partitions.

To address the problems of MPLS L2VPN, EVPN, which extends MP-BGP, provides the control plane that VXLAN lacks. EVPN for MPLS L2VPN is defined in RFC 7432, and its extension for VXLAN is defined in RFC 8365.

Using EVPN as the control plane has the following advantages:

- It enables automatic VTEP discovery and automatic VXLAN tunnel establishment, which simplifies network deployment and expansion.
- It advertises Layer 2 MAC information and Layer 3 route information at the same time.
- It reduces flooding traffic in the network.

14.4.1 VXLAN and EVPN Overview

14.4.1.1 VXLAN Overview

VXLAN has the following characteristics:

Supports a large number of tenants: VXLAN uses a 24-bit network identifier, which greatly increases the number of supported tenants and solves the VLAN shortage of traditional Layer 2 networks.

Easy to maintain: VXLAN builds a large Layer 2 network over an IP network, which simplifies deployment and maintenance and makes full use of existing IP network technologies, such as ECMP routing for load balancing. Only the edge devices of the IP core network perform VXLAN processing; intermediate devices forward packets based only on the IP header, which reduces deployment complexity and cost.

Saves MAC address table entries on the switch: The access switch learns only the MAC addresses of physical servers, not the MAC address of each virtual machine, which greatly saves MAC table space and improves switching performance.

The switch supports VXLAN only over IPv4 networks.

The typical VXLAN network model, shown in the figure below, includes the following components:

VM (Virtual Machine): Multiple VMs can be created on one server, and different VMs can belong to different VXLANs. VMs in the same VXLAN are in the same logical Layer 2 network and can communicate with each other. VMs in different VXLANs are isolated from each other at Layer 2. A VXLAN is identified by its VXLAN ID, also called VNI (VXLAN Network Identifier, 24 bits long).

VTEP (VXLAN Tunnel End Point): A VXLAN edge device. The VTEP performs VXLAN processing, such as identifying the VXLAN to which an Ethernet frame belongs, forwarding frames at Layer 2 based on VXLAN, and encapsulating and decapsulating packets. A VTEP can be a standalone physical device or the server that hosts the VMs.

VXLAN tunnel: A point-to-point logical tunnel between two VTEPs. The VTEP adds the VXLAN header, UDP header, and IP header to the frame and forwards the encapsulated packet through the VXLAN tunnel to the remote VTEP, which decapsulates it.

Core device: A device in the IP core network (such as the P device in the figure below). The core device does not perform VXLAN processing; it only forwards the encapsulated packet at Layer 3 based on its destination IP address.

NVI (Network Virtual Instance): A virtual network instance on a VTEP that provides Layer 2 switching services for a VXLAN network. An NVI can be regarded as a VXLAN-based virtual switch that performs Layer 2 forwarding on the VTEP. It has all the functions of a traditional Ethernet switch, including source MAC address learning, MAC address aging, and flooding. NVIs and VXLANs have a one-to-one mapping.

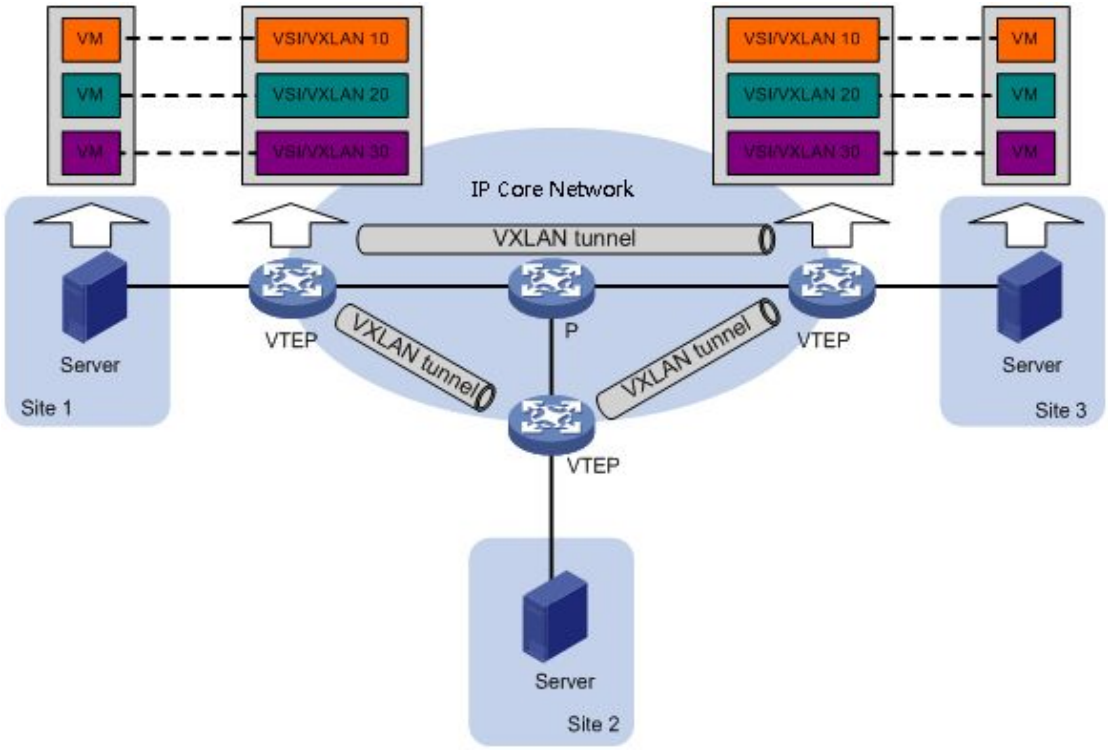


Figure 14-1 Typical VXLAN Network Model

14.4.1.2 VXLAN Packet Encapsulation

VXLAN packet encapsulation format: An 8-byte VXLAN header, an 8-byte UDP header, and a 20-byte IP header are added to the original Layer 2 frame. The destination port number in the UDP header is the VXLAN UDP port number (default 4789).

The VXLAN header consists mainly of the following two parts:

Flags: When the I bit is 1, the VXLAN ID in the VXLAN header is valid; when it is 0, the VXLAN ID is invalid. The other bits are reserved and set to 0.

VXLAN ID: Identifies a VXLAN network; it is 24 bits long.

The figure below shows the VXLAN packet encapsulation format:

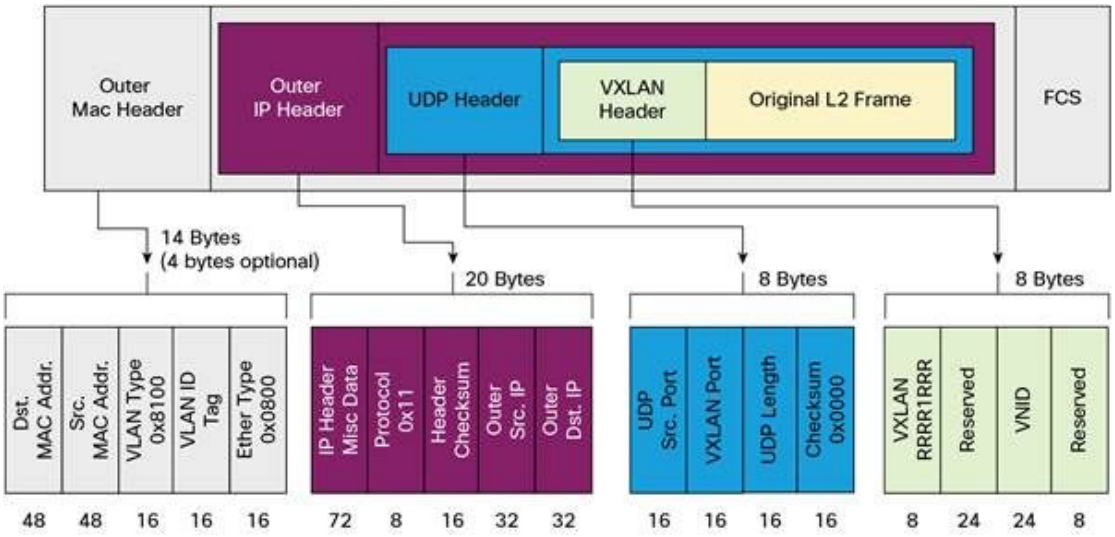


Figure 14-2 VXLAN Packet Encapsulation Format

14.4.1.3 EVPN Overview

EVPN (Ethernet Virtual Private Network) is a Layer 2 VPN technology. Its control plane uses MP-BGP to advertise EVPN route information, and its data plane uses VXLAN or MPLS encapsulation to forward packets. When a tenant's sites are in different physical locations, EVPN can provide Layer 2 interconnection for the same subnet of the same tenant over an existing service provider or enterprise IP network.

As the VXLAN control plane, EVPN first establishes EVPN peers between the VTEPs (VXLAN Tunnel End Points) at both ends. The peers then exchange VNI (VXLAN Network Identifier) and VTEP IP address information through EVPN routes, and the VXLAN tunnel is established based on this information.

To support EVPN, MP-BGP defines a new subaddress family, the EVPN address family, under the L2VPN address family, and adds the following EVPN NLRI (Network Layer Reachability Information), that is, EVPN routes:

(1) Ethernet Auto-discovery Route (RT-1): Advertises ES information in a multihomed site network to implement features such as split horizon, aliasing, and active/standby backup.

(2) MAC/IP Advertisement Route (RT-2): Advertises MAC/IP address information.

(3) Inclusive Multicast Ethernet Tag Route (RT-3): Advertises VTEPs and their VXLAN information to enable automatic VTEP discovery, automatic VXLAN tunnel establishment, automatic creation of VXLAN broadcast tables, and so on.

(4) Ethernet Segment Route (RT-4): Advertises ES and connected VTEP information to discover other members of the VTEP redundancy group connected to the same ES and to elect the designated forwarder in the redundancy group.

(5) IP Prefix Advertisement Route (RT-5): Advertises imported external routes as IP prefixes.

Currently, the most widely used EVPN solution uses the MP-BGP extension defined in RFC 7432 as the control plane and VXLAN encapsulation and forwarding as the data plane. EVPN inherits the advantages of MP-BGP and VXLAN and has the following characteristics:

(1) Simplified configuration: MP-BGP enables automatic VTEP discovery, automatic VXLAN tunnel establishment, and automatic association of VXLAN tunnels with VXLANs, without manual configuration, which simplifies network deployment.

(2) Separate control and data planes: The control plane advertises route information, and the data plane forwards packets. This clear division of labor simplifies management.

(3) Multihoming support: When a site connects to the VXLAN network through multiple VTEPs, all paths to the site can forward traffic, which improves bandwidth utilization.

(4) Symmetric IRB (Integrated Routing and Bridging) support: MP-BGP advertises Layer 2 MAC addresses and Layer 3 route information at the same time, so a VTEP can perform both Layer 2 forwarding and Layer 3 routing. This ensures that traffic is forwarded along the optimal path and reduces broadcast traffic.

14.4.2 VXLAN and EVPN Configuration

14.4.2.1 VXLAN Configuration

VXLAN configuration task list:

Configure a network virtual instance globally (required)

Configure the VXLAN ID in NVI Mode (required)

Configure an NVE tunnel interface globally (required)

Configure VXLAN access mapping on the access port (required)

Display detailed network virtual instance information and sent/received packet statistics (optional)

Configure a network virtual instance globally (required)

Command	Explanation
Global Mode	
nvi <*nvi-id*> **no nvi <*nvi-id*>**	Creates a network virtual instance or enters NVI Mode. The no command deletes the network virtual instance.

Configure the VXLAN ID in NVI Mode (required)

Command	Explanation
NVI Mode	
vxlan-id <*vxlan-id*>	Sets the VXLAN ID and associates it with the network virtual instance.

Configure an NVE tunnel interface globally (required)

Command	Explanation
Global Mode	
interface nve <*nve-id*> **no interface nve <*nve-id*>**	Creates a VXLAN tunnel interface or enters NVE Interface Mode. The no command deletes the VXLAN tunnel interface.
NVE Interface Mode	
nve mode vxlan	Sets the tunnel mode to VXLAN.
source {tunnel-source-ip|word} **no source {tunnel-source-ip|word}**	Configures the IPv4 address of the source VTEP. The no command deletes the IPv4 address of the source VTEP.
destination {tunnel-destination-ip|word} **no destination {tunnel-destination-ip|word}**	Configures the IPv4 address of the destination VTEP. The no command deletes the IPv4 address of the destination VTEP.
join nvi <*nvi-id* **no join nvi <*nvi-id*>**	Associates the network virtual instance with the VXLAN tunnel. The no command removes the association.

description <*word*> **no description**	Configures the description of the network virtual instance or NVE tunnel interface. The no command deletes the description.

Configure VXLAN access mapping on the access port (required)

Command	Explanation
Port Mode	
xconnect nvi <*nvi-id*> {|mode {ethernet|vlan svid<*vid*>}} **no xconnect nvi <*nvi-id*> {|mode {ethernet|vlan svid<*vid*>}}**	Configures VXLAN access mapping on the access port. The no command deletes the VXLAN access mapping.

Display detailed network virtual instance information and sent/received packet statistics (optional)

Command	Explanation
Admin Mode	
show interface nve <*nve-id*>	Displays the numbers of packets and bytes sent and received on the NVE tunnel interface.
show nvi <*nvi-id*> detail	Displays detailed information about the network virtual instance.

14.4.2.2 EVPN Configuration

EVPN configuration task list:

1. Configure the global EVPN NVE source address (required)
2. Configure the designated forwarder (DF) election delay (optional)
3. Configure the EVPN NVI-VLAN mapping monitor (optional)
4. Configure automatic learning of remote MAC/ARP/ND entries (optional)
5. Configure the service loopback group referenced by VXLAN (required)
6. Configure a virtual equipment group (optional)
7. Configure a VXLAN broadcast domain (required)

(1) Create a virtual switch instance (required)

(2) Configure the VXLAN ID of the NVI instance (required)

(3) Configure ARP/ND suppression (optional)

(4) Configure ARP/ND proxy answer (optional)

8. Configure an EVPN instance (required)

(1) Create an EVPN instance associated with the NVI (required)

(2) Configure the route distinguisher of the EVPN instance (required)

(3) Configure the route target of the EVPN instance (required)

(4) Enable the EVPN instance associated with the NVI (required)

9. Configure service access (required)

(1) Configure the ESI of the port (optional)

(2) Configure the NVI access mapping mode of the port (optional)

10. Configure an L3VPN instance (optional)

（1）Configure the L3 VNI associated with the Layer 3 VXLAN tunnel (optional)

11. Configure the gateway interface of the virtual switch instance (optional)

(1) Create a gateway interface for the virtual switch instance (optional)

(2) Associate the gateway interface with the L3VPN instance (optional)

(3) Configure the IP and MAC addresses of the gateway interface (optional)

(4) Configure the gateway interface as a distributed gateway interface (optional)

(5) Bind the virtual equipment group (optional)

12. Configure BGP (required)

(1) Configure BGP neighbors (required)

(2) Enter EVPN address family configuration mode (required)

(3) Advertise the EVPN capability to BGP peers (required)

(4) Configure a route reflector client (optional)

(5) Exit EVPN address family configuration mode (required)

(6) Enter BGP VPN address family configuration mode (optional)

(7) Configure redistribution of routes from other sources into BGP (optional)

(8) Exit BGP VPN address family configuration mode (optional)

1. **Configure the global EVPN NVE source address**

The NVE source address must be the address of a local Layer 3 interface.

Command	Explanation
Global Mode	
evpn nve source-address <ip-address> **no evpn nve source-address**	Configures the global EVPN NVE source address. The no command deletes the global EVPN NVE source address.

2. **Configure the designated forwarder (DF) election delay**

In a multihomed access network, a PE starts the DF election delay timer after sending a Type 4 route, and performs DF election when the timer expires.

Command	Explanation
Global Mode	
evpn timer df-delay <delay-value> **no evpn timer df-delay**	Configures the designated forwarder election delay. The no command restores the default value.

3. **Configure the EVPN NVI-VLAN mapping monitor**

Command	Explanation
Global Mode	
[no] evpn nvi-vlan-mapping-monitor disable	Disables the mapping of ingress VLANs to ETI IDs. The no command enables the mapping of ingress VLANs to ETI IDs.

4. **Configure automatic learning of remote MAC/ARP/ND entries**

Command	Explanation
Global Mode	
[no] vxlan remote mac-address-learning disable	Disables automatic learning of VXLAN remote MAC addresses. The no command enables automatic learning of VXLAN remote MAC addresses.
[no] vxlan remote arp-learning disable	Disables automatic learning of VXLAN remote ARP entries. The no command enables automatic learning of VXLAN remote ARP entries.
[no] vxlan remote nd-learning disable	Disables automatic learning of VXLAN remote ND entries. The no command enables automatic learning of VXLAN remote ND entries.

5. **Configure a virtual equipment group**

Command	Explanation
Global Mode	
[no] virtual-equipment-group <1-1>	Configures the virtual equipment group. The no command deletes the virtual equipment group.
VEG Mode	
source ip A.B.C.D	Configures the local IP address of the virtual equipment group.
VEG Mode	
[no] remote ip A.B.C.D	Configures the remote IP address of the virtual equipment group. The no command deletes the remote IP address of the virtual equipment group.

6. **Configure a VXLAN broadcast domain**

In VXLAN, a VNI identifies a broadcast domain.

Command	Explanation
Global Mode	
[no] nvi <nvi-id>	Creates a virtual switch instance. The no command deletes the virtual switch instance.
NVI Mode	
vxlan-id <vxlan-id>	Configures the VXLAN ID of the NVI instance.
NVI Mode	
[no] arp suppression enable **[no] nd suppression enable**	Enables ARP/ND suppression. The no command disables ARP/ND suppression.
NVI Mode	
[no] arp proxy-answer enable **[no] nd proxy-answer enable**	Enables ARP/ND proxy answer. The no command disables ARP/ND proxy answer.

7. **Configure an EVPN instance**

Command	Explanation
NVI Mode	
[no] evpn	Creates an EVPN instance associated with the NVI. The no command deletes the EVPN instance associated with the NVI.
EVI Mode	
rd <ASN:nn_or_IP-address:nn> **no rd**	Configures the route distinguisher of the EVPN instance. The no command deletes the route distinguisher of the EVPN instance.
EVI Mode	
[no] route-target {import | export | both} <rt-value>	Configures the route target of the EVPN instance. The no command deletes the route target of the EVPN instance.
EVI Mode	
enable **disable**	The enable command enables the EVPN instance of the NVI; the disable command disables it.

8. **Configure service access**

Command	Explanation
Port Mode	
esi <esi-id> **no esi**	Configures the ESI of the port. The no command deletes the ESI of the port.
Port Mode	
[no] xconnect nvi <nvi-id> ((mode (ethernet | (vlan svid <1-4094>)))|)	Configures the NVI access mapping mode of the port. The no command deletes the NVI access mapping mode of the port.

9. **Configure an L3VPN instance**

Command	Explanation
VRF Mode	
l3-vni <vxlan-id> **no l3-vni**	Configures the L3 VNI associated with the Layer 3 VXLAN tunnel. The no command deletes the L3 VNI associated with the Layer 3 VXLAN tunnel.

10. **Configure the gateway interface of the virtual switch instance**

Command	Explanation
Global Mode	
[no] interface nvi-interface <nvi-id>	Creates a gateway interface for the virtual switch instance. The no command deletes the gateway interface.
NVI Interface Mode	
[no] ip vrf forwarding <vrf-name>	Associates the gateway interface with the L3VPN instance. The no command removes the association between the gateway interface and the L3VPN instance.
NVI Interface Mode	
[no] ip address A.B.C.D A.B.C.D (secondary |) **[no] ipv6 address X:X::X:X/M** **mac-address XX-XX-XX-XX-XX-XX** **no mac-address**	Configures the IP and MAC addresses of the gateway interface. The no command deletes the IP and MAC addresses of the gateway interface.
NVI Interface Mode	
[no] distributed-gateway enable	Configures the gateway interface as a distributed gateway interface. The no command restores the default centralized gateway interface.
NVI Interface Mode	
[no] virtual-equipment-group <1-1>	Binds the virtual equipment group. The no command unbinds the virtual equipment group.

11. **Configure BGP**

Command	Explanation
BGP Router Mode	
[no] neighbor {<ip-address>|<TAG>} **remote-as <as-id>**	Configures a BGP neighbor. The no command deletes the BGP neighbor.
BGP Router Mode	
address-family l2vpn evpn	Enters EVPN address family configuration mode.
EVPN Address Family Mode	

[no] neighbor {<ip-address>|<TAG>} activate	Exchanges EVPN address family routes with the BGP neighbor. The no command stops exchanging EVPN address family routes.
EVPN Address Family Mode	
[no] neighbor {<ip-address>|<TAG>} route-reflector-client	Configures a route reflector client. The no command deletes the route reflector client.
EVPN Address Family Mode	
exit-address-family	Exits EVPN address family configuration mode.
BGP Router Mode	
address-family ipv4 vrf <vrf-name> **address-family ipv6 vrf <vrf-name>**	Enters BGP VPN address family configuration mode.
BGP VPN Address Family Mode	
[no] redistribute <ROUTES> [route-map <WORD>]	Configures redistribution of routes from other sources into BGP. The no command deletes the route redistribution configuration.
BGP VPN Address Family Mode	
exit-address-family	Exits BGP VPN address family configuration mode.

14.4.3 VXLAN Examples

14.4.3.1 Static VXLAN

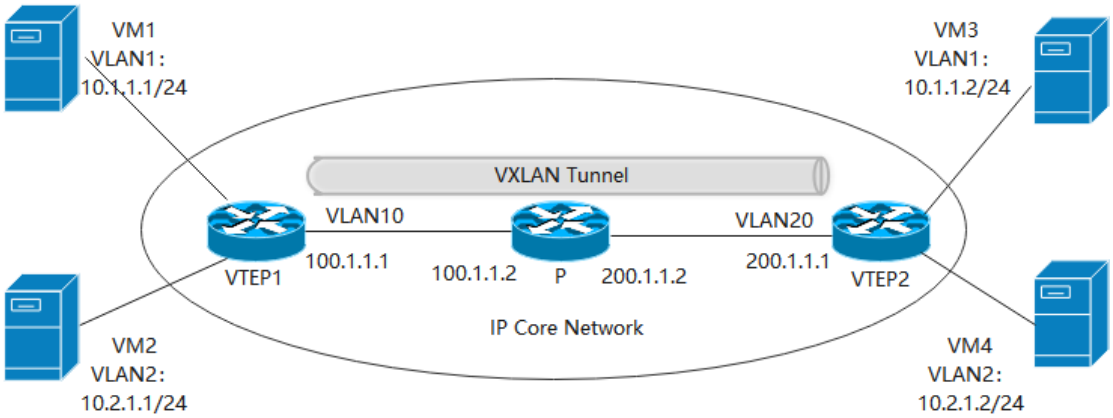


Figure 14-3 Static VXLAN Network

Configuration steps:

Only the VXLAN-related configuration is shown; the rest of the configuration is omitted.

(1) VTEP1 configuration:

#Configure the network virtual instances

```
VTEP1(config)#nvi 1
VTEP1(config-nvi)#vxlan-id 1
VTEP1(config-nvi)#description VM1
VTEP1(config)#nvi 2
VTEP1(config-nvi)#vxlan-id 2
VTEP1(config-nvi)#description VM2
```

#Configure the VXLAN tunnel interface

```
VTEP1(config)#interface nve 1
```

```
VTEP1(config-if-nve1)#nve mode vxlan
VTEP1(config-if-nve1)#source 100.1.1.1
VTEP1(config-if-nve1)#destination 200.1.1.1
VTEP1(config-if-nve1)#join nvi 1
VTEP1(config-if-nve1)#join nvi 2
VTEP1(config-if-nve1)#description VTEP1
```

#Configure VXLAN access mapping on the access ports

```
VTEP1(config)#interface ethernet 1/0/1
VTEP1(config-if-ethernet1/0/1)#switchport mode access
VTEP1(config-if-ethernet1/0/1)#description VM1
VTEP1(config-if-ethernet1/0/1)#xconnect nvi 1 mode vlan svid 1
VTEP1(config)#interface ethernet 1/0/2
VTEP1(config-if-ethernet1/0/2)#switchport mode access
VTEP1(config-if-ethernet1/0/1)#description VM2
VTEP1(config-if-ethernet1/0/2)#xconnect nvi 2 mode vlan svid 2
```

(2) VTEP2 configuration:

#Configure the network virtual instances

```
VTEP2(config)#nvi 1
VTEP2(config-nvi)#vxlan-id 1
VTEP2(config-nvi)#description VM3
VTEP2(config)#nvi 2
VTEP2(config-nvi)#vxlan-id 2
VTEP2(config-nvi)#description VM4
```

#Configure the VXLAN tunnel interface

```
VTEP2(config)#interface nve 1
VTEP2(config-if-nve1)#nve mode vxlan
VTEP2(config-if-nve1)#source 200.1.1.1
VTEP2(config-if-nve1)#destination 100.1.1.1
VTEP2(config-if-nve1)#join nvi 1
VTEP2(config-if-nve1)#join nvi 2
VTEP2(config-if-nve1)#description VTEP2
```

#Configure VXLAN access mapping on the access ports

```
VTEP2(config)#interface ethernet 1/0/1
VTEP2(config-if-ethernet1/0/1)#switchport mode access
VTEP2(config-if-ethernet1/0/1)#description VM3
VTEP2(config-if-ethernet1/0/1)#xconnect nvi 1 mode vlan svid 1
VTEP2(config)#interface ethernet 1/0/2
VTEP2(config-if-ethernet1/0/2)#switchport mode access
VTEP2(config-if-ethernet1/0/1)#description VM4
VTEP2(config-if-ethernet1/0/2)#xconnect nvi 2 mode vlan svid 2
```

With VXLAN access mapping configured on the access ports, VM1 and VM3 can communicate with each other, and VM2 and VM4 can communicate with each other, while VM1 and VM2, and VM3 and VM4, are isolated from each other.

14.4.3.2 EVPN VXLAN Layer 2 Interconnection

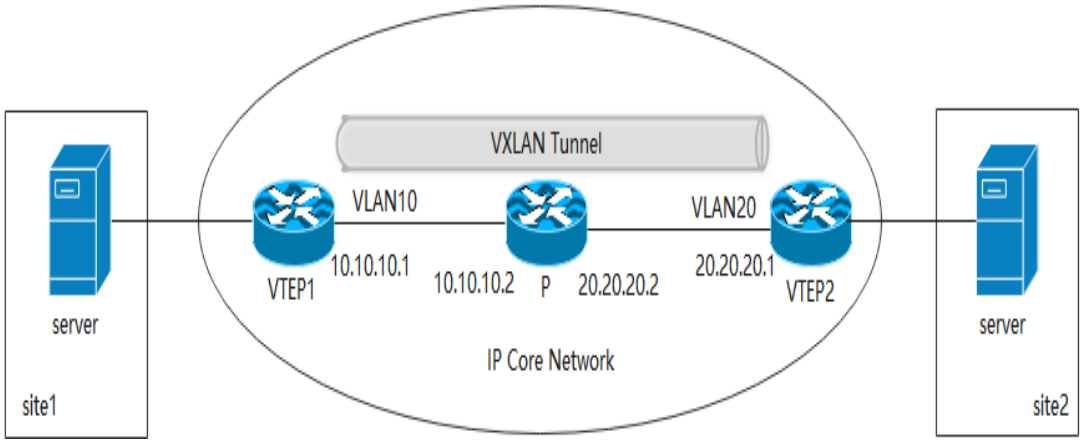


Figure 14-4 EVPN VXLAN Layer 2 Interconnection Network

VTEP1 configuration:

#Configure interfaces vlan10 and loopback1

```
VTEP1(config)#interface vlan 10
VTEP1(config-if-vlan10)#ip address 10.10.10.1 255.255.255.0
VTEP1(config)#interface loopback 1
VTEP1(config-if-loopback1)#ip address 1.1.1.1 255.255.255.255
```

#Configure the global NVE source address

```
VTEP1(config)#evpn nve source-address 1.1.1.1
```

#Configure virtual switch instance NVI 10 and enable EVPN

```
VTEP1(config)#nvi 10
VTEP1(config-nvi)#vxlan-id 10
VTEP1(config-nvi)#evpn
VTEP1(config-nvi-evpn)#rd 1:1
VTEP1(config-nvi-evpn)#route-target both 1:1
VTEP1(config-nvi-evpn)#enable
```

#Configure service access

```
VTEP1(config)#interface ethernet 1/0/1
VTEP1(config-if-ethernet1/0/1)#switchport mode trunk
VTEP1(config-if-ethernet1/0/1)#xconnect nvi 10
```

#Enable OSPF and configure the OSPF area of the interfaces

```
VTEP1(config)#router ospf
VTEP1(config-router)#ospf router-id 1.1.1.1
VTEP1(config-router)#network 1.1.1.1/32 area 0
VTEP1(config-router)#network 10.10.10.0/24 area 0
```

#Enable BGP and advertise the EVPN capability to the BGP peer

```
VTEP1(config)#router bgp 100
VTEP1(config-router)#neighbor 3.3.3.3 remote-as 100
VTEP1(config-router)#neighbor 3.3.3.3 update-source 1.1.1.1
VTEP1(config-router)#address-family l2vpn evpn
VTEP1(config-router-af)#neighbor 3.3.3.3 activate
VTEP1(config-router-af)#exit-address-family
```

P configuration:

#Configure interfaces vlan10, vlan20, and loopback1

```
P(config)#interface vlan 10
P(config-if-vlan10)#ip address 10.10.10.2 255.255.255.0
P(config)#interface vlan 20
P(config-if-vlan20)#ip address 20.20.20.2 255.255.255.0
```

```
P(config)#interface loopback 1
P(config-if-loopback1)#ip address 2.2.2.2 255.255.255.255
```

#Enable OSPF and configure the OSPF area of the interfaces

```
P(config)#router ospf
P(config-router)#ospf router-id 2.2.2.2
P(config-router)#network 2.2.2.2/32 area 0
P(config-router)#network 10.10.10.0/24 area 0
P(config-router)#network 20.20.20.0/24 area 0
```

VTEP2 configuration:

#Configure interfaces vlan20 and loopback1

```
VTEP2(config)#interface vlan 20
VTEP2(config-if-vlan10)#ip address 20.20.20.1 255.255.255.0
VTEP2(config)#interface loopback 1
VTEP2(config-if-loopback1)#ip address 3.3.3.3 255.255.255.255
```

#Configure the global NVE source address

```
VTEP2(config)#evpn nve source-address 3.3.3.3
```

#Configure virtual switch instance NVI 10 and enable EVPN

```
VTEP2(config)#nvi 10
VTEP2(config-nvi)#vxlan-id 10
VTEP2(config-nvi)#evpn
VTEP2(config-nvi-evpn)#rd 2:2
VTEP2(config-nvi-evpn)#route-target both 1:1
VTEP2(config-nvi-evpn)#enable
```

#Configure service access

```
VTEP2(config)#interface ethernet 1/0/1
VTEP2(config-if-ethernet1/0/1)#switchport mode trunk
VTEP2(config-if-ethernet1/0/1)#xconnect nvi 10
```

#Enable OSPF and configure the OSPF area of the interfaces

```
VTEP2(config)#router ospf
VTEP2(config-router)#ospf router-id 3.3.3.3
VTEP2(config-router)#network 3.3.3.3/32 area 0
VTEP2(config-router)#network 20.20.20.0/24 area 0
```

#Enable BGP and advertise the EVPN capability to the BGP peer

```
VTEP2(config)#router bgp 100
VTEP2(config-router)#neighbor 1.1.1.1 remote-as 100
VTEP2(config-router)#neighbor 1.1.1.1 update-source 3.3.3.3
VTEP2(config-router)#address-family l2vpn evpn
VTEP2(config-router-af)#neighbor 1.1.1.1 activate
VTEP2(config-router-af)#exit-address-family
```

14.4.3.3 VXLAN Centralized Gateway

14.4.3.3.1 EVPN VXLAN Centralized Gateway

EVPN typically uses a spine-leaf architecture. Leaf devices act as VTEPs and perform EVPN processing on packets; spine devices are core devices that forward packets based on their destination IP addresses.

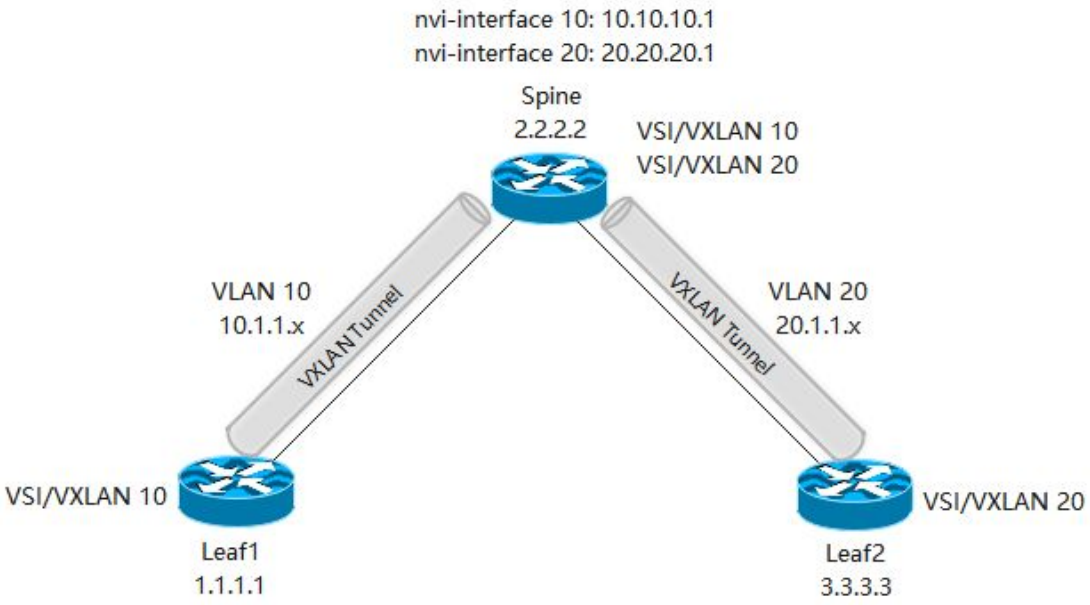


Figure 14-5 EVPN VXLAN Centralized Gateway

Spine configuration:

#Configure interfaces vlan10, vlan20, and loopback1

```
Spine(config)#interface vlan 10
Spine(config-if-vlan10)#ip address 10.1.1.2 255.255.255.0
Spine(config)#interface vlan 20
Spine(config-if-vlan20)#ip address 20.1.1.2 255.255.255.0
Spine(config)#interface loopback 1
Spine(config-if-loopback1)#ip address 2.2.2.2 255.255.255.255
```

#Disable the EVPN NVI-VLAN mapping monitor

```
Spine(config)#evpn nvi-vlan-mapping-monitor disable
```

#Configure the global NVE source address

```
Spine(config)#evpn nve source-address 2.2.2.2
```

#Configure virtual switch instances NVI 10 and NVI 20 and enable EVPN

```
Spine(config)#nvi 10
Spine(config-nvi)#vxlan-id 10
Spine(config-nvi)#evpn
Spine(config-nvi-evpn)#rd 10:10
Spine(config-nvi-evpn)#route-target both 1000:1000
Spine(config-nvi-evpn)#enable
Spine(config)#nvi 20
Spine(config-nvi)#vxlan-id 20
Spine(config-nvi)#evpn
Spine(config-nvi-evpn)#rd 20:20
Spine(config-nvi-evpn)#route-target both 2000:2000
Spine(config-nvi-evpn)#enable
```

#Configure the gateway interface of the virtual switch instance

```
Spine(config)#interface nvi-interface 10
Spine(config-if-nvi-interface10)#ipv6 address 10::1/64
Spine(config-if-nvi-interface10)#ip address 10.10.10.1 255.255.255.0
Spine(config)#interface nvi-interface 20
Spine(config-if-nvi-interface20)#ipv6 address 20::1/64
Spine(config-if-nvi-interface20)#ip address 20.20.20.1 255.255.255.0
```

#Enable OSPF and configure the OSPF area of the interfaces

```
Spine(config)#router ospf 1
```

```
Spine(config-router)#ospf router-id 2.2.2.2
Spine(config-router)#network 2.2.2.2 0.0.0.0 area 0
Spine(config-router)#network 10.1.1.0 0.0.0.255 area 0
Spine(config-router)#network 20.1.1.0 0.0.0.255 area 0
```

#Enable BGP and advertise the EVPN capability to the BGP peer

```
Spine(config)#router bgp 100
Spine(config-router)#neighbor 1.1.1.1 remote-as 100
Spine(config-router)#neighbor 1.1.1.1 update-source 2.2.2.2
Spine(config-router)#neighbor 3.3.3.3 remote-as 100
Spine(config-router)#neighbor 3.3.3.3 update-source 2.2.2.2
Spine(config-router)#address-family l2vpn evpn
Spine(config-router-af)#neighbor 1.1.1.1 activate
Spine(config-router-af)#neighbor 3.3.3.3 activate
Spine(config-router-af)#exit-address-family
```

Leaf1 configuration:

#Configure interfaces vlan10 and loopback1

```
Leaf1(config)#interface vlan 10
Leaf1(config-if-vlan10)#ip address 10.1.1.1 255.255.255.0
Leaf1(config)#interface loopback 1
Leaf1(config-if-loopback1)#ip address 1.1.1.1 255.255.255.255
```

#Configure the global NVE source address

```
Leaf1(config)#evpn nve source-address 1.1.1.1
```

#Configure virtual switch instance NVI 10 and enable EVPN

```
Leaf1(config)#nvi 10
Leaf1(config-nvi)#vxlan-id 10
Leaf1(config-nvi)#evpn
Leaf1(config-nvi-evpn)#rd 1:1
Leaf1(config-nvi-evpn)#route-target both 1000:1000
Leaf1(config-nvi-evpn)#enable
```

#Configure ARP/ND suppression

```
Leaf1(config-nvi)#arp suppression enable
Leaf1(config-nvi)#nd suppression enable
```

#Configure service access

```
Leaf1(config)#interface ethernet 1/0/5
Leaf1(config-if-ethernet1/0/5)#xconnect nvi 10 mode vlan svid 100
```

#Enable OSPF and configure the OSPF area of the interfaces

```
Leaf1(config)#router ospf 1
Leaf1(config-router)#ospf router-id 1.1.1.1
Leaf1(config-router)#network 1.1.1.1 0.0.0.0 area 0
Leaf1(config-router)#network 10.1.1.0 0.0.0.255 area 0
```

#Enable BGP and advertise the EVPN capability to the BGP peer

```
Leaf1(config)#router bgp 100
Leaf1(config-router)#neighbor 2.2.2.2 remote-as 100
Leaf1(config-router)#neighbor 2.2.2.2 update-source 1.1.1.1
Leaf1(config-router)#address-family l2vpn evpn
Leaf1(config-router-af)#neighbor 2.2.2.2 activate
Leaf1(config-router-af)#exit-address-family
```

Leaf2 configuration:

#Configure interfaces vlan20 and loopback1

```
Leaf2(config)#interface vlan 20
Leaf2(config-if-vlan20)#ip address 20.1.1.1 255.255.255.0
Leaf2(config)#interface loopback 1
Leaf2(config-if-loopback1)#ip address 3.3.3.3 255.255.255.255
```

#Configure the global NVE source address

```
Leaf2(config)#evpn nve source-address 3.3.3.3
```

#Configure virtual switch instance NVI 20 and enable EVPN

```
Leaf2(config)#nvi 20
Leaf2(config-nvi)#vxlan-id 20
Leaf2(config-nvi)#evpn
Leaf2(config-nvi-evpn)#rd 2:2
Leaf2(config-nvi-evpn)#route-target both 2000:2000
Leaf2(config-nvi-evpn)#enable
```

#Configure ARP/ND suppression

```
Leaf2(config-nvi)#arp suppression enable
Leaf2(config-nvi)#nd suppression enable
```

#Configure service access

```
Leaf2(config)#interface ethernet 1/0/5
Leaf2(config-if-ethernet1/0/5)#xconnect nvi 20 mode vlan svid 100
```

#Enable OSPF and configure the OSPF area of the interfaces

```
Leaf2(config)#router ospf 1
Leaf2(config-router)#ospf router-id 3.3.3.3
Leaf2(config-router)#network 3.3.3.3 0.0.0.0 area 0
Leaf2(config-router)#network 20.1.1.0 0.0.0.255 area 0
```

#Enable BGP and advertise the EVPN capability to the BGP peer

```
Leaf2(config)#router bgp 100
Leaf2(config-router)#neighbor 2.2.2.2 remote-as 100
Leaf2(config-router)#neighbor 2.2.2.2 update-source 3.3.3.3
Leaf2(config-router)#address-family l2vpn evpn
Leaf2(config-router-af)#neighbor 2.2.2.2 activate
Leaf2(config-router-af)#exit-address-family
```

14.4.3.3.2 Static VXLAN Centralized Gateway

Refer to the Static VXLAN example earlier in this chapter.

14.4.3.4 EVPN VXLAN Distributed Gateway

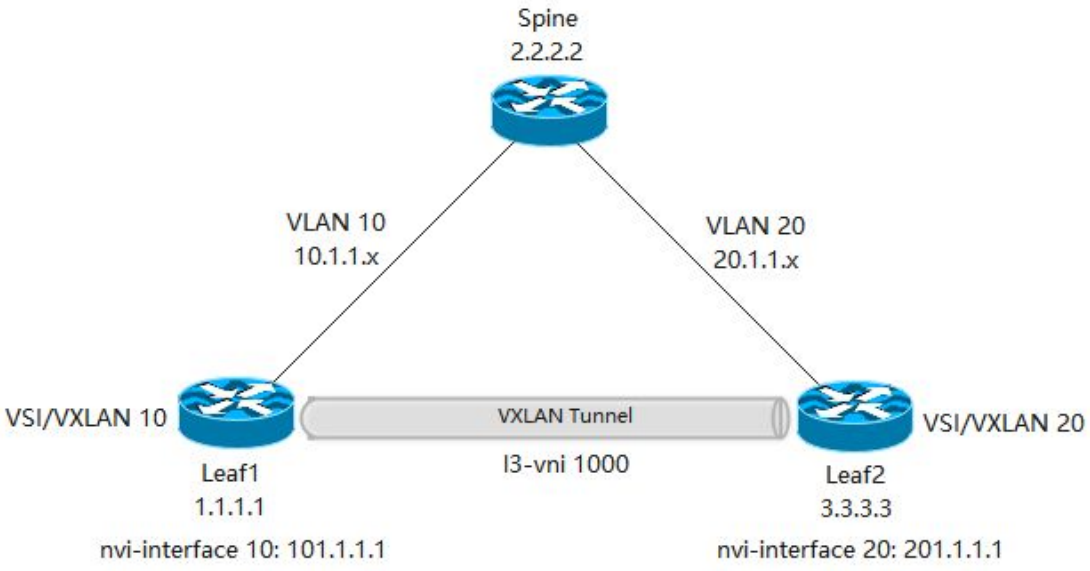


Figure 14-6 EVPN VXLAN Distributed Gateway

Spine configuration:

#Configure interfaces vlan10, vlan20, and loopback1

```
Spine(config)#interface vlan 10
Spine(config-if-vlan10)#ip address 10.1.1.2 255.255.255.0
Spine(config)#interface vlan 20
Spine(config-if-vlan20)#ip address 20.1.1.2 255.255.255.0
Spine(config)#interface loopback 1
Spine(config-if-loopback1)#ip address 2.2.2.2 255.255.255.255
```

#Enable OSPF and configure the OSPF area of the interfaces

```
Spine(config)#router ospf 1
Spine(config-router)#ospf router-id 2.2.2.2
Spine(config-router)#network 2.2.2.2 0.0.0.0 area 0
Spine(config-router)#network 10.1.1.0 0.0.0.255 area 0
Spine(config-router)#network 20.1.1.0 0.0.0.255 area 0
```

Leaf1 configuration:

#Configure interfaces vlan10 and loopback1

```
Leaf1(config)#interface vlan 10
Leaf1(config-if-vlan10)#ip address 10.1.1.1 255.255.255.0
Leaf1(config)#interface loopback 1
Leaf1(config-if-loopback1)#ip address 1.1.1.1 255.255.255.255
```

#Configure the global NVE source address

```
Leaf1(config)#evpn nve source-address 1.1.1.1
```

#Disable automatic learning of remote MAC/ARP/ND entries

```
Leaf1(config)#vxlan remote mac-address-learning disable
Leaf1(config)#vxlan remote arp-learning disable
Leaf1(config)#vxlan remote nd-learning disable
```

#Configure virtual switch instance NVI 10 and enable EVPN

```
Leaf1(config)#nvi 10
Leaf1(config-nvi)#vxlan-id 10
Leaf1(config-nvi)#evpn
Leaf1(config-nvi-evpn)#rd 1:1
Leaf1(config-nvi-evpn)#route-target both 1000:1000
Leaf1(config-nvi-evpn)#enable
```

#Configure ARP/ND suppression

```
Leaf1(config-nvi)#arp suppression enable
Leaf1(config-nvi)#nd suppression enable
```

#Configure service access

```
Leaf1(config)#interface ethernet 1/0/5
Leaf1(config-if-ethernet1/0/5)#xconnect nvi 10 mode vlan svid 100
```

#Configure an L3VPN instance and associate it with L3 VNI 1000

```
Leaf1(config)#ip vrf vpn1
Leaf1(config-vrf)#rd 11:1
Leaf1(config-vrf)#route-target both 10:1
Leaf1(config-vrf)#l3-vni 1000
```

#Configure the distributed gateway interface of the virtual switch instance

```
Leaf1(config)#interface nvi-interface 10
Leaf1(config-if-nvi-interface10)#ip vrf forwarding vpn1
Leaf1(config-if-nvi-interface10)#mac-address 00-ff-ff-00-00-01
Leaf1(config-if-nvi-interface10)#distributed-gateway enable
Leaf1(config-if-nvi-interface10)#ipv6 address 1001::1/64
Leaf1(config-if-nvi-interface10)#ip address 101.1.1.1 255.255.255.0
```

#Enable OSPF and configure the OSPF area of the interfaces

```
Leaf1(config)#router ospf 1
```

```
Leaf1(config-router)#ospf router-id 1.1.1.1
Leaf1(config-router)#network 1.1.1.1 0.0.0.0 area 0
Leaf1(config-router)#network 10.1.1.0 0.0.0.255 area 0
```

#Enable BGP and advertise the EVPN capability to the BGP peer

```
Leaf1(config)#router bgp 100
Leaf1(config-router)#neighbor 3.3.3.3 remote-as 100
Leaf1(config-router)#neighbor 3.3.3.3 update-source 1.1.1.1
Leaf1(config-router)#address-family l2vpn evpn
Leaf1(config-router-af)#neighbor 3.3.3.3 activate
Leaf1(config-router-af)#exit-address-family
Leaf1(config-router)#address-family ipv4 vrf vpn1
Leaf1(config-router-af)#redistribute connected
Leaf1(config-router-af)#exit-address-family
```

Leaf2 configuration:

#Configure interfaces vlan20 and loopback1

```
Leaf2(config)#interface vlan 20
Leaf2(config-if-vlan20)#ip address 20.1.1.1 255.255.255.0
Leaf2(config)#interface loopback 1
Leaf2(config-if-loopback1)#ip address 3.3.3.3 255.255.255.255
```

#Configure the global NVE source address

```
Leaf2(config)#evpn nve source-address 3.3.3.3
```

#Disable automatic learning of remote MAC/ARP/ND entries

```
Leaf2(config)#vxlan remote mac-address-learning disable
Leaf2(config)#vxlan remote arp-learning disable
Leaf2(config)#vxlan remote nd-learning disable
```

#Configure virtual switch instance NVI 20 and enable EVPN

```
Leaf2(config)#nvi 20
Leaf2(config-nvi)#vxlan-id 20
Leaf2(config-nvi)#evpn
Leaf2(config-nvi-evpn)#rd 2:2
Leaf2(config-nvi-evpn)#route-target both 1000:1000
Leaf2(config-nvi-evpn)#enable
```

#Configure ARP/ND suppression

```
Leaf2(config-nvi)#arp suppression enable
Leaf2(config-nvi)#nd suppression enable
```

#Configure service access

```
Leaf2(config)#interface ethernet 1/0/5
Leaf2(config-if-ethernet1/0/5)#xconnect nvi 20 mode vlan svid 100
```

#Configure an L3VPN instance and associate it with L3 VNI 1000

```
Leaf2(config)#ip vrf vpn1
Leaf2(config-vrf)#rd 11:1
Leaf2(config-vrf)#route-target both 10:1
Leaf2(config-vrf)#l3-vni 1000
```

#Configure the distributed gateway interface of the virtual switch instance

```
Leaf2(config)#interface nvi-interface 20
Leaf2(config-if-nvi-interface20)#ip vrf forwarding vpn1
Leaf2(config-if-nvi-interface20)#mac-address 00-ff-ff-00-00-02
Leaf2(config-if-nvi-interface20)#distributed-gateway enable
Leaf2(config-if-nvi-interface20)#ipv6 address 2001::1/64
Leaf2(config-if-nvi-interface20)#ip address 201.1.1.1 255.255.0.0
```

#Enable OSPF and configure the OSPF area of the interfaces

```
Leaf2(config)#router ospf 1
```

```
Leaf2(config-router)#ospf router-id 3.3.3.3
Leaf2(config-router)#network 3.3.3.3 0.0.0.0 area 0
Leaf2(config-router)#network 20.1.1.0 0.0.0.255 area 0
```

#Enable BGP and advertise the EVPN capability to the BGP peer

```
Leaf2(config)#router bgp 100
Leaf2(config-router)#neighbor 1.1.1.1 remote-as 100
Leaf2(config-router)#neighbor 1.1.1.1 update-source 3.3.3.3
Leaf2(config-router)#address-family l2vpn evpn
Leaf2(config-router-af)#neighbor 1.1.1.1 activate
Leaf2(config-router-af)#exit-address-family
Leaf2(config-router)#address-family ipv4 vrf vpn1
Leaf2(config-router-af)#redistribute connected
Leaf2(config-router-af)#exit-address-family
```

14.4.3.5 VXLAN Centralized Multi-Active Gateway

14.4.3.5.1 EVPN VXLAN Centralized Multi-Active Gateway

After the VEG function is enabled on both spines, the same NVE source address is configured, and the VEG is bound to the NVI interface, ARP and ND synchronization is enabled automatically.

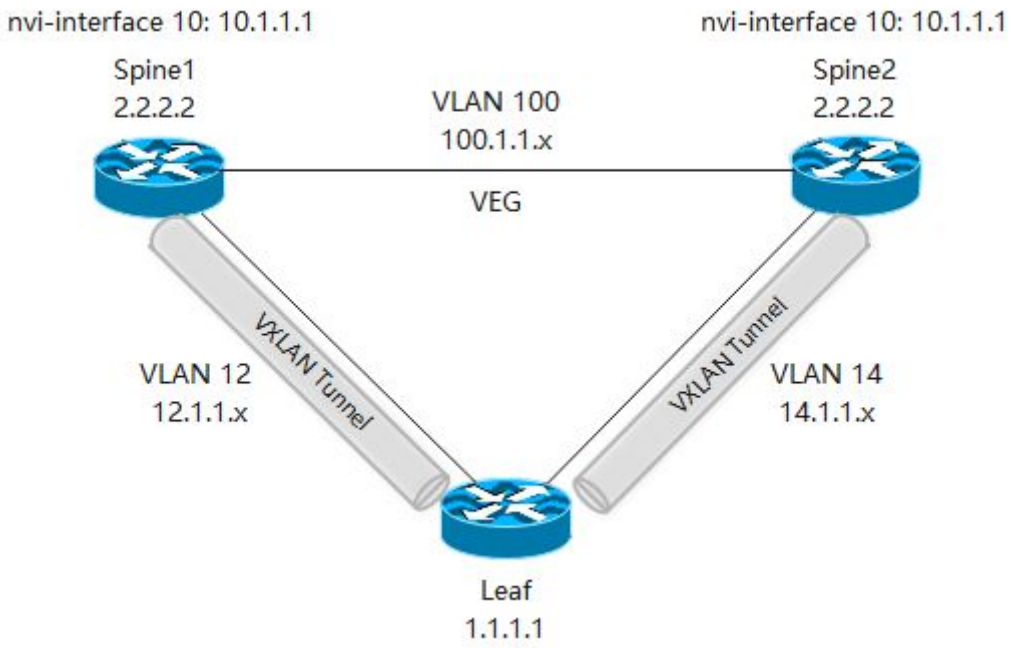


Figure 14-7 EVPN VXLAN Centralized Multi-Active Gateway

Spine1 configuration:

#Configure interfaces vlan12, vlan100, loopback1, and loopback2

```
Spine1(config)#interface vlan 12
Spine1(config-if-vlan12)#ip address 12.1.1.2 255.255.255.0
Spine1(config)#interface vlan 100
Spine1(config-if-vlan100)#ip address 100.1.1.3 255.255.255.0
Spine1(config)#interface loopback 1
Spine1(config-if-loopback1)#ip address 2.2.2.2 255.255.255.255
Spine1(config)#interface loopback 2
Spine1(config-if-loopback2)#ip address 5.5.5.5 255.255.255.255
```

#Configure the global NVE source address

```
Spine1(config)#evpn nve source-address 2.2.2.2
```

#Disable the EVPN NVI-VLAN mapping monitor

```
Spine1(config)#evpn nvi-vlan-mapping-monitor disable
```

#Configure the virtual equipment group

```
Spine1(config)#virtual-equipment-group 1
Spine1(config-veg1)#source ip 100.1.1.3
Spine1(config-veg1)#remote ip 100.1.1.2
```

#Configure virtual switch instance NVI 10 and enable EVPN

```
Spine1(config)#nvi 10
Spine1(config-nvi)#vxlan-id 10
Spine1(config-nvi)#evpn
Spine1(config-nvi-evpn)#rd 1:1
Spine1(config-nvi-evpn)#route-target both 1000:1000
Spine1(config-nvi-evpn)#enable
```

#Configure the gateway interface of the virtual switch instance

```
Spine1(config)#interface nvi-interface 10
Spine1(config-if-nvi-interface10)#mac-address 90-3c-bb-aa-ee-ff
Spine1(config-if-nvi-interface10)#ipv6 address 10::1/64
Spine1(config-if-nvi-interface10)#virtual-equipment-group 1
Spine1(config-if-nvi-interface10)#ip address 10.1.1.1 255.255.255.0
```

#Enable OSPF and configure the OSPF area of the interfaces

```
Spine1(config)#router ospf 1
Spine1(config-router)#ospf router-id 5.5.5.5
Spine1(config-router)#network 2.2.2.2 0.0.0.0 area 0
Spine1(config-router)#network 5.5.5.5 0.0.0.0 area 0
Spine1(config-router)#network 12.1.1.0 0.0.0.255 area 0
Spine1(config-router)#network 100.1.1.0 0.0.0.255 area 0
```

#Enable BGP and advertise the EVPN capability to the BGP peer

```
Spine1(config)#router bgp 100
Spine1(config-router)#neighbor 1.1.1.1 remote-as 100
Spine1(config-router)#neighbor 1.1.1.1 update-source 5.5.5.5
Spine1(config-router)#address-family l2vpn evpn
Spine1(config-router-af)#neighbor 1.1.1.1 activate
Spine1(config-router-af)#exit-address-family
```

Spine2 configuration:

#Configure interfaces vlan14, vlan100, loopback1, and loopback2

```
Spine2(config)#interface vlan 14
Spine2(config-if-vlan14)#ip address 14.1.1.2 255.255.255.0
Spine2(config)#interface vlan 100
Spine2(config-if-vlan100)#ip address 100.1.1.2 255.255.255.0
Spine2(config)#interface loopback 1
Spine2(config-if-loopback1)#ip address 2.2.2.2 255.255.255.255
Spine2(config)#interface loopback 2
Spine2(config-if-loopback2)#ip address 6.6.6.6 255.255.255.255
```

#Configure the global NVE source address

```
Spine2(config)#evpn nve source-address 2.2.2.2
```

#Disable the EVPN NVI-VLAN mapping monitor

```
Spine2(config)#evpn nvi-vlan-mapping-monitor disable
```

#Configure the virtual equipment group

```
Spine2(config)#virtual-equipment-group 1
Spine2(config-veg1)#source ip 100.1.1.2
Spine2(config-veg1)#remote ip 100.1.1.3
```

#Configure virtual switch instance NVI 10 and enable EVPN

```
Spine2(config)#nvi 10
```

```
Spine2(config-nvi)#vxlan-id 10
Spine2(config-nvi)#evpn
Spine2(config-nvi-evpn)#rd 1:1
Spine2(config-nvi-evpn)#route-target both 1000:1000
Spine2(config-nvi-evpn)#enable
```

#Configure the gateway interface of the virtual switch instance

```
Spine2(config)#interface nvi-interface 10
Spine2(config-if-nvi-interface10)#mac-address 90-3c-bb-aa-ee-ff
Spine2(config-if-nvi-interface10)#ipv6 address 10::1/64
Spine2(config-if-nvi-interface10)#virtual-equipment-group 1
Spine2(config-if-nvi-interface10)#ip address 10.1.1.1 255.255.255.0
```

#Enable OSPF and configure the OSPF area of the interfaces

```
Spine2(config)#router ospf 1
Spine2(config-router)#ospf router-id 6.6.6.6
Spine2(config-router)#network 2.2.2.2 0.0.0.0 area 0
Spine2(config-router)#network 6.6.6.6 0.0.0.0 area 0
Spine2(config-router)#network 14.1.1.0 0.0.0.255 area 0
Spine2(config-router)#network 100.1.1.0 0.0.0.255 area 0
```

#Enable BGP and advertise the EVPN capability to the BGP peer

```
Spine2(config)#router bgp 100
Spine2(config-router)#neighbor 1.1.1.1 remote-as 100
Spine2(config-router)#neighbor 1.1.1.1 update-source 6.6.6.6
Spine2(config-router)#address-family l2vpn evpn
Spine2(config-router-af)#neighbor 1.1.1.1 activate
Spine2(config-router-af)#exit-address-family
```

Leaf configuration:

#Configure interfaces vlan12, vlan14, and loopback1

```
Leaf(config)#interface vlan 12
Leaf(config-if-vlan12)#ip address 12.1.1.1 255.255.255.0
Leaf(config)#interface vlan 14
Leaf(config-if-vlan14)#ip address 14.1.1.1 255.255.255.0
Leaf(config)#interface loopback 1
Leaf(config-if-loopback1)#ip address 1.1.1.1 255.255.255.255
```

#Configure the global NVE source address

```
Leaf(config)#evpn nve source-address 1.1.1.1
```

#Configure virtual switch instance NVI 10 and enable EVPN

```
Leaf(config)#nvi 10
Leaf(config-nvi)#vxlan-id 10
Leaf(config-nvi)#evpn
Leaf(config-nvi-evpn)#rd 1:1
Leaf(config-nvi-evpn)#route-target both 1000:1000
Leaf(config-nvi-evpn)#enable
```

#Configure ARP/ND suppression

```
Leaf(config-nvi)#arp suppression enable
Leaf(config-nvi)#nd suppression enable
```

#Configure service access

```
Leaf(config)#interface ethernet 1/0/5
Leaf(config-if-ethernet1/0/5)#xconnect nvi 10 mode vlan svid 100
```

#Enable OSPF and configure the OSPF area of the interfaces

```
Leaf(config)#router ospf 1
Leaf(config-router)#ospf router-id 1.1.1.1
Leaf(config-router)#network 1.1.1.1 0.0.0.0 area 0
Leaf(config-router)#network 12.1.1.0 0.0.0.255 area 0
```

```
Leaf(config-router)#network 14.1.1.0 0.0.0.255 area 0
```

#Enable BGP and advertise the EVPN capability to the BGP peer

```
Leaf(config)#router bgp 100
Leaf(config-router)#neighbor 5.5.5.5 remote-as 100
Leaf(config-router)#neighbor 5.5.5.5 update-source 1.1.1.1
Leaf(config-router)#neighbor 6.6.6.6 remote-as 100
Leaf(config-router)#neighbor 6.6.6.6 update-source 1.1.1.1
Leaf(config-router)#address-family l2vpn evpn
Leaf(config-router-af)#neighbor 5.5.5.5 activate
Leaf(config-router-af)#neighbor 6.6.6.6 activate
Leaf(config-router-af)#exit-address-family
```

14.4.3.5.2 Static VXLAN Centralized Multi-Active Gateway

Refer to the Static VXLAN example earlier in this chapter.

14.4.3.6 EVPN VXLAN Dual-Homing Access

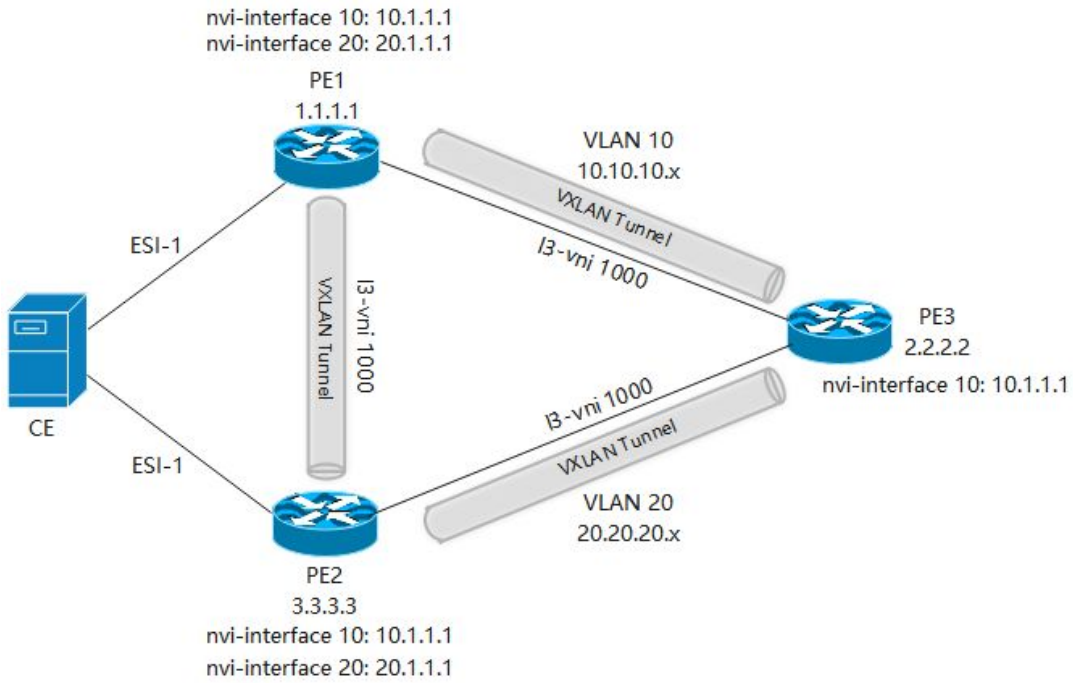


Figure 14-8 EVPN VXLAN Dual-Homing Access

PE1 configuration:

#Configure interfaces vlan10 and loopback1

```
PE1(config)#interface vlan 10
PE1(config-if-vlan10)#ip address 10.10.10.1 255.255.255.0
PE1(config)#interface loopback 1
PE1(config-if-loopback1)#ip address 1.1.1.1 255.255.255.255
```

#Configure the global NVE source address

```
PE1(config)#evpn nve source-address 1.1.1.1
```

#Disable automatic learning of remote MAC/ARP/ND entries

```
PE1(config)#vxlan remote mac-address-learning disable
PE1(config)#vxlan remote arp-learning disable
PE1(config)#vxlan remote nd-learning disable
```

#Configure virtual switch instances NVI 10 and NVI 20 and enable EVPN

```
PE1(config)#nvi 10
```

```
PE1(config-nvi)#vxlan-id 10
PE1(config-nvi)#evpn
PE1(config-nvi-evpn)#rd 1:1
PE1(config-nvi-evpn)#route-target both 100:100
PE1(config-nvi-evpn)#enable
PE1(config)#nvi 20
PE1(config-nvi)#vxlan-id 20
PE1(config-nvi)#evpn
PE1(config-nvi-evpn)#rd 30:30
PE1(config-nvi-evpn)#route-target both 200:200
PE1(config-nvi-evpn)#enable
```

#Configure service access and associate it with an ESI

```
PE1(config)#interface ethernet 1/0/5
PE1(config-if-ethernet1/0/5)#xconnect nvi 10 mode vlan svid 100
PE1(config-if-ethernet1/0/5)#xconnect nvi 20 mode vlan svid 200
PE1(config-if-ethernet1/0/5)#esi 0011.2222.3333.4444.4444
```

#Configure an L3VPN instance and associate it with L3 VNI 1000

```
PE1(config)#ip vrf vpn1
PE1(config-vrf)#rd 33:1
PE1(config-vrf)#route-target both 100:1
PE1(config-vrf)#l3-vni 1000
```

#Configure the distributed gateway interface of the virtual switch instance

```
PE1(config)#interface nvi-interface 10
PE1(config-if-nvi-interface10)#ip vrf forwarding vpn1
PE1(config-if-nvi-interface10)#mac-address 00-00-33-33-99-99
PE1(config-if-nvi-interface10)#distributed-gateway enable
PE1(config-if-nvi-interface10)#ipv6 address 2103::1/64
PE1(config-if-nvi-interface10)#ip address 10.1.1.1 255.255.255.0
PE1(config)#interface nvi-interface 20
PE1(config-if-nvi-interface20)#ip vrf forwarding vpn1
PE1(config-if-nvi-interface20)#mac-address 00-00-33-34-99-99
PE1(config-if-nvi-interface20)#distributed-gateway enable
PE1(config-if-nvi-interface20)#ipv6 address 2203::1/64
PE1(config-if-nvi-interface20)#ip address 20.1.1.1 255.255.255.0
```

#Enable OSPF and configure the OSPF area of the interfaces

```
PE1(config)#router ospf 1
PE1(config-router)#ospf router-id 1.1.1.1
PE1(config-router)#network 1.1.1.1 0.0.0.0 area 0
PE1(config-router)#network 10.10.10.0 0.0.0.255 area 0
```

#Enable BGP and advertise the EVPN capability to the BGP peer

```
PE1(config)#router bgp 100
PE1(config-router)#neighbor 2.2.2.2 remote-as 100
PE1(config-router)#neighbor 2.2.2.2 update-source 1.1.1.1
PE1(config-router)#neighbor 3.3.3.3 remote-as 100
PE1(config-router)#neighbor 3.3.3.3 update-source 1.1.1.1
PE1(config-router)#address-family l2vpn evpn
PE1(config-router-af)#neighbor 2.2.2.2 activate
PE1(config-router-af)#neighbor 3.3.3.3 activate
PE1(config-router-af)#exit-address-family
PE1(config-router)#address-family ipv4 vrf vpn1
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#exit-address-family
```

PE2 configuration:

#Configure interfaces vlan20 and loopback1

```
PE2(config)#interface vlan 20
PE2(config-if-vlan20)#ip address 20.20.20.1 255.255.255.0
```

```
PE2(config)#interface loopback 1
PE2(config-if-loopback1)#ip address 3.3.3.3 255.255.255.255
```

#Configure the global NVE source address

```
PE2(config)#evpn nve source-address 3.3.3.3
```

#Disable automatic learning of remote MAC/ARP/ND entries

```
PE2(config)#vxlan remote mac-address-learning disable
PE2(config)#vxlan remote arp-learning disable
PE2(config)#vxlan remote nd-learning disable
```

#Configure virtual switch instances NVI 10 and NVI 20 and enable EVPN

```
PE2(config)#nvi 10
PE2(config-nvi)#vxlan-id 10
PE2(config-nvi)#evpn
PE2(config-nvi-evpn)#rd 3:3
PE2(config-nvi-evpn)#route-target both 100:100
PE2(config-nvi-evpn)#enable
PE2(config)#nvi 20
PE2(config-nvi)#vxlan-id 20
PE2(config-nvi)#evpn
PE2(config-nvi-evpn)#rd 30:30
PE2(config-nvi-evpn)#route-target both 200:200
PE2(config-nvi-evpn)#enable
```

#Configure service access and associate it with an ESI

```
PE2(config)#interface ethernet 1/0/5
PE2(config-if-ethernet1/0/5)#xconnect nvi 10 mode vlan svid 100
PE2(config-if-ethernet1/0/5)#xconnect nvi 20 mode vlan svid 200
PE2(config-if-ethernet1/0/5)#esi 0011.2222.3333.4444.4444
```

#Configure an L3VPN instance and associate it with L3 VNI 1000

```
PE2(config)#ip vrf vpn1
PE2(config-vrf)#rd 33:1
PE2(config-vrf)#route-target both 100:1
PE2(config-vrf)#l3-vni 1000
```

#Configure the distributed gateway interface of the virtual switch instance

```
PE2(config)#interface nvi-interface 10
PE2(config-if-nvi-interface10)#ip vrf forwarding vpn1
PE2(config-if-nvi-interface10)#mac-address 00-00-33-33-99-99
PE2(config-if-nvi-interface10)#distributed-gateway enable
PE2(config-if-nvi-interface10)#ipv6 address 2103::1/64
PE2(config-if-nvi-interface10)#ip address 10.1.1.1 255.255.255.0
PE2(config)#interface nvi-interface 20
PE2(config-if-nvi-interface20)#ip vrf forwarding vpn1
PE2(config-if-nvi-interface20)#mac-address 00-00-33-34-99-99
PE2(config-if-nvi-interface20)#distributed-gateway enable
PE2(config-if-nvi-interface20)#ipv6 address 2203::1/64
PE2(config-if-nvi-interface20)#ip address 20.1.1.1 255.255.255.0
```

#Enable OSPF and configure the OSPF area of the interfaces

```
PE2(config)#router ospf 1
PE2(config-router)#ospf router-id 3.3.3.3
PE2(config-router)#network 3.3.3.3 0.0.0.0 area 0
PE2(config-router)#network 20.20.20.0 0.0.0.255 area 0
```

#Enable BGP and advertise the EVPN capability to the BGP peer

```
PE2(config)#router bgp 100
PE2(config-router)#neighbor 2.2.2.2 remote-as 100
PE2(config-router)#neighbor 2.2.2.2 update-source 3.3.3.3
PE2(config-router)#neighbor 1.1.1.1 remote-as 100
PE2(config-router)#neighbor 1.1.1.1 update-source 3.3.3.3
```

```
PE2(config-router)#address-family l2vpn evpn
PE2(config-router-af)#neighbor 2.2.2.2 activate
PE2(config-router-af)#neighbor 1.1.1.1 activate
PE2(config-router-af)#exit-address-family
PE2(config-router)#address-family ipv4 vrf vpn1
PE2(config-router-af)#redistribute connected
PE2(config-router-af)#exit-address-family
```

PE3 configuration:

#Configure interfaces vlan10, vlan20, and loopback1

```
PE3(config)#interface vlan 10
PE3(config-if-vlan10)#ip address 10.10.10.2 255.255.255.0
PE3(config)#interface vlan 20
PE3(config-if-vlan20)#ip address 20.20.20.2 255.255.255.0
PE3(config)#interface loopback 1
PE3(config-if-loopback1)#ip address 2.2.2.2 255.255.255.255
```

#Configure the global NVE source address

```
PE3(config)#evpn nve source-address 2.2.2.2
```

#Disable automatic learning of remote MAC/ARP/ND entries

```
PE3(config)#vxlan remote mac-address-learning disable
PE3(config)#vxlan remote arp-learning disable
PE3(config)#vxlan remote nd-learning disable
```

#Configure virtual switch instance NVI 10 and enable EVPN

```
PE3(config)#nvi 10
PE3(config-nvi)#vxlan-id 10
PE3(config-nvi)#evpn
PE3(config-nvi-evpn)#rd 2:2
PE3(config-nvi-evpn)#route-target both 100:100
PE3(config-nvi-evpn)#enable
```

#Configure an L3VPN instance and associate it with L3 VNI 1000

```
PE3(config)#ip vrf vpn1
PE3(config-vrf)#rd 1:1
PE3(config-vrf)#route-target both 100:1
PE3(config-vrf)#l3-vni 1000
```

#Configure the distributed gateway interface of the virtual switch instance

```
PE3(config)#interface nvi-interface 10
PE3(config-if-nvi-interface10)#ip vrf forwarding vpn1
PE3(config-if-nvi-interface10)#mac-address 00-00-33-33-99-99
PE3(config-if-nvi-interface10)#distributed-gateway enable
PE3(config-if-nvi-interface10)#ipv6 address 2103::1/64
PE3(config-if-nvi-interface10)#ip address 10.1.1.1 255.255.255.0
```

#Enable OSPF and configure the OSPF area of the interfaces

```
PE3(config)#router ospf 1
PE3(config-router)#ospf router-id 2.2.2.2
PE3(config-router)#network 2.2.2.2 0.0.0.0 area 0
PE3(config-router)#network 10.10.10.0 0.0.0.255 area 0
PE3(config-router)#network 20.20.20.0 0.0.0.255 area 0
```

#Enable BGP and advertise the EVPN capability to the BGP peer

```
PE3(config)#router bgp 100
PE3(config-router)#neighbor 3.3.3.3 remote-as 100
PE3(config-router)#neighbor 3.3.3.3 update-source 2.2.2.2
PE3(config-router)#neighbor 1.1.1.1 remote-as 100
PE3(config-router)#neighbor 1.1.1.1 update-source 2.2.2.2
PE3(config-router)#address-family l2vpn evpn
PE3(config-router-af)#neighbor 3.3.3.3 activate
```

```
PE3(config-router-af)#neighbor 1.1.1.1 activate
PE3(config-router-af)#exit-address-family
PE3(config-router)#address-family ipv4 vrf vpn1
PE3(config-router-af)#redistribute connected
PE3(config-router-af)#exit-address-family
```

14.4.4 VXLAN Troubleshooting

VXLAN services may fail because of physical connection problems or configuration errors. Observe the following points:

- First, make sure that the physical connections are correct.
- Second, make sure that the interface and the link protocol are up (run the **show interface** command).
- Then, make sure that the OSPF neighbors between VTEP1, P, and VTEP2 are established, that routes, including those of the loopback interfaces, are advertised normally, and that the BGP neighbors between the VTEPs are established.
- After the configuration is complete, run the **show ip bgp summary** command to check that all BGP neighbor connections are established, the **show ip bgp evpn all** command to view the BGP EVPN routing table, and the **show ip route** command to view the IP routing table.
  - Finally, use the show commands to view the sent and received packet statistics on the NVI and NVE interfaces for troubleshooting.