



Alcatel-Lucent

7450 ESS, 7750 SR and 7950 XRS | RELEASES UP TO 13.0.R6
ADVANCED CONFIGURATION GUIDE - PART I

Alcatel-Lucent – Proprietary & Confidential
Contains proprietary/trade secret information which is the property of Alcatel-Lucent. Not to be made available to, or copied or used by anyone who is not an employee of Alcatel-Lucent except when there is a valid non-disclosure agreement in place which covers such information and contains appropriate non-disclosure and limited use obligations.
Copyright 2015 © Alcatel-Lucent. All rights reserved.

All specifications, procedures, and information in this document are subject to change and revision at any time without notice. The information contained herein is believed to be accurate as of the date of publication. Alcatel-Lucent provides no warranty, express or implied, regarding its contents. Users are fully responsible for application or use of the documentation.

Alcatel, Lucent, Alcatel-Lucent and the Alcatel-Lucent logo are trademarks of Alcatel-Lucent. All other trademarks are the property of their respective owners.

Copyright 2015 Alcatel-Lucent.

All rights reserved.

Disclaimers

Alcatel-Lucent products are intended for commercial uses. Without the appropriate network design engineering, they must not be sold, licensed or otherwise distributed for use in any hazardous environments requiring fail-safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life-support machines, or weapons systems, in which the failure of products could lead directly to death, personal injury, or severe physical or environmental damage. The customer hereby agrees that the use, sale, license or other distribution of the products for any such application without the prior written consent of Alcatel-Lucent, shall be at the customer's sole risk. The customer hereby agrees to defend and hold Alcatel-Lucent harmless from any claims for loss, cost, damage, expense or liability that may arise out of or in connection with the use, sale, license or other distribution of the products in such applications.

This document may contain information regarding the use and installation of non-Alcatel-Lucent products. Please note that this information is provided as a courtesy to assist you. While Alcatel-Lucent tries to ensure that this information accurately reflects information provided by the supplier, please refer to the materials provided with any non-Alcatel-Lucent product and contact the supplier for confirmation. Alcatel-Lucent assumes no responsibility or liability for incorrect or incomplete information provided about non-Alcatel-Lucent products.

However, this does not constitute a representation or warranty. The warranties provided for Alcatel-Lucent products, if any, are set forth in contractual documentation entered into by Alcatel-Lucent and its customers.

This document was originally written in English. If there is any conflict or inconsistency between the English version and any other version of a document, the English version shall prevail.

Table of Contents

Preface	21
Basic System	25
IEEE 1588 for Frequency, Phase, and Time Distribution	27
Multi-Chassis Synchronization for IGMP Snooping	65
Synchronous Ethernet	83
System Management	101
Distributed CPU Protection	103
Event Handling System	129
Interface Configuration	145
Multi-Chassis APS and Pseudowire Redundancy Interworking	147
Multi-Chassis LAG and Pseudowire Redundancy Interworking	169
Router Configuration	193
Aggregate Route Indirect Next-Hop Option	195
Bi-Directional Forwarding Detection	207
LFA Policies Using OSPF as IGP	255
Routing Protocols	273
Associating Communities with Static and Aggregate Routes	275
IS-IS Link Bundling	307
MPLS	329
Automatic Bandwidth Adjustment in P2P LSPs	331
Automatic Creation of RSVP-TE LSPs	371
BGP Anycast	393
IGP Shortcuts	437
Inter-Area TE Point-to-Point LSPs	499
LDP over RSVP Using OSPF as IGP	527

Table of Contents

MPLS LDP FRR using ISIS as IGP	569
MPLS Transport Profile	595
Point-to-Point LSPs	623
RSVP Signaled Point-to-Multipoint LSPs	657
Segment Routing with IS-IS Control Plane	707
Shared Risk Link Groups for RSVP-Based LSP	731
Services Overview	755
G.8032 Ethernet Ring Protection Multiple Ring Topology	757
G.8032 Ethernet Ring Protection Single Ring Topology	805
Services: Layer 2 and EVPN	829
BGP Multi-Homing for VPLS Networks	831
BGP VPLS	871
BGP Virtual Private Wire Services	911
EVPN for MPLS Tunnels	943
EVPN for PBB over MPLS (PBB-EVPN)	997
EVPN for VXLAN Tunnels (Layer 2)	1039
EVPN for VXLAN Tunnels (Layer 3)	1069
Inter-AS Model C for VLL	1109
LDP VPLS using BGP-Auto Discovery	1133
Multi-Chassis Endpoint for VPLS Active/Standby Pseudowire	1165
Multi-Segment Pseudowire Routing	1195
PBB-Epipe	1249
PBB-VPLS	1275
Shortest Path Bridging for MAC	1317
Services: Layer 3	1351
Carrier Supporting Carrier IP VPNs	1353
Layer 3 VPN: VPRN Type Spoke	1379
Multicast in a VPN I	1395

Multicast in a VPN II.	1453
Multicast VPN: Core Diversity	1509
Multicast VPN: Inter-AS Option B	1539
Multicast VPN: Sender-Only, Receiver-Only	1565
Multicast VPN: Use of Wildcard Selective PMSI.	1617
Source Redundancy in a Multicast VPN	1649
Spoke Termination for IPv6-6VPE	1689
VPRN Inter-AS VPN Model C	1721
Quality of Service	1737
Class Fair Hierarchical Policing for SAPs.	1739
Pseudowire QoS	1783
QoS Architecture and Basic Operation	1809

Table of Contents

List of Tables

Table 1:	Revertive, Non-Revertive Timing Reference Switching Operation	84
Table 2:	RSVP LSP Role As Outcome of LSP Level and IGP Level Configuration Options.	459
Table 3:	Mode Comparison	706
Table 4:	Terminology Comparison	755
Table 5:	VE-IDs and Labels	877
Table 6:	VE-IDs and Number of Labels	878
Table 7:	Comparing EVPN Multi-homing and BGP Multi-homing	982
Table 8:	EVPN and PBB-EVPN SR OS Feature Comparison	993
Table 9:	PBB-EVPN Multi-Homing Supported Combinations in SR OS	1014
Table 10:	Next Generation MVPN Components	1392
Table 11:	S-PMSI Auto-Discovery BGP NLRI	1615
Table 12:	Burst Levels	1737
Table 13:	Policer stat-mode	1748
Table 14:	SAP Ingress Classification Match Criteria	1809
Table 15:	QinQ Dot1p Bit Classification	1811
Table 16:	Forwarding Classes	1812
Table 17:	Queue Priority vs. Profile Mode	1816
Table 18:	Network QoS Policy DSCP Remarking	1823

Table of Contents

List of Figures

Figure 1:	PTP Messages and Timestamp Exchange	24
Figure 2:	1588 Topology for Frequency Distribution	26
Figure 3:	1588 Topology for Time Distribution	27
Figure 4:	Frequency Distribution with 1588 as Last Mile	28
Figure 5:	Unicast Message Negotiation	29
Figure 6:	Floor Packet Counting for FPP (n , W , δ)	32
Figure 7:	G.8271.1 Time Error Budget	34
Figure 8:	Master and Slave Clocks for Frequency	37
Figure 9:	Boundary Clock	46
Figure 10:	Boundary Clocks with Edge VPRN Access	51
Figure 11:	Configuration without MCS for IGMP Snooping	63
Figure 12:	Configuration with MCS for IGMP Snooping	70
Figure 13:	Multicast Stream Forwarded to PE-20	74
Figure 14:	SyncE Hypothetical Reference Network Architecture	80
Figure 15:	Packet Based Network Timing Infrastructure	82
Figure 16:	Current 7x50 Timing Sub-System Architecture	83
Figure 17:	Network Considerations for Ethernet Timing Distribution	86
Figure 18:	Test Topology	100
Figure 19:	Count Traffic with DCP Policy Count	106
Figure 20:	Limit Traffic with dcp-static-policy-1	109
Figure 21:	Dynamic Policing – Local Monitor	117
Figure 22:	Dynamic Policers Instantiated	117
Figure 23:	Example Topology	126
Figure 24:	MC-APS Network Topology	144
Figure 25:	Access Node and Network Resilience	145
Figure 26:	Association of SAPs/SDPs and Endpoints	151
Figure 27:	ICB Spoke SDPs and Association with the Endpoints	155
Figure 28:	Additional Setup Example 1	158
Figure 29:	Additional Setup Example 2 (Part 1)	159
Figure 30:	Additional Setup Example 2 (Part 2)	160
Figure 31:	MC-LAG Network Topology	166
Figure 32:	Network Resiliency	167
Figure 33:	Association of SAPs/SDPs and Endpoints	176
Figure 34:	ICB Spoke SDPs and Their Association with the Endpoints	180
Figure 35:	Additional Setup Example 1	183
Figure 36:	Additional Setup Example 2	184
Figure 37:	Aggregate Routes	191
Figure 38:	Test topology	192
Figure 39:	BFD Multi-Scenarios	204
Figure 40:	BFD Centralized Sessions	206
Figure 41:	BFD Interface Configuration	207
Figure 42:	BFD for ISIS	213
Figure 43:	BFD for OSPF	215
Figure 44:	BFD for OSPF and PIM	217
Figure 45:	BFD for Static Routes	219
Figure 46:	BFD for IES over Spoke SDP	222
Figure 47:	BFD for RSVP	227

Table of Contents

Figure 48:	BFD for T-LDP	231
Figure 49:	BFD for OSPF PE-CE I/F	234
Figure 50:	BFD Sessions within IPsec Tunnels	237
Figure 51:	Logic for Shared BFD Sessions	240
Figure 52:	BFD for VRRP	241
Figure 53:	Network Topology	252
Figure 54:	Network Topology	272
Figure 55:	CE Connections for Next-Hops	276
Figure 56:	CE-1 Connectivity	290
Figure 57:	CE-3 Connectivity	294
Figure 58:	Link Bundle Schematic	303
Figure 59:	Effect of Single Link Failure on Bundle Group	304
Figure 60:	Double Link Failure	305
Figure 61:	Test Topology	307
Figure 62:	Link Failure	315
Figure 63:	Second Link Failure	319
Figure 64:	Auto-Bandwidth Adjustment Implementation	329
Figure 65:	Underflow-Triggered Auto-Bandwidth Implementation	334
Figure 66:	Lab Setup for Auto-Bandwidth Point-to-Point LSPs	340
Figure 67:	Test Topology	368
Figure 68:	IGP Shortcuts with RSVP-TE Auto-Mesh	372
Figure 69:	Test Topology for Single-Hop LDP-over-RSVP with ECMP	380
Figure 70:	BGP Anycast Operation in GRT	390
Figure 71:	BGP Anycast Operation with IP-VPN	392
Figure 72:	BGP Anycast Data Path (BGP to BGP Swapping)	393
Figure 73:	BGP Anycast BGP Topology	394
Figure 74:	Anycast Address Configuration	397
Figure 75:	E2E MPLS Between Access Nodes	409
Figure 76:	Data Path with Failing ABR	411
Figure 77:	End-to-End Transport Tunnel Using Additional Loopback Interfaces	412
Figure 78:	IP-VPN with Anycast NH	414
Figure 79:	IP-VPN with Anycast NH, Data Path	428
Figure 80:	Normal SPF Tree Sourced by PE-1	433
Figure 81:	SPF Tree Sourced by PE-1 Using LSP Shortcuts	434
Figure 82:	Tested Network Topology	435
Figure 83:	LSPs Between PE-1 and PE-6	452
Figure 84:	RSVP Shortcuts LFA Use Case Example	464
Figure 85:	Network Topology to Verify Installation of Shortcuts into RTM	468
Figure 86:	Shortcuts Within a VRF Topology Network	485
Figure 87:	Inter-Area TE LSP Setup	497
Figure 88:	Inter-Area TE LSP Path	497
Figure 89:	ABR Protection	508
Figure 90:	Protection of All Nodes/Links Along the LSP Path	509
Figure 91:	Admin Group Example	511
Figure 92:	Share Risk Link Groups	515
Figure 93:	Initial Topology	524
Figure 94:	VPRN 1 with LDPoRSVP and No Intra-Area PE Connectivity	540
Figure 95:	VPRN 1 with LDPoRSVP and Intra-Area PE Connectivity	557
Figure 96:	Initial Topology	566
Figure 97:	LFA Computation, Inequality 3 for Prefix PE-4 (D) on PE1 (S)	573
Figure 98:	Data Verification, Direction PE-1 => PE-5 Using VLL Service	576

Figure 99:	LFA Computation, Inequality 3 for Prefix PE-5 (D) on PE-1 (S)	583
Figure 100:	LFA Computation, Inequality 1 for Prefix PE-5 (D) on PE-1 (S)	583
Figure 101:	IS-IS Overload on PE-2, Inequality 1 for 192.168.24.0/30 (D) on PE-1 (S)	587
Figure 102:	MPLS-TP Example Network Showing LSPs	594
Figure 103:	MPLS-TP Example Network Showing Services Detail	595
Figure 104:	MPLS-TP Configuration Steps	595
Figure 105:	LSP Path Label Value Configurations	604
Figure 106:	Generic MPLS Network, MPLS Label Operations	620
Figure 107:	MPLS Testbed Topology	622
Figure 108:	Static LSP Running Over PE-1 PE-2 PE-5 PE-6	624
Figure 109:	P2MP Network Topology	654
Figure 110:	P2MP LSP LSP-p2mp-1	659
Figure 111:	P2MP LSP p-to-mp-1 with Metric Change	678
Figure 112:	P2MP LSP LSP-p2mp-1 with Strict S2L Path Towards PE-7	681
Figure 113:	Intelligent Remerge, Case 1	684
Figure 114:	Intelligent Re-merge, Case 2	688
Figure 115:	Intelligent Re-merge, Case 3	693
Figure 116:	Network Topology	704
Figure 117:	RLFA Traffic Path During Protection	716
Figure 118:	Initial Topology	728
Figure 119:	SRLG Topology	729
Figure 120:	Path Primary RSVP_TE LSP	736
Figure 121:	SRLG for FRR Path With and Without SRLG	739
Figure 122:	SRLG for Secondary Path	742
Figure 123:	SRLG Database Example	745
Figure 124:	G.8032 Major Ring and Sub-Ring	756
Figure 125:	G.8032 Ring Components	758
Figure 126:	G.8032 Sub-Ring Interconnection Components	759
Figure 127:	Ethernet Test Topology	764
Figure 128:	ETH-CFM MEP Associations	766
Figure 129:	Sub-Ring to VPLS Topology	791
Figure 130:	G.8032 Operation and Topologies	803
Figure 131:	Test Topology	804
Figure 132:	Ethernet CFM Configuration	808
Figure 133:	Network Topology	829
Figure 134:	Nodes Involved in BGP MH	834
Figure 135:	MAC Flush for BGP MH	849
Figure 136:	Access PE/CE Signaling	850
Figure 137:	Oper-Groups and BGP-MH	854
Figure 138:	Network Topology	868
Figure 139:	BGP VPLS Using Auto-Provisioned SDPs	875
Figure 140:	BGP VPLS Using Pre-Provisioned SDP	893
Figure 141:	Network Topology	907
Figure 142:	Single Homed BGP VPWS using Auto-Provisioned SDPs	914
Figure 143:	Single Homed BGP VPWS using Pre-Provisioned SDP	921
Figure 144:	Dual Homed BGP VPWS with Single Pseudowire	925
Figure 145:	Dual Homed BGP VPWS with Active/Standby Pseudowire	931
Figure 146:	EVPN Route Types and NLRIs	940
Figure 147:	EVPN-MPLS for VPLS Services	942
Figure 148:	EVPN-MPLS All-Active Multi-Homing Concepts	956
Figure 149:	EVPN-MPLS Single-Active Multi-Homing: Mass-Withdraw, Backup Path	970

Table of Contents

Figure 150:	EVPN Route Types	994
Figure 151:	PBB-EVPN Network without Multi-Homing	997
Figure 152:	PBB-EVPN — Flooding Lists	1000
Figure 153:	PBB-EVPN Multi-homing	1012
Figure 154:	The Use of es-bmac to Minimize CMAC Flush	1014
Figure 155:	PBB-EVPN Single-Active Support for Epipes	1028
Figure 156:	EVPN-VXLAN Topology	1037
Figure 157:	BGP Adjacencies and Enabled Families	1040
Figure 158:	EVPN MAC Mobility	1054
Figure 159:	EVPN-VXLAN for R-VPLS Services	1066
Figure 160:	BGP adjacencies and enabled families	1069
Figure 161:	EVPN-VXLAN for IRB Backhaul R-VPLS Services	1074
Figure 162:	EVPN-VXLAN in EVPN-tunnel R-VPLS Services	1082
Figure 163:	Routing Policies for Egress EVPN Routes	1089
Figure 164:	Routing Policies for Ingress EVPN Routes	1090
Figure 165:	EVPN in Parallel R-VPLS Services	1095
Figure 166:	Network Setup - Inter-AS Model C for VLL	1106
Figure 167:	Inter-AS Model C for VLL	1106
Figure 168:	Network Setup Configuration	1107
Figure 169:	Network Topology	1130
Figure 170:	VPLS Instance with Auto-Provisioned SDPs	1139
Figure 171:	VPLS Instance using Pre-Provisioned SDPs	1151
Figure 172:	H-VPLS with STP	1161
Figure 173:	VPLS Pseudowire Redundancy	1162
Figure 174:	Multi-Chassis Endpoint with Mesh Resiliency	1162
Figure 175:	Multi-Chassis Endpoint with Square Resiliency	1163
Figure 176:	Network Topology	1164
Figure 177:	Core Node Failure	1180
Figure 178:	Multi-Chassis Node Failure	1182
Figure 179:	Multi-Chassis Passive Mode	1185
Figure 180:	FEC129 Structure	1192
Figure 181:	All Type 2 Format	1193
Figure 182:	Pseudowire Routing NLRI (the AC ID is always zero)	1193
Figure 183:	Configuration Flow Chart	1195
Figure 184:	Intra-AS MS-PW Network Topology	1214
Figure 185:	Inter-AS MS-PW Network Topology	1228
Figure 186:	Network Topology	1246
Figure 187:	Setup Detailed View	1248
Figure 188:	Virtual MEPs for Flooding Avoidance	1257
Figure 189:	Network Topology	1272
Figure 190:	MTU-1 and PE-1 Nodes as Configuration Examples	1274
Figure 191:	Blackhole	1287
Figure 192:	Send Flush on BVPLS Failure Example	1291
Figure 193:	Inter-Domain B-VPLS and MMRP Policies/ISID-Based Filters Example	1298
Figure 194:	Basic SPBM Topology	1315
Figure 195:	Control and User B-VPLS Test Topology	1326
Figure 196:	Access Resiliency Test Topology	1331
Figure 197:	Access Resiliency Test Topology	1335
Figure 198:	CSC Network Topology	1349
Figure 199:	CE Hub and Spoke Data Path	1376
Figure 200:	CE Hub and Spoke Control Plane Isolation	1377

Figure 201:	Internal VPRN Logic on a PE Router	1378
Figure 202:	CE Hub and Spoke Topology and Addressing Scheme	1379
Figure 203:	Network Topology	1394
Figure 204:	Network Topology for Anycast RP	1419
Figure 205:	IGMP and PIM Control Messaging Schematic	1426
Figure 206:	PIM SSM in Customer Signaling Plane	1430
Figure 207:	Network Topology	1451
Figure 208:	VPRN 1 Topology used for mLDP	1458
Figure 209:	VPRN 2 Topology used for RSVP-TE P2MP	1475
Figure 210:	VPRN 2 Topology used for MVPN Source Redundancy	1493
Figure 211:	VPRN 3 Topology used for MVPN Source Redundancy	1499
Figure 212:	Core Diversity Schematic	1506
Figure 213:	Core Diversity Network	1507
Figure 214:	Core Diversity Network — Base OSPF	1508
Figure 215:	Core Diversity Network - OSPF Instance 1	1509
Figure 216:	General Topology for Inter-AS MVPN	1535
Figure 217:	Protocols Used for Inter-AS MVPN	1535
Figure 218:	BGP Signaling Steps	1538
Figure 219:	PIM-P Signaling Steps for Default MDT	1539
Figure 220:	PIM-C Signaling	1540
Figure 221:	PIM-P Signaling Steps for Data MDT	1541
Figure 222:	Test Topology Details	1543
Figure 223:	BGP Signaling Steps	1546
Figure 224:	PIM-P Signaling Steps for Default MDT	1550
Figure 225:	PIM-C Signaling	1553
Figure 226:	PIM-P Signaling Steps for Data MDT	1555
Figure 227:	Default PMSI Hierarchy	1562
Figure 228:	Optimized PMSI Structure	1563
Figure 229:	Test Topology	1564
Figure 230:	RSVP-Based BGP Message Flow Between PE-2 and PE-3	1572
Figure 231:	RSVP-Based BGP Message Flow Between PE-1 and PE-3	1575
Figure 232:	mLDP-Based BGP Message Flow Between PE-2 and PE-3	1591
Figure 233:	mLDP-Based BGP Message Flow Between PE-1 and PE-3	1598
Figure 234:	Multicast VPN	1614
Figure 235:	Schematic Topology	1617
Figure 236:	S-PMSI P2MP LSP Schematic	1637
Figure 237:	Source Redundancy Example.	1646
Figure 238:	Schematic Topology	1649
Figure 239:	Spoke Termination for IPv6	1686
Figure 240:	IPv6 Addressing and IPv6 Prefixes	1686
Figure 241:	MP-BGP VPNv6	1688
Figure 242:	Spoke Termination for IPv6 Addressing	1690
Figure 243:	PE-3 VPRN with SAP to CE-2	1701
Figure 244:	Inter-AS VPN Model C	1718
Figure 245:	Protocol Overview	1718
Figure 246:	Policer Token Bucket Model	1736
Figure 247:	Peak Information Rate (PIR) Bucket	1738
Figure 248:	Committed Information Rate (CIR) Bucket	1739
Figure 249:	Fair Information Rate (FIR) Bucket	1740
Figure 250:	Policer and Arbiter Hierarchy	1742
Figure 251:	Parent Policer and Root Arbiter	1743

Table of Contents

Figure 252: Configuration Example	1745
Figure 253: Post Policing Queues	1746
Figure 254: Parent Policer Thresholds	1752
Figure 255: Ingress PW QoS	1780
Figure 256: Egress PW QoS	1780
Figure 257: Example Epipe Pseudowire Topology	1784
Figure 258: Service and Network QoS Policies	1807
Figure 259: Visualization of Default Network Policies	1827
Figure 260: Default Buffer Pools	1829
Figure 261: WRED Slope Characteristics	1834
Figure 262: Buffer Pools and Queue Sizing	1836
Figure 263: Scheduling (Dequeuing Packets from the Queue)	1839
Figure 264: IOM QoS Overview	1840

Preface

About This Guide

The Advanced Configuration Guide is divided into two books, The Part I Guide and the Part II Guide.

Part I provides advanced configurations for basic systems, system management, interface configuration, router configuration, routing protocols, MPLS, services overview, Layer 2 and EVPN services, Layer 3 services and Quality of Service.

Part II provides advanced configurations for Multi-Service Integrated Service Adapter and Triple Play Service Delivery Architecture.

Parts I and II of the Advanced Configuration Guide supplement the user configuration guides listed below.

The guide is organized alphabetically within each chapter and provides feature and configuration explanations, CLI descriptions and overall solutions. The chapters in the Advanced Configuration Guide are written for and tested on different releases, up to 13.0.R6. The Applicability section in each chapter specifies on which release the configuration was tested.

Audience

This manual is intended for network administrators who are responsible for configuring the routers. It is assumed that the network administrators have a detailed understanding of networking principles and configurations.

List of Technical Publications

The 7x50 series documentation set also includes of the following guides:

- Basic System Configuration Guide
This guide describes basic system configurations and operations.

- **System Management Guide**
This guide describes system security and access configurations as well as event logging and accounting logs.
- **Interface Configuration Guide**
This guide describes card, Media Dependent Adapter (MDA) and port provisioning.
- **Router Configuration Guide**
This guide describes logical IP routing interfaces and associated attributes such as an IP address, as well as IP and MAC-based filtering, and VRRP and Cflowd.
- **Routing Protocols Guide**
This guide provides an overview of routing concepts and provides configuration examples for RIP, OSPF, IS-IS, BGP, and route policies.
- **MPLS Configuration Guide**
This guide describes how to configure Multiprotocol Label Switching (MPLS) and Label Distribution Protocol (LDP).
- **Services Overview Guide**
This guide describes how to configure service parameters such as service distribution points (SDPs), customer information, and user services.
- **Layer 2 Services and EVPN Guide**
This guide describes Virtual Leased Lines (VLL), Virtual Private LAN Service (VPLS), Provider Backbone Bridging (PBB), and Ethernet VPN (EVPN).
- **Layer 3 Services Guide**
This guide describes Internet Enhanced Services (IES) and Virtual Private Routed Network (VPRN) services.
- **Versatile Service Module Guide**
This guide describes how to configure service parameters for the Versatile Service Module (VSM).
- **OAM and Diagnostics Guide**
This guide describes how to configure features such as service mirroring and Operations, Administration and Management (OAM) tools.
- **Triple Play Guide**
This guide describes Triple Play services and support provided by the routers and presents examples to configure and implement various protocols and services.
- **Quality of Service Guide**
This guide describes how to configure Quality of Service (QoS) policy management.
- **RADIUS Attributes Guide**

This guide describes all supported RADIUS Authentication, Authorization and Accounting attributes.

- Multi-Service Integrated Service Adapter Guide

This guide describes services provided by integrated service adapters such as Application Assurance, IPSec, ad insertion (ADI) and Network Address Translation (NAT).

- Gx AVPs Reference Guide

This guide describes Gx Attribute Value Pairs (AVP).

In This Section

This section provides configuration information for the following topics:

- [IEEE 1588 for Frequency, Phase, and Time Distribution](#) on page 21
- [Multi-Chassis Synchronization for IGMP Snooping](#) on page 59
- [Synchronous Ethernet](#) on page 77

IEEE 1588 for Frequency, Phase, and Time Distribution

In This Chapter

This section provides information about IEEE 1588 for frequency, phase, and time distribution.

Topics in this section include:

- [Applicability on page 22](#)
- [Summary on page 23](#)
- [Configuration on page 36](#)
- [Conclusion on page 57](#)

Applicability

This section is applicable to all of the 7750 SR and 7450 ESS series, except for the SR-1, ESS-1, and ESS-6/6v. It is not applicable to the 7710 SR nor the 7950 XRS series. Description and examples are based on release 12.0.R2. The only software pre-requisites are IP reachability between the node and neighboring 1588 clocks.

IEEE 1588 has several hardware dependencies both for the basic functionality as well as the 1588 port based timestamping necessary for high accuracy time distribution. Please consult the related Alcatel-Lucent documentation for the details of all the hardware requirements.

Summary

Defined in IEEE Std 1588™-2008 (1588v2), Precision Time Protocol (PTP) is a protocol that distributes frequency, phase and time over packet based networks¹. The IEEE 1588 protocol has become the standard for distribution of high accuracy time. Following guidelines for specific network architectures allows the delivery of time to accuracies of one microsecond. This level of accuracy is required for mobile base stations using either Time Division Duplex technology and/or advanced LTE functions, as well as in the power industry for intelligent electronic device alignment.

More lenient architectures can still achieve 100 microseconds or better accuracies which can greatly enhance the usefulness of event logging and network one way delay measurements.

In addition, 1588 has been used to deliver a frequency reference for T1/E1 ports or for mobile base station frequency alignment. This is useful in environments where the transport network does not provide physical layer synchronization services.

The following 1588 capabilities are provided within the 7750 SR and 7450 ESS nodes:

- CPM/CFM based 1588 master, boundary, and slave clock functionality
- Transport over Unicast UDP/IPv4 packets
- Access to 1588 process through base routing, IES, and VPRNs
- Port based timestamping of 1588 packets
- IEEE 1588 Profiles: 2008 standard default and ITU-T G.8265.1
- Utilization of 1588 derived time for NTP and System time.

1. Many applications do not need time alignment but only phase alignment. However, phase is derived from time and so, for the remainder of the document, discussion refers to time but those references imply both time and phase.

PTP Basics

PTP uses an exchange of four timestamps between a reference clock (master port) and the clock to be synchronized (slave port). A simplified illustration of this mechanism is shown in [Figure 1](#).

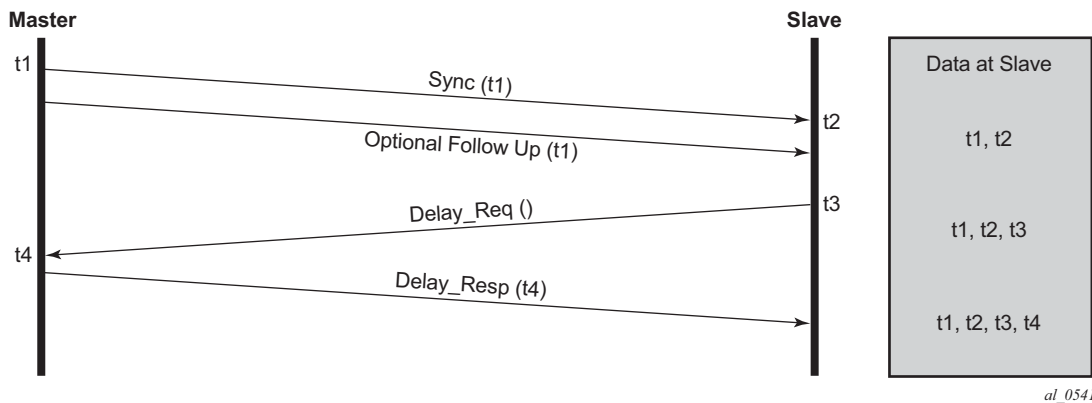


Figure 1: PTP Messages and Timestamp Exchange

The master sends a PTP Sync message containing a timestamp of when the Sync message is transmitted (t1) to the slave. In a two-step master clock, the t1 timestamp is sent in a Follow_Up² message. The slave records the time it receives the Sync message (t2). At some point after receiving the Sync message, the slave sends a Delay_Req message back to the master. The slave records the time of transmission of the Delay_Req message (t3) locally. The master records the time it receives the Delay_Req message (t4) and sends this timestamp back to the slave in a Delay_Resp message.

After the four timestamp exchange the slave can calculate the mean path delay and the clock offset from master using the following two equations:

-
- Note the Follow_Up message was defined to allow for implementations to generate a timestamp for the transmission of the Sync message but not have to try to insert that timestamp into the Sync message and update any frame checksums on the fly as it is in the process of transmission. While many recent implementations can perform the timestamping, update and checksum calculation on the fly, not all devices could perform this three step process with the desired accuracy. By using the Follow_Up message to transmit the timestamp of the Sync message, the master port can still provide extremely accurate timestamps for the transmission of the Sync message to the slave port. Apart from the extra message required, there is no detriment to a master port using one-step clock versus a two-step clock procedures. All PTP clocks that have slave port capability must accept timing information from both types of master port. There is no requirement to force a clock that is a one-step clock to use two-step clock procedures on its master ports. The nodes covered by this example all support one-step clock master port procedures.

$$\text{mean_path_delay} = [(t4-t1) - (t3-t2)] / 2$$
$$\text{offset_from_master} = [(t2 - t1) - \text{mean_path_delay}]$$

These calculations can occur on every message exchange or some initial packet selection can be performed so that only optimal message exchanges are used. The latter is useful if there is variable delay between the master and slave ports.

If only frequency is necessary, then the slave may use one or both pairs of timestamps (t1, t2) and (t3, t4). The slave can monitor the change in the perceived delay master-to-slave (t2 - t1) or slave-to-master (t4 - t3) over time. If the delay (t2 - t1) decreases over time, it means the t2 timestamps are not progressing quickly enough and the slave clock frequency needs to be increased.

If time is necessary, then all four timestamps must be used. It is also important to note how the equation for offset uses the mean_path_delay. If the delays in the two directions are actually different, then the equation will introduce an error in the offset_from_master that is half of the difference of the two delays. The IEEE 1588 standard includes procedures to compensate for this asymmetry, if it is known, but if it is uncompensated it does introduce time error.

PTP Deployment Architectures

It is important to understand that there are very different topologies recommended for using 1588 for frequency distribution and using 1588 for time distribution.

Frequency distribution was developed for an architecture where there are mobile providers who have points of presence at the mobile telephony switching offices (MTSOs) and the cell site locations which depend on other parties for the connectivity between the MTSOs and the cell site locations. The mobile providers wanted a solution that could span the transport networks with minimal dependence on that network. This can be achieved by placing a 1588 grandmaster at the MTSO and a slave in a cell site router or directly in the basestation and distributing the timestamped packets between the two, as shown in [Figure 2](#). The transport network does introduce packet delay variation (PDV) to the 1588 messages which makes it more difficult to track the frequency of the grandmaster's clock. However, the slaves have been designed to perform packet selection and noise filtering to allow for the recovery of a frequency within the required accuracies of the mobile basestations. This architecture and the performance requirements are covered by the ITU-T G.826x series of recommendations.



Figure 2: 1588 Topology for Frequency Distribution

For time distribution, it has been recognized that the architecture used above is extremely unlikely to be successful. The fundamental reason is that the performance requirement is much tighter and the network introduces not only PDV but also potentially asymmetric delay which causes time error in the slave. The topology recommended for time distribution is what is sometimes referred to as “Full On-Path Support (OPS)”. Full OPS means that every network element between the grandmaster clock and the slave clock is either a 1588 boundary clock or a 1588 transparent clock, as shown in [Figure 3](#). Boundary clocks and transparent clocks process the 1588 messages and remove the PDV noise that would be present in a non 1588 network element. By using network elements that have very tight constraints on the time error they introduced, the network can be built to guarantee time accuracy under all network traffic conditions. This architecture and the performance requirements are covered by the ITU-T G.827x series of recommendations.

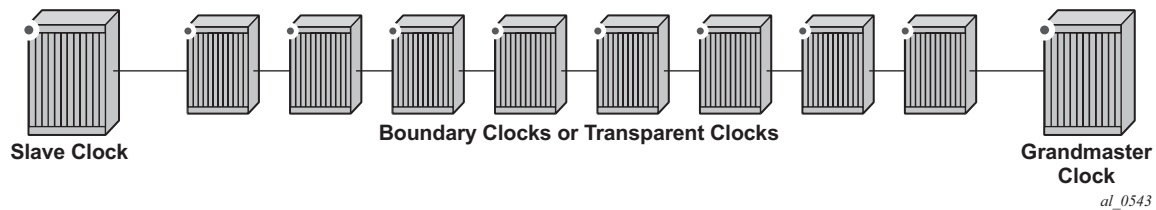


Figure 3: 1588 Topology for Time Distribution

PTP Profiles

The 1588v2 standard includes the concept of a PTP profile. A PTP profile allows standardization bodies or industry groups to adapt the 1588v2 standard to a particular application. A profile defines which aspects of the 1588v2 standard are included or excluded, along with configurable ranges and defaults necessary for the application.

The 1588 standard itself includes a **default** profile that can be used for either time or frequency distribution. The default profile was defined principally for multicast operation. However, it can be used with the unicast sessions as described below. The default profile supports all 1588 clock types and includes the Best Master Clock Algorithm (BMCA) that automatically builds the synchronization distribution hierarchy amongst the PTP clocks. The SR OS only supports the unicast session version of the default profile.

In the telecommunications industry, the ITU-T is the body that develops these profiles. They have generated a profile for frequency distribution (G.8265.1) and a profile for time distribution (G.8275.1). The frequency profile permits only grandmaster and slave clocks and can be used to extend a traditional physical layer synchronization distribution (SONET/SDH, PDH, or SyncE) with a final leg of 1588 messages. The frequency source of the 1588 grandmaster could be a GPS receiver, a central office BITS or SASE device or it could use the frequency recovered from a Synchronous Ethernet or SONET/SDH interface. This is shown in [Figure 4](#)

Because a 1588 distribution system is significantly noisier than a physical layer distribution system, it should only be used as the final segment to connect the end application into the synchronization network. It should not be used to connect two Synchronous Ethernet or SONET/SDH islands.

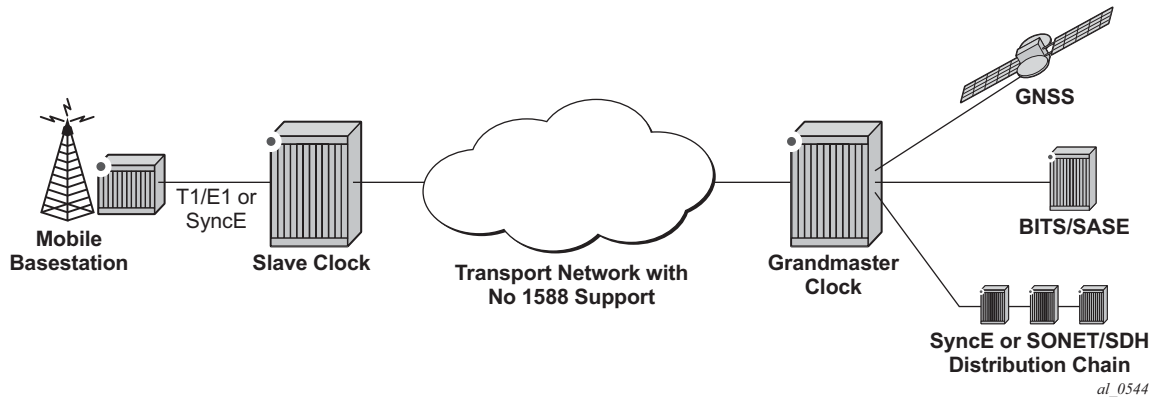


Figure 4: Frequency Distribution with 1588 as Last Mile

The important features defined in the G.8265.1 profile are:

- Only master clocks and slave clocks are allowed.
- Unicast Message Negotiation using Signaling messages from the slave clocks toward the master clocks is used to establish communications.
- PTP messages are encapsulated over UDP over IPv4.
- PTP clock class values are based on a mapping of traditional quality levels from SSM/ESMC³.

The slave clock uses an alternate BMCA to select the grandmaster clock from the available master clocks based on:

- Quality Level.
- Relative Priority.

The ITU-T has defined the first time distribution profile in G.8275.1. It uses an architecture of a Global Navigation Satellite System (GNSS) based grandmaster clock distributing time through a chain of boundary clocks to a final slave device and end application. It includes the use of Synchronous Ethernet and 1588 at the same time for optimal performance. Physical layer Synchronous Ethernet is an excellent tool for the distribution of an accurate and stable frequency. This frequency can be used to advance time between offset adjustments made using the 1588 information.

3. SSM stands for Synchronization Status Messages and ESMC stands for Ethernet Synchronization Messaging Channel. These are two capabilities in SDH/SONET and Synchronous Ethernet respectively for the relaying of source clock quality information.

Unicast Message Negotiation

The initial IEEE 1588-2002 standard defined a multicast messaging model. IEEE 1588-2008 introduced the option of using unicast messaging with unicast discovery to establish a message exchange between a master and slave.

The typical unicast message flow between a master and slave is illustrated in [Figure 5](#).

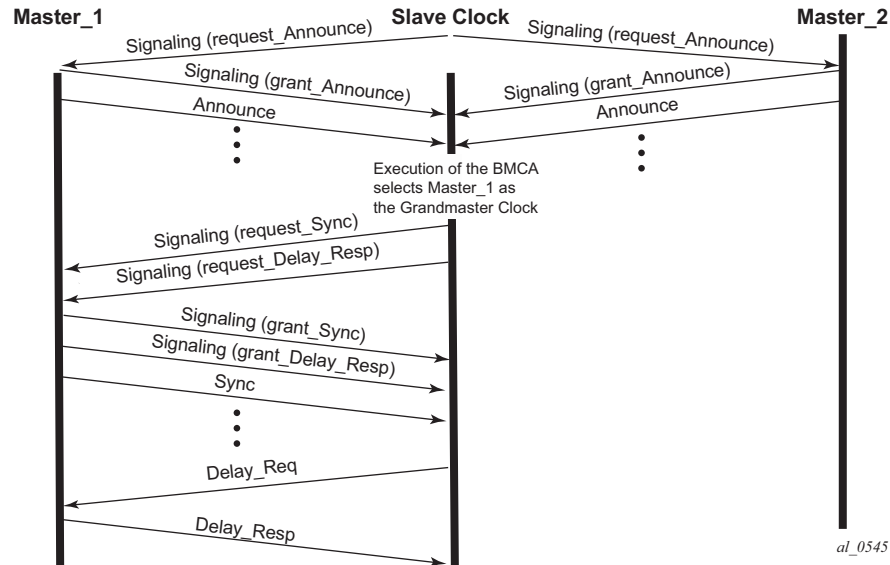


Figure 5: Unicast Message Negotiation

A slave clock initiates unicast discovery by sending a Signaling message to one of its configured master clocks requesting the master send unicast Announce messages to the slave. The request includes the desired rate for the Announce messages and the duration over which the messages should be sent. If the master can support the request it replies with a Signaling message indicating that the session for unicast Announce messages has been granted.

From this point on, the master sends unicast Announce messages to the slave at the rate requested. A slave will generally establish an Announce message session with at least two master clocks.

The slave then uses the Announce messages it receives from all masters as input to the BMCA that determines which master clock is the best source for information. The selected master becomes the grandmaster clock to the slave. The slave then sends additional Signaling messages to the grandmaster to request unicast delivery of Sync and Delay_Resp messages. Assuming the grandmaster clock has sufficient resources, the request is granted and unicast Sync and Delay_Resp messages are sent from the grandmaster to the slave.

As with the Announce messages, the rate at which the Sync and Delay_Resp messages are sent and the duration of the unicast sessions is requested by the slave in the initial Signaling messages.

The unicast sessions for Announce, Sync and Delay Response messages have an expiry time. The slave renews all three sessions before this time is reached.

Network Limits

A common concern around 1588 is whether it will work on or over a specific customer network. For time distribution using full OPS as shown in [Figure 3](#), there are well defined limits on the number of network elements allowed in the distribution chain (see below). However, for the frequency distribution using the architecture shown in [Figure 2](#), it is a more difficult question to answer. There are so many different types of network elements and inter-node links that a simple limit on the number of network elements is not adequate. What has been specified is a limit to the noise that the network can introduce to the 1588 message flow between the grandmaster and slave clocks. This noise occurs as packet delay variation (PDV). The following sections provide some description of this PDV and a new metric that has been defined for PDV as well as the recommended limit to PDV for 1588 deployments.

Packet Delay Variation

If the packet delay through the packet network is constant, then it is relatively easy to use a series of timestamp exchanges to remove the delay as an unknown and track the master clock frequency. However, in most network technologies, the packet delay will be different for each individual packet. This PDV makes it more difficult to track the master clock since observations have both the master information and PDV noise included.

PDV is introduced when packets get placed in queues before they are forwarded. The time each packet sits in any one queue is influenced by multiple factors:

- the speed of the interface toward which the queue drains, for example 100Mbps versus 100Gbps,
- the traffic load on the interface, for example 20% versus 100% of line capacity,
- the distribution of packet sizes and priorities in the traffic load toward the interface, and
- the underlying physical technology used, xPON, xDSL, Ethernet, or microwave.

In addition, the load and packet distribution within the load will vary over time so the distribution of the PDV can shift rapidly such as when a network event triggers congestion or slowly, for example as end customers gradually come online over a period of several hours.

Also there are pipeline effects that can occur in a chain of queuing devices, where the small timing packets can catch up to a large packets moving across the network. Once behind such a packet, the timing packet can remain stuck behind that packet on all subsequent transmit queues.

QoS prioritization of packets helps reduce PDV significantly during congestion periods, but does not remove the PDV effects during lighter loading. This is due to the fact that a timing packet may be delivered to the egress queue for an interface while the interface is busy transmitting a packet. Pre-emption of packet transmissions is not used in today's networks.

Having stated all of the above, most of the time, the network will still present a percentage of packets that get across the network with minimal queuing delays. These are often referred to as 'lucky' or 'fastest' packets. Since these lucky packets are never waiting in queues or have minimal wait times, their transit across the network is relatively consistent. By running a selection filter on all 1588 packets to find these lucky packets, a level of variation of network delay can be removed or reduced significantly. Then the slave clocks have a much easier time determining the frequency of the grandmaster.

However, there will always be a limit to the amount of PDV that can be tolerated. The ITU has defined a metric to quantify the PDV, the limit of the PDV for a compliant network, and the required tolerance of a slave clock.

PDV Metrics

In order to know whether a particular timing-over-packet implementation will meet the performance targets in a given network deployment, it is desirable to both characterize the limits on the PDV that the implementation can tolerate and to measure the network against these limits. In 2012, the ITU-T published three documents that address these requirements:

- G.8260 defines the Floor Packet Percentage (FPP) metric.
- G.8261.1 defines a network limit for PDV in terms of FPP.
- G.8263 defines the input tolerance expected of a 1588 frequency slave in terms of FPP.

The Floor Packet Percentage (FPP) metric provides an indication of the guarantee that there are packets experiencing minimal delay across the network. The rationale behind this focus on 'fastest' packets is that many networks do provide good consistency of these packets in most operating conditions and because most slave clocks are capable of operating using only the information from these fastest packets.

There are four parameters associated with the metric:

- **W** is the width of the windows used to monitor for the presence of fastest packets.
- **Floor Delay** is a value that is as close as possible to the absolute minimum transit delay across the network. Every actual delay measurement must be equal to or larger than this value.

- δ is the range above the floor to be analyzed for the presence of fastest packets.
- ρ is the percentage of all the packets received in a window whose delay must be within the range floor delay to floor delay + δ .

Figure 6 illustrates how these parameters and the metric work. First the delays of all individual 1588 packets are plotted over the period of observation. Next the observation period is broken down into a series of consecutive windows of width **W** seconds. Then for each window a count is made of all the 1588 packets whose delays are within the range **floor delay** to **floor delay** + δ and this count is compared with all the 1588 packets received during the window to turn the count into a percentage. Finally the percentage of each window is checked against the threshold percentage ρ . For the FPP metric to be met, every window must have a percentage greater or equal to the threshold. If even one single window does not meet this threshold then the metric condition is not met.

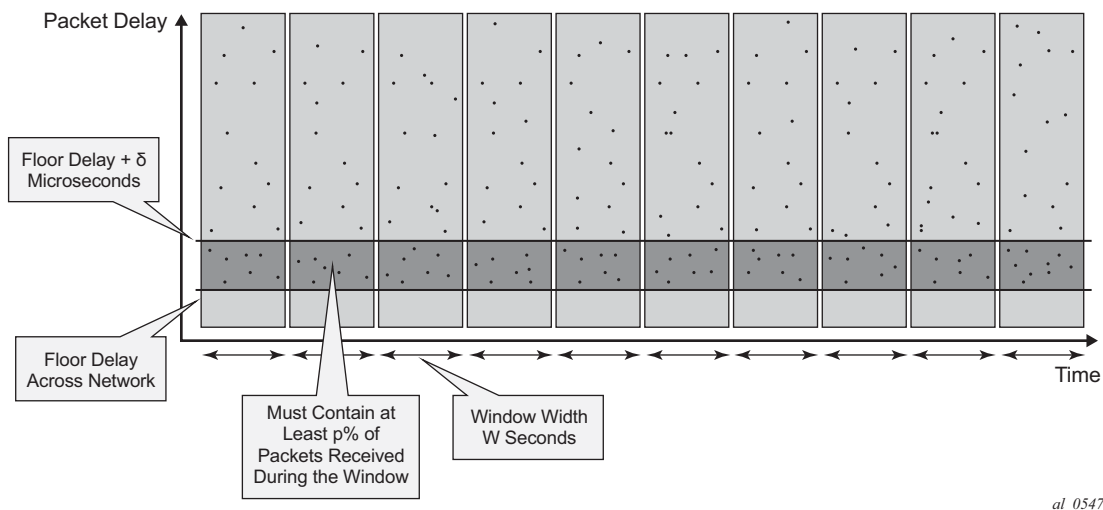


Figure 6: Floor Packet Counting for FPP (n , W , δ)

Note: This metric is not perfect as it does not take into account slaves that use other aspects of the packet delay distribution (such as average delay), nor does it discuss the impact of reroutes, nor do the limits discuss how to apply these limits to the forward and backward message exchanges at the same time. However, it was agreed that this metric was a good start for the definitions of the network and slave limits. Expect to see timing test equipment vendors providing the tools to generate 1588 PDV profiles providing FPP based distributions.

ITU-T Budget for Frequency

The network limit on PDV for frequency distribution is defined in G.8271.1 using the FPP metrics defined above.

In general most carrier grade networks with spans of up to 10 nodes and which do not exceed 80% load on their internode links should meet the requirement. However, very low (sub 50 Mbps) shaping or very long networks or last mile technologies such as xDSL or xPON may need to be studied to determine their acceptability.

A general strategy for rolling out 1588 frequency distribution is to evaluate the specific grandmaster and slave pairing in a lab environment using a network emulator to introduce controlled PDV. Once the grandmaster and slave have passed the lab tests, then field trial locations should be identified. Ideally, the sites should include locations where the PDV of the network will likely be at its worst. This would be sites with the most intervening network elements between the grandmasters and the slaves and include segments of the network that have a high load. The slaves' clocks should be deployed and monitored over several days to ensure that their frequency recovery engines can maintain lock with the grandmasters. During the initial field trials, it is beneficial to use external frequency test equipment at the slave locations to accurately monitor the frequency generated out of the slaves and ensure it stays within limits. As more sites are evaluated and confidence in the PDV environment increases, more deployments can be rolled out. In the deployed network, PTP frequency recovery slave states can be monitored to ensure the solution continues to work.

There may be some locations in the network where the PDV will be too large preventing the slaves to achieve or maintain lock. If it is possible to utilize an alternate network interface to obtain a frequency such as a leased T1 or E1 interface then that could be used. A last resort would be the deployment of a GNSS receiver at the location to provide the frequency reference.

ITU-T Budget for Time

The ITU-T has defined a topology for time distribution based on a full OPS environment. This means that every network element in the time distribution chain is a 1588 clock of some type. Currently the work has defined an environment using Boundary Clocks, but this might be modified in the future to include transparent clocks. The ITU-T tackled the time distribution problem in a more traditional way when compared with the frequency distribution. The ITU-T first defined specific network element clock performance constraints and then defined a longest chain network permitted to ensure that the solution meets the end to end budget. The breakdown of the chain and the budget is shown in [Figure 7](#).

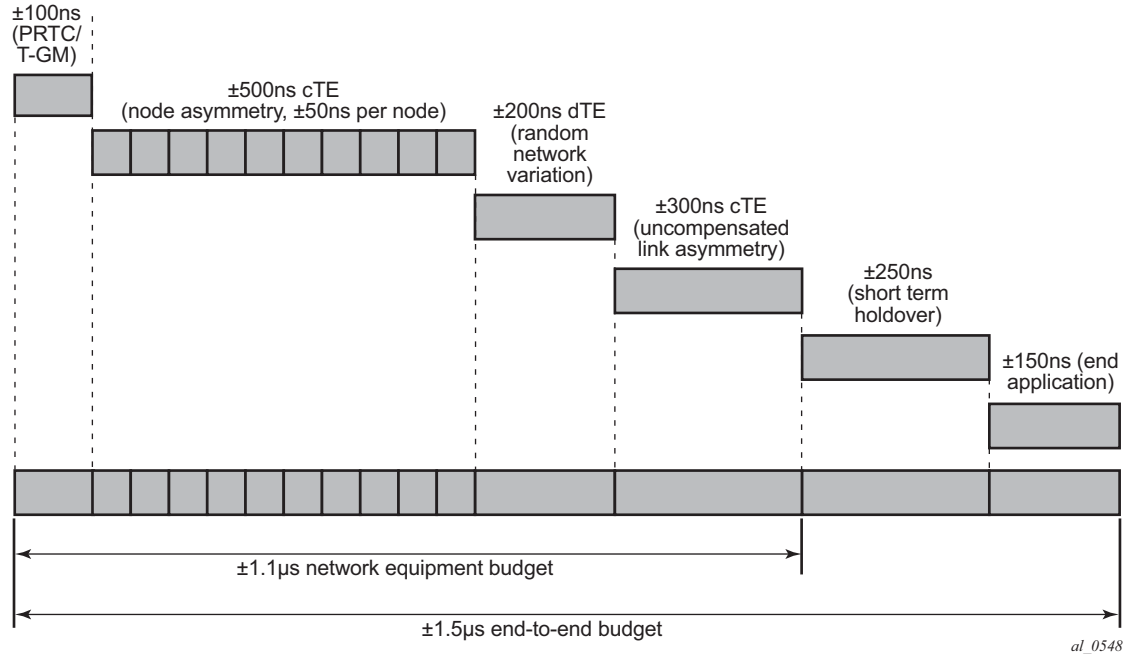


Figure 7: G.8271.1 Time Error Budget

The overall end to end budget is defined as ± 1.5 microseconds. From this the following allocations are made:

- $\pm 100\text{ns}$ Time error due to the GNSS receiver and the 1588 grandmaster.
- $\pm 500\text{ns}$ Constant Time error due to ten Telecom Boundary Clocks (50 ns limit per boundary clock).
- $\pm 200\text{ns}$ Dynamic Time error presented at the end of the boundary clock chain into the end slave.
- $\pm 300\text{ns}$ Time error due to errors in cable latency asymmetry compensation (see below).
- $\pm 150\text{ns}$ Time error due to the end slave and any internals of the basestation between the recovery and the presentation on the air interface.
- $\pm 200\text{ns}$ Time error in the end application during short term holdovers such as network topology re-arrangements.

Note there is discussion that some of these elements could be traded-off against each other. For example, if the link asymmetry needs a higher budget then the holdover budget would have to be less – implying a better end device or a shorter duration of holdover.

The link asymmetries are an important aspect of this budget. The network topology not only has to have the network elements that meet the clock specifications but it also needs to have links that

meet certain requirements. As explained above, the time offset calculation makes the assumptions that the master-to-slave latency is the same from the slave-to-master latency. When the latency is not equal, an error is introduced. Some analysis of network intersite connections may need to be performed to determine the budget for the link asymmetries.

Configuration

IP Addressing for PTP Communication

The system supports communication to the PTP process on the CPM using any of the IPv4 local interface addresses or an IPv4 local loopback addresses. The system will record both the source and destination address information from the received Signaling message which establishes the unicast session. The system will then swap these addresses for use for the Sync and/or Delay_Req messages generated toward the external clock.

The IP address becomes more significant when 1588 port based timestamping is enabled. The port level functionality will filter received PTP packets for a known IP address. This ensures that only PTP messages intended for the node are modified and not PTP messages merely transiting the node.

If the 1588 nodes are directly connected or it is ensured that the PTP messages for a peer shall always enter/exit the system through a single interface, then the IP address of that interface can be used for the PTP message communication. If the PTP messages from a peer could enter through more than one interface, then it may be easier to utilize a loopback address for the PTP message communication.

If using a loopback address and 1588 port based timestamping is also to be used, then the specific loopback address must be assigned to PTP for use using the source-address command. An example is provided in the “Port Based Timestamping” section below. Note: When a source address is defined for the PTP process within a given routing context, then the source address for all Signaling messages originating out of the node within that routing context shall use that address.

Note: The procedures to establish IP connectivity for the specific addresses used in these examples are not included.

Master and Slave Clocks for Frequency

A typical deployment scenario for a system configured as an ordinary master to distribute frequency to an external slave clock, often a cell site router or a base station, is shown in [Figure 8](#). The central clock of the system is locked via its BITS ports or a Synchronous Ethernet port to an external source that is traceable to a primary reference. The frequency of the central clock is used to generate the timestamps contained in PTP event messages. The timestamps generated do not correlate to any standard epoch and therefore indicate an arbitrary timescale. As such it is only the rate of progression of the timestamps that has meaning.

The 7750 SR and the 7450 ESS can be configured as a 1588 slave clock for frequency recovery. In real deployments, it is more likely for the slave devices to be smaller cell site routers or basestations instead of another 7750 SR or 7450 ESS.

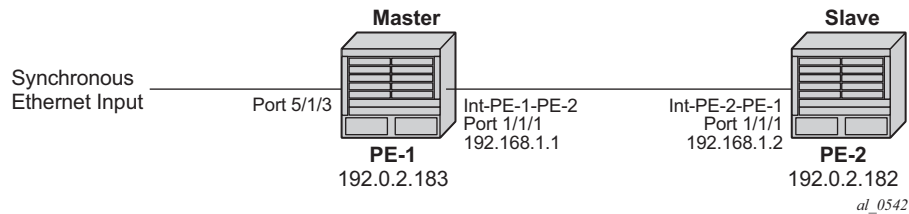


Figure 8: Master and Slave Clocks for Frequency

In the topology in [Figure 8](#), the systems will most likely be configured with the ITU-T G.8265.1 Profile.

For this example, a loopback address is used for PTP communication between the nodes.

Ordinary Master Configuration

The steps to configure PE-1 as a PTP ordinary-clock master for frequency distribution using the G.8265.1 Telecom profile are outlined below:

Configure a /32 IPv4 system address on PE-1 and an interface to reach PE-2.

```
*A:PE-1#
configure
router
    interface "system"
        address 192.0.2.183/32
        no shutdown
    exit
    interface "int-PE-1-PE-2"
        address 192.168.1.1/30
        port 1/1/1
        no shutdown
    exit
exit
```

Configure an input reference for the central clock on PE-1. In this example, Synchronous Ethernet port 5/1/3 is used as the source for **ref2**.

```
*A:PE-1#
configure
port 5/1/3
    description "Sync-E reference for node"
    ethernet
        ssm
            code-type sonet
            no shutdown
        exit
    exit
no shutdown
exit
system
    sync-if-timing
        begin
        ql-selection
        ref2
            source-port 5/1/3
            no shutdown
        exit
    commit
exit
exit
```

The default clock type is set to ordinary slave so that must be changed to ordinary master. The only other relevant configuration parameter for the master clock running the G.8265.1 profile is the network-type. The coding of the SSM/ESMC Quality Level into PTP clock Class must match the environment. The system supports both SONET and SDH networks. The default network-type

is sdh but for this example, the system is configured for the North American market so the network-type is set to sonet.

```
*A:PE-1#
  configure
    system
      ptp
        clock-type ordinary master
        network-type sonet
        no shutdown
      exit
    exit
```

Ordinary Slave Configuration

To configure PE-2 as a PTP ordinary slave for frequency distribution using the G.8265.1 Telecom profile, firstly configure a /32 IPv4 system address on PE-2 and an interface to reach PE-1.

```
*A:PE-2#
  configure
    router
      interface "system"
        address 192.0.2.182/32
        no shutdown
      exit
      interface "int-PE-2-PE-1"
        address 192.168.1.2/30
        port 1/1/1
        no shutdown
      exit
    exit
```

As the default clock type is ordinary slave, PE-1 is configured as a peer clock, and the PTP process is enabled. In this example, the Quality Level encoding is also changed to sonet in order to match the North American market

```
*A:PE-1#
  configure
    system
      ptp
        network-type sonet
        peer 192.0.2.183 create
        no shutdown
      exit
    no shutdown
  exit
```

Usually a 1588 slave has at least two peers configured in order to provide redundant sources.

Configure PTP as the reference for the central clock on PE-2.

```
*A:PE-2#
configure
system
sync-if-timing
begin
ql-selection
ptp
no shutdown
exit
commit
exit
exit
```

Verification of Session Establishment

When PTP is set to no shutdown on PE-2, it initiates a PTP unicast session with PE-1. Correct session establishment can be verified by checking PTP related information as follows:

```
*A:PE-1# show system ptp unicast
=====
IEEE 1588/PTP Unicast Negotiation Information
=====
Router
  IP Address      Dir Type      Rate      Duration State      Time
-----
Base
  192.0.2.182     Tx  Announce  1 pkt/2 s  300      Granted  05/30/2014 06:08:38
  192.0.2.182     Tx  Sync      64 pkt/s   300      Granted  05/30/2014 06:08:43
  192.0.2.182     Rx  DelayReq   64 pkt/s   300      Granted  05/30/2014 06:08:43
  192.0.2.182     Tx  DelayRsp   64 pkt/s   300      Granted  05/30/2014 06:08:43
-----
PTP Peers          : 1
Total Packet Rate  : 192 packets/second
=====
```

```
*A:PE-2# show system ptp unicast
=====
IEEE 1588/PTP Unicast Negotiation Information
=====
Router
  IP Address      Dir Type      Rate      Duration State      Time
-----
Base
  192.0.2.183     Rx  Announce  1 pkt/2 s  300      Granted  05/30/2014 09:08:38
  192.0.2.183     Rx  Sync      64 pkt/s   300      Granted  05/30/2014 09:08:43
  192.0.2.183     Tx  DelayReq   64 pkt/s   300      Granted  05/30/2014 09:08:43
  192.0.2.183     Rx  DelayRsp   64 pkt/s   300      Granted  05/30/2014 09:08:43
-----
PTP Peers          : 1
Total Packet Rate  : 192 packets/second
=====
```

A **Pending** state indicates the system has sent a Unicast Request toward the peer but has not received a response. If the state remains **Pending**, then the IP connectivity between the systems should be verified.

To verify the slave frequency is operating properly, first check the high level information for PTP on PE-2. Note that the PTP Recovery State initially shows phase-tracking and then changes to locked. The time to achieve locked state varies based on the PDV.

```
*A:PE-2# show system ptp
=====
IEEE 1588/PTP Clock Information
=====
-----
Local Clock
-----
Clock Type       : ordinary,slave   PTP Profile      : ITU-T G.8265.1
Domain          : 4                 Network Type     : sonet
Admin State     : up                 Oper State       : up
Announce Interval : 1 pkt/2 s       Announce Rx Timeout: 3 intervals
Peer Limit      : none (Base Router)
Clock Id        : 00233efffe808250  Clock Class      : 255 (slave-only)
Clock Accuracy   : unknown           Clock Variance   : ffff (not computed)
Clock Priority1  : 128                Clock Priority2   : 128
PTP Port State   : slave             Last Changed     : 05/30/2014 09:08:42
PTP Recovery State: phase-tracking   Last Changed     : 05/30/2014 09:08:42
Frequency Offset : -2.704 ppb
-----
Parent Clock
-----
IP Address       : 192.0.2.183       Router           : Base
Parent Clock Id  : 00233efffe69f250 Remote PTP Port  : 1
GM Clock Id      : 00233efffe69f250 GM Clock Class   : 80 (prs)
GM Clock Accuracy : unknown          GM Clock Variance : ffff (not computed)
GM Clock Priority1: 128               GM Clock Priority2 : 128
-----
Time Information
-----
Timescale       : Arbitrary
Current Time    : 2014/05/30 14:12:52.9 (ARB)
Frequency Traceable : yes
Time Traceable  : no
Time Source     : other
=====
```

In addition PTP packet statistics can be checked to verify reception of the PTP messages and the execution of the frequency slave:

```
*A:PE-2# show system ptp statistics
=====
IEEE 1588/PTP Packet Statistics
=====
```

	Input	Output
PTP Packets	5506	2742
Announce	23	0
Sync	2740	0
Follow Up	0	0
Delay Request	0	2740
Delay Response	2740	0
Signaling	3	3
Request Unicast Transmission TLVs	0	3
Announce	0	1
Sync	0	1
Delay Response	0	1
Grant Unicast Transmission (Accepted) TLVs	3	0
Announce	1	0
Sync	1	0
Delay Response	1	0
Grant Unicast Transmission (Denied) TLVs	0	0
Announce	0	0
Sync	0	0
Delay Response	0	0
Cancel Unicast Transmission TLVs	0	0
Announce	0	0
Sync	0	0
Delay Response	0	0
Ack Cancel Unicast Transmission TLVs	0	0
Announce	0	0
Sync	0	0
Delay Response	0	0
Other TLVs	0	0
Other	0	0
Event Packets timestamped at port	0	0
Event Packets timestamped at cpm	2740	2740
Discards	0	0
Bad PTP domain	0	0
Alternate Master	0	0
Out Of Sequence	0	0
Peer Disabled	0	0
Other	0	0

```
=====
IEEE 1588/PTP Frequency Recovery State Statistics
=====
```

State	Seconds
Initial	0
Acquiring	0
Phase-Tracking	43
Locked	0
Hold-over	0

```

=====
IEEE 1588/PTP Event Statistics
=====
Event                                     Sync Flow Delay Flow
-----
Packet Loss                             0             0
Excessive Packet Loss                   0             0
Excessive Phase Shift Detected           0             0
Too Much Packet Delay Variation          0             0
=====
*

```

Secondly, the central clock status on the system can be checked:

```

*A:PE-2# show system sync-if-timing
=====
System Interface Timing Operational Info
=====
System Status CPM B                     : Master Locked
  Reference Input Mode                   : Non-revertive
  Quality Level Selection                 : Disabled
  Reference Selected                     : ptp
  System Quality Level                   : prs
  Current Frequency Offset (ppm)         : +0

Reference Order                          : bits ref1 ref2 ptp

Reference Mate CPM
  Qualified For Use                       : No
    Not Qualified Due To                  :      LOS
  Selected For Use                       : No
    Not Selected Due To                   :      not qualified

Reference Input 1
  Admin Status                           : down
  Rx Quality Level                       : unknown
  Quality Level Override                  : none
  Qualified For Use                       : No
    Not Qualified Due To                  :      disabled
  Selected For Use                       : No
    Not Selected Due To                   :      disabled
  Source Port                            : None

Reference Input 2
  Admin Status                           : down
  Rx Quality Level                       : unknown
  Quality Level Override                  : none
  Qualified For Use                       : No
    Not Qualified Due To                  :      disabled
  Selected For Use                       : No
    Not Selected Due To                   :      disabled
  Source Port                            : None

Reference BITS B
  Input Admin Status                     : down
  Rx Quality Level                       : failed
  Quality Level Override                  : none

```

IP Addressing for PTP Communication

```
Qualified For Use           : No
  Not Qualified Due To      : disabled
Selected For Use           : No
  Not Selected Due To      : disabled
Interface Type             : DS1
Framing                    : ESF
Line Coding                 : B8ZS
Line Length                : 0-110ft
Output Admin Status        : down
Output Source              : line reference
Output Reference Selected  : none
Tx Quality Level           : N/A

Reference PTP
Admin Status               : up
Rx Quality Level           : prs
Quality Level Override     : none
Qualified For Use          : Yes
Selected For Use           : Yes
```

Optional Configuration Items for Ordinary Master or Slave Configuration

The G.8265.1 profile is the default PTP profile on the system and it uses domain number value of 4. The domain number must match at both ends of the communication path or the PTP messages will be dropped. Some very old 1588 devices, may have the domain number set to zero which is the value used by the IEEE1588 default profile. In this case, the system would need to have its domain number changed to match that of the external slave.

```
configure
system
  ptp
    shutdown
    domain 0
    no shutdown
  exit
exit
```

Note that the domain number can only be adjusted if PTP is shutdown and only one common domain number is allowed for all 1588 messages to and from the system.

When using the system as a 1588 slave for frequency distribution, it is strongly recommended to use the default message rate of 64 pps for Sync and Delay_Resp messages. If for some reason the parent 1588 peer cannot offer this rate, then the rate that the system requests must be adjusted. For example, if the maximum rate supported by the external 1588 grandmaster device (with an IP address of 192.0.2.166) only is 32 pps, then the system can be adjusted to request that rate as follows:


```

configure
  system
    ptp
      peer 192.0.2.166 create
      log-sync-interval -5
      no shutdown
    exit
  exit
exit

```

Note that the Sync message rate can only be adjusted if the peer is shutdown.

The message rates are entered as the base 2 logarithm of the inter-message interval. So 32 pps has an inter message interval of 1/32 seconds and a log-sync-interval of -5.

The Announce message rate impact the speed at which PTP can detect communication failures and the speed at which the PTP topology is re-arranged. The default Announce rate is one message every two seconds and this should be adequate for networks with short chains of PTP clocks, for example G.8265.1 architectures. However, in network with longer chains of PTP clocks (for example, more than 5 boundary clocks), it may be desired to use a faster Announce message rate. In the following example, the slave is configured to request two Announce messages per second:

```

configure
  system
    ptp
      shutdown
      log-anno-interval -1
      no shutdown
    exit
  exit
exit

```

Note that the Announce rate can only be adjusted if PTP is shutdown. In addition, there is one common Announce rate for all unicast sessions; it cannot be configured on an individual peer basis.

Boundary Clock

With the increase interest in high accuracy time distribution across networks, the system most likely takes on the role of a 1588 boundary clock. In this role, the system requests time from a GNSS driven grandmaster clock or from a neighboring boundary clock. The system only supports boundary clock configuration when the ptp profile is configured as the default profile.

In this mode of operation, it is strongly recommended to have Synchronous Ethernet physical layer frequency distribution configured at the same time.

The example in [Figure 9](#) shows a boundary clock (PE-1) communicating directly with the GNSS driven grandmaster (GM-1) and a second boundary clock (PE-2) communicating with the first boundary clock.

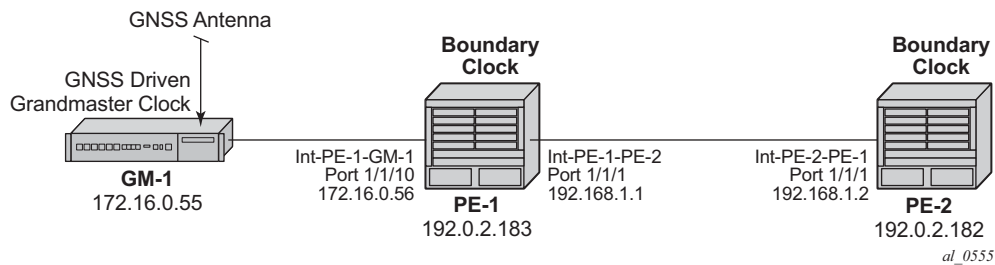


Figure 9: Boundary Clock

The steps to configure the systems as boundary clocks running the IEEE default profile are:

On PE-1, configure a /32 IPv4 system address, an interface to reach PE-2, and an interface to reach GM-1.

```
*A:PE-1#
configure
router
    interface "system"
        address 192.0.2.183/32
        no shutdown
    exit
    interface "int-PE-1-PE-2"
        address 192.168.1.1/30
        port 1/1/1
        no shutdown
    exit
    interface "int-PE-1-GM-1"
        address 172.16.0.56/30
        port 1/1/2
        no shutdown
    exit
exit
```

On PE-2, configure a /32 IPv4 system address and an interface to reach PE-1.

```
*A:PE-2#
configure
router
    interface "system"
        address 192.0.2.182/32
        no shutdown
    exit
    interface "int-PE-2-PE-1"
        address 192.168.1.2/30
        port 1/1/1
        no shutdown
    exit
exit
```

Configure both PE-1 and PE-2 to have physical layer frequency sources into their central clocks. PE-2 is configured to receive Synchronous Ethernet from PE-1 on the same port as is used for PTP. This commonality is not a requirement but might be common in the network topology.

On PE-1, configure the port toward PE-2 as a Synchronous Ethernet port. This will cause the port transmit timing to be sourced from the node timing. Also configure the port to transmit ssm codes using the sonet codes.

```
*A:PE-1#
configure card 1 mda 1 sync-e
configure port 1/1/1 ethernet
    code-type sonet
    no shutdown
exit
```

On PE-2, configure the port on towards PE-1 as a Synchronous Ethernet port and to use sonet codes and to be the reference into the central clock of PE-2.

```
*A:PE-2#
configure card 1 mda 1 sync-e
configure port 1/1/1
    ethernet
        ssm
            code-type sonet
            no shutdown
    exit
exit
configure system sync-it-timing
    begin
    ql-selection
    refl
        source-port 1/1/1
        no shutdown
    exit
commit
```

IP Addressing for PTP Communication

```
exit
```

Next configure PE-1 as a boundary clock requesting service from GM-1 using the default profile. In this example, the interface address of GM-1 is used for the PTP communication.

```
*A:PE-1#
configure system ptp
shutdown
profile ieee1588-2008
clock-type boundary
peer 172.16.0.55 create
no shutdown
exit
no shutdown
exit
```

If it is desired to operate the network at the default for the G.8275.1 profile, then the Announce messages should be set to 8 pps and the Sync and Delay_Resp messages should be set to 16 pps.

```
*A:PE-1#
configure system ptp
shutdown
log-anno-interval -3
peer 172.16.0.55
shutdown
log-sync-interval -4
no shutdown
exit
no shutdown
exit
```

Configure PE-2 as a boundary clock using PE-1 as its parent clock and the same set of 1588 parameters. In this example, PE-2 uses a loopback address of PE-1 for communication.

```
*A:PE-2#
configure system ptp
shutdown
profile ieee1588-2008
clock-type boundary
log-anno-interval -3
peer 192.0.2.183 create
shutdown
log-sync-interval -4
no shutdown
exit
no shutdown
exit
```

On PE-1, validate the status of the PTP topology by checking the unicast sessions. Also validate the PTP process has elected GM-1 as both the parentClock and the grandmaster clock.

```
*A:PE-1# show system ptp unicast
=====
IEEE 1588/PTP Unicast Negotiation Information
=====
Router
  IP Address      Dir Type      Rate      Duration State      Time
-----
Base
  192.0.2.182     Tx  Announce  8 pkt/s    300      Granted  05/30/2014 07:02:36
  192.0.2.182     Tx   Sync     16 pkt/s    300      Granted  05/30/2014 07:02:37
  192.0.2.182     Rx DelayReq  16 pkt/s    300      Granted  05/30/2014 07:02:37
  192.0.2.182     Tx DelayRsp  16 pkt/s    300      Granted  05/30/2014 07:02:37
  172.16.0.55     Rx  Announce  8 pkt/s    300      Granted  05/30/2014 07:02:42
  172.16.0.55     Rx   Sync     16 pkt/s    300      Granted  05/30/2014 07:02:43
  172.16.0.55     Tx DelayReq  16 pkt/s    300      Granted  05/30/2014 07:02:43
  172.16.0.55     Rx DelayRsp  16 pkt/s    300      Granted  05/30/2014 07:02:43
-----
PTP Peers          : 2
Total Packet Rate  : 112 packets/second
=====

*A:PE-1# show system ptp
=====
IEEE 1588/PTP Clock Information
=====
-----
Local Clock
-----
Clock Type       : boundary      PTP Profile      : IEEE 1588-2008
Domain          : 0             Network Type     : sdh
Admin State     : up             Oper State       : up
Announce Interval : 8 pkt/s      Announce Rx Timeout: 3 intervals
Peer Limit      : none (Base Router)
Clock Id        : 00233efffe69f250  Clock Class      : 248 (default)
Clock Accuracy   : unknown        Clock Variance   : ffff (not computed)
Clock Priority1  : 128            Clock Priority2   : 128
PTP Recovery State: locked        Last Changed     : 05/30/2014 07:05:17
Frequency Offset : +50.305 ppb
-----
Parent Clock
-----
IP Address       : 172.16.0.55      Router           : Base
Parent Clock Id  : 8887868584838281 Remote PTP Port  : 1
GM Clock Id      : 8887868584838281 GM Clock Class   : 7
GM Clock Accuracy: within 250 ns   GM Clock Variance: 0x6400 (3.7E-09)
GM Clock Priority1: 128            GM Clock Priority2: 128
-----
Time Information
-----
Timescale       : PTP
Current Time    : 2014/05/30 15:07:01.1 (UTC)
Frequency Traceable : yes
Time Traceable  : yes
```

IP Addressing for PTP Communication

Time Source : GPS

On PE-2, validate the PTP process has elected PE-1 as its parentClock and that the grandmaster clock is GM-1.

```
*A:PE-2# show system ptp
```

```
=====
IEEE 1588/PTP Clock Information
=====
```

```
-----
Local Clock
-----
```

Clock Type	: boundary	PTP Profile	: IEEE 1588-2008
Domain	: 0	Network Type	: sdh
Admin State	: up	Oper State	: up
Announce Interval	: 8 pkt/s	Announce Rx Timeout	: 3 intervals
Peer Limit	: none (Base Router)		
Clock Id	: 00233efffe808250	Clock Class	: 248 (default)
Clock Accuracy	: unknown	Clock Variance	: ffff (not computed)
Clock Priority1	: 128	Clock Priority2	: 128
PTP Recovery State	: locked	Last Changed	: 05/30/2014 10:02:36
Frequency Offset	: -9.345 ppb		

```
-----
Parent Clock
-----
```

IP Address	: 192.0.2.183	Router	: Base
Parent Clock Id	: 00233efffe69f250	Remote PTP Port	: 1
GM Clock Id	: 8887868584838281	GM Clock Class	: 7
GM Clock Accuracy	: within 250 ns	GM Clock Variance	: 0x6400 (3.7E-09)
GM Clock Priority1	: 128	GM Clock Priority2	: 128

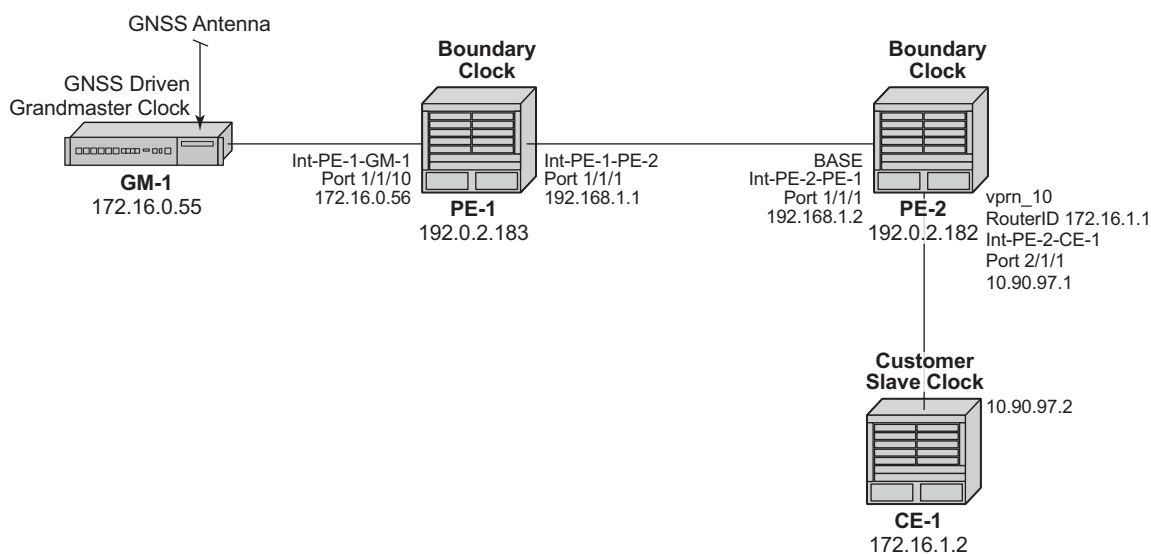
```
-----
Time Information
-----
```

Timescale	: PTP
Current Time	: 2014/05/30 15:09:26.5 (UTC)
Frequency Traceable	: yes
Time Traceable	: yes
Time Source	: GPS

```
=====
```

Boundary Clock with VPRN Access

The system supports access to the 1588 process through Base routing, IES, and VPRN contexts. This permits the system 1588 topology to be created and managed in one context with access for edge distribution through other contexts. For example, building on top of the base routing distribution shown in the previous example, access can be given to the 1588 process on PE-2 via a VPRN existing on that node. This allows the VPRN customer to have access to the high accuracy time available within the system in the customer edge equipment connecting into that node.



al_0556

Figure 10: Boundary Clocks with Edge VPRN Access

For the example shown in [Figure 10](#), it is assumed that a VPRN service is already configured and operational on PE-2 providing connectivity between PE-2 and CE-1:

```
*A:PE-2#
configure service vprn 10 customer 1 create
  router-id 176.16.1.1
  autonomous-system 64496
  route-distinguisher 64496:10
  interface "int-PE-2-CE-1" create
    address 10.90.97.1/30
    sap 2/1/1 create
  exit
exit
no shutdown
exit
```

IP Addressing for PTP Communication

To enable access to the PTP process via VPRN 10 in PE-2, PTP must be enabled within the VPRN context. To ensure that no more than 10 external clocks access the system PTP through this VPRN at any one time, a peer-limit may be defined.

```
*A:PE-2#
    configure service vprn 10
        peer-limit 10
        ptp no shutdown
    exit
```

To confirm PTP access with the VPRN, the PTP information with the VPRN context can be queried. Either of the following two commands can be used:

```
*A:PE-2# show system ptp unicast router 10
```

or

```
*A:PE-2# show service id 10 ptp unicast
```

These two commands provide the same information as shown below.

```
*A:PE-2# show system ptp unicast router 10
=====
IEEE 1588/PTP Unicast Negotiation Information
=====
Router
  IP Address      Dir Type      Rate      Duration State      Time
-----
10
  172.16.1.2      Tx  Announce 1 pkt/2 s  300      Granted 05/30/2014 12:40:53
  172.16.1.2      Tx   Sync    64 pkt/s   300      Granted 05/30/2014 12:40:59
  172.16.1.2      Rx  DelayReq 64 pkt/s   300      Granted 05/30/2014 12:40:59
  172.16.1.2      Tx  DelayRsp 64 pkt/s   300      Granted 05/30/2014 12:40:59
-----
PTP Peers          : 1
Total Packet Rate   : 192 packets/second
=====
```


Port Based Timestamping

As described above, optimal performance is achieved when the 1588 port based timestamping (PBT) feature is used. This feature is not available on all hardware and the interfaces for PTP should be planned in advance if this feature is to be used.

Since 1588 messages ingress and egress the node through router interfaces, the configuration of the 1588 PBT feature is enabled within the router interface context. In the previous examples, if 1588 PBT is to be enabled on all the PTP interfaces the following commands are required.

On PE-1, enable 1588 PBT on the interface toward GM-1 and PE-2.

```
*A:PE-1#
  configure
    router
      interface "int-PE-1-PE-2"
        ptp-hw-assist
      exit
      interface "int-PE-1-GM-1"
        ptp-hw-assist
      exit
    exit
```

On PE-2, enable 1588 PBT on the interface toward PE-1 and CE-1.

```
*A:PE-2#
  configure
    router
      interface "int-PE-2-PE-1"
        ptp-hw-assist
      exit
    exit
  exit
  configure service vprn 10 customer 1
    interface "int-PE-2-CE-1"
      ptp-hw-assist
    exit
  exit
```

To verify 1588 PBT is active on the 1588 messages to the peers, check the timestamp point for the specific peer. It now indicates *port* rather than *cpm*.

On PE-1 for the CE-1 communication:

```
*A:PE-1# show system ptp peer 172.16.0.55
=====
IEEE 1588/PTP Peer Information
=====
Router           : Base
IP Address       : 172.16.0.55      Announce Direction : rx
Admin State      : up              G.8265.1 Priority   : n/a
Sync Interval    : 16 pkt/s
Local PTP Port   : 2               PTP Port State     : slave
```

IP Addressing for PTP Communication

```
Clock Id          : 8887868584838281  Remote PTP Port   : 1
GM Clock Id       : 8887868584838281  GM Clock Class    : 7
GM Clock Accuracy : within 250 ns      GM Clock Variance : 0x6400 (3.7E-09)
GM Clock Priority1: 128                GM Clock Priority2 : 128
Steps Removed     : 0                 Parent Clock       : yes
Tx Timestamp Point: port              Rx Timestamp Point: port
Last Tx Port      : 5/1/1             Last Rx Port       : 5/1/1
=====
```

On PE-1 the communication with the PE-2 will still be CPM timestamping since the port has not been configured to watch for the 'system' loopback address.

```
*A:PE-1# show system ptp peer 192.0.2.182
=====
IEEE 1588/PTP Peer Information
=====
Router          : Base
IP Address       : 192.0.2.182         Announce Direction : tx
Admin State      : n/a                 G.8265.1 Priority   : n/a
Sync Interval    : n/a
Local PTP Port   : 3                   PTP Port State      : master
Clock Id         : 00233efffe808250   Remote PTP Port     : 4
Tx Timestamp Point: cpm                 Rx Timestamp Point  : cpm
Last Tx Port     : 5/1/2               Last Rx Port        : 5/1/2
=====
```

In order to configure the **system** loopback address for PTP, enter the following on PE-1:

```
*A:PE-1#
      configure
      system security
      source-address application ptp "system"
      exit
      exit
```

Now the timestamp point on PE-1 will be the port.

```
*A:PE-1# show system ptp peer 192.0.2.182
=====
IEEE 1588/PTP Peer Information
=====
Router          : Base
IP Address       : 192.0.2.182         Announce Direction : tx
Admin State      : n/a                 G.8265.1 Priority   : n/a
Sync Interval    : n/a
Local PTP Port   : 3                   PTP Port State      : master
Clock Id         : 00233efffe808250   Remote PTP Port     : 4
Tx Timestamp Point: port                 Rx Timestamp Point  : port
Last Tx Port     : 5/1/2               Last Rx Port        : 5/1/2
=====
```

Repeat this configuration of system address for the base routing context on PE-2

```
*A:PE-2#
configure
system security
source-address application ptp "system"
exit
exit
```

Now the timestamp point on PE-2 will be the port.

```
*A:PE-2# show system ptp peer 192.0.2.183
=====
IEEE 1588/PTP Peer Information
=====
Router           : Base
IP Address       : 192.0.2.183      Announce Direction : rx
Admin State      : up              G.8265.1 Priority   : n/a
Sync Interval    : 16 pkt/s
Local PTP Port   : 4               PTP Port State      : slave
Clock Id         : 00233efffe69f250 Remote PTP Port     : 3
GM Clock Id      : 8887868584838281 GM Clock Class       : 6
GM Clock Accuracy : within 100 ns   GM Clock Variance    : 0x6400 (3.7E-09)
GM Clock Priority1: 128              GM Clock Priority2    : 128
Steps Removed    : 1               Parent Clock         : yes
Tx Timestamp Point: port            Rx Timestamp Point   : port
Last Tx Port     : 1/1/2            Last Rx Port         : 1/1/2
=====
```

On PE-2, a loopback address must assigned for PTP communication as follows:

```
*A:PE-2#
configure service vprn 10
interface "ptp_loopback"
address 172.16.1.1/32
loopback
exit
source-address
application ptp "ptp_loopback"
exit
exit
```

1588 as NTP Local Clock (server)

If the system is configured as a boundary clock or slave clock then the time recovered from the 1588 slave port can be used as the source of system time on the node. This allows for higher accuracy and better stability in the timebase when compared to NTP. To enable this, PTP must be made the preferred server in the NTP context in the node.

Note that if the system is acting as an NTP server or peer to other NTP clocks, then turning on this feature will impact the existing NTP topology. The system shall advertise itself as an NTP Stratum 1 server to external clients and peers. Given the much higher accuracies achievable with PTP time distribution, this change in topology does not degrade the time in the clients and peers.

```
*A:PE-1#
      configure system time ntp
        server ptp prefer
      exit
```

To validate PTP is now being used for NTP time and system time, use the following command:

```
*A:PE-1# show system ntp all
=====
NTP Status
=====
Configured           : Yes           Stratum              : 1
Admin Status         : up            Oper Status           : up
Server Enabled        : No            Server Authenticate   : No
Clock Source          : ptp
Auth Check            : Yes
Current Date & Time: 2014/05/30 17:53:11 UTC
=====
NTP Active Associations
=====
State                Reference ID   St Type  A  Poll Reach   Offset(ms)
-----
Remote
-----
chosen                PTP           0  srvr  -  64   .....YY  0.000
ptp
=====
NTP Clients
=====
vRouter                                     Time Last Request Rx
Address
-----
=====
```

Conclusion

The systems provide support for IEEE 1588 frequency and time distribution for the synchronization applications of the mobile networks. They can be configured as frequency distribution grandmasters and slave clocks or time distribution boundary and slave clocks.

Multi-Chassis Synchronization for IGMP Snooping

In This Chapter

This section provides information about multi-chassis synchronization (MCS) for IGMP snooping.

Topics in this section include:

- [Applicability on page 60](#)
- [Overview on page 61](#)
- [Configuration on page 63](#)
- [Conclusion on page 75](#)

Applicability

This example is applicable to all 7x50 and 7710 platforms and was tested on release 12.0.R1.

There are no hardware dependencies for MCS.

Overview

Multi-Chassis Synchronization (MCS) is a proprietary protocol used for synchronizing state information between two 7x50/7710 SR peers. MCS can be used for the following applications:

- IGMP
- IGMP snooping
- MLDP snooping
- IPSec
- DHCP server
- Subscriber management
- Subscriber Router Redundancy Protocol (SRRP)

The focus of this example is on using MCS to synchronize IGMP snooping state information, that is, Layer 2 multicast forwarding entries called IGMP snooping entries, between two peer nodes.

MCS also supports synchronization of IGMP state information on IGMP-enabled router interfaces, but that is outside the scope of this example.

Standard behavior in an IGMP snooping-enabled Layer 2 domain is for the system to send the IGMP reports from the IGMP receivers towards the IGMP querier along the multicast router (Mrouter) ports. The Mrouter ports are either dynamically elected (because they are on the Layer 2 logical path to the IGMP querier) or statically configured on ports in the Layer 2 domain (either on the VPLS SAP or on the SDP).

As IGMP reports are forwarded towards the IGMP querier through the Mrouter ports, multicast forwarding entries are created in the data plane for each (S,G), or (*,G) separately. This data is stored as multicast forwarding information base (mFIB) entries in the IGMP snooping-enabled VPLS service.

Because the Layer 2 path between the IGMP querier and the IGMP receivers can change over time due to Layer 2 network failures, the multicast service (IGMP traffic and multicast streams) can be disrupted. To minimize the outage of the multicast service when the path between the IGMP receivers and IGMP querier changes between redundant nodes, MCS copies and maintains the multicast forwarding entries in the data plane between redundant nodes. It then activates the entries on the standby node when it becomes active in the event of a Layer 2 network failure. This ensures that IGMP and multicast traffic continues to flow with minimal loss. Traffic converges faster and independently from the recovery mechanism provided by the IGMP querier, which is controlled through the IGMP query timer value.

Overview

It is worth noting that a very similar configuration (but using IGMP instead of IGMP snooping as an MCS application) can also be used to synchronize IGMP states on a Layer 3 multicast interface. This can be one way of speeding up multicast convergence; for example, if there is a failure of a PIM designated router (which is also an IGMP querier).

Configuration

The benefits of MCS for IGMP snooping are demonstrated by comparing two scenarios: one without and one with MCS for IGMP snooping.

The redundant Layer 2 path between the IGMP querier (and the multicast source) and the IGMP receiver for both scenarios can be managed through a number of mechanisms, for example Spanning Tree Protocol (STP), G.8032 or LAG.

Configuring redundant Layer 2 paths is outside the scope of this example.

For both scenarios, the same spanning tree and VPLS snooping configuration is used.

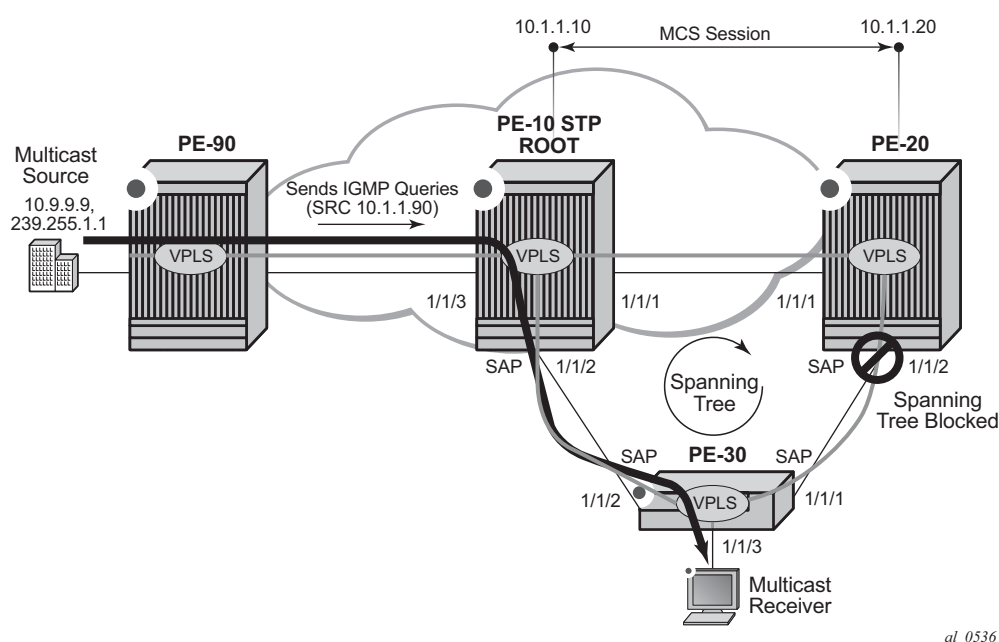


Figure 11: Configuration without MCS for IGMP Snooping

The baseline configuration common to both scenarios is shown in [Figure 11](#).

The nodes PE-10, PE-20, PE-30 and PE-90 are part of the Layer 2 domain, and all have a VPLS with a set of SAPs. PE-10 is the root of the STP topology, and all the SAPs in this topology are in the forwarding state except for SAP 1/1/2:800 on PE-20. This SAP is pruned from the Layer 2 topology and is therefore in the blocked state as the following output shows.

Configuration

```
*A:PE-20# show service id 800 sap
=====
SAP(Summary), Service 800
=====
PortId          SvcId
  Ing.  Ing.    Egr.  Egr.  Adm  Opr
                               QoS  Fltr   QoS  Fltr
-----
1/1/1:800          800    1   none   1   none  Up   Up
1/1/2:800          800    1   none   1   none  Up  Prun
-----
```

VPLS service 800 is configured on all Layer 2 nodes with IGMP snooping enabled, with the following properties:

- The multicast source is connected to PE-90.
- The multicast receiver is connected to PE-30, SAP 1/1/3:800.
- IGMP snooping is enabled in all VPLSs on PE-10, PE-20, PE-30, and PE-90.
- PE-90 sends IGMP queries with source-address 10.1.1.90, as it is enabled on the VPLS SAP.

Configuration of PE-90 with VPLS 800 sending queries on SAP 1/1/3 towards PE-10:

```
*A:7750-90>config>service>vpls# info
-----
      stp
        shutdown
      exit
      igmp-snooping
        query-src-ip 10.1.1.90
        no shutdown
      exit
      sap 1/1/3:800 create
        igmp-snooping
          send-queries
        exit
      exit
      no shutdown
```

Configuration of MCS without IGMP snooping

For verification purposes, the IGMP querier status is checked on every node that is part of the Layer 2 domain.

PE-10 VPLS 800 IGMP querier status:

```
*A:PE-10# show service id 800 igmp-snooping querier
=====
IGMP Snooping Querier info for service 800
=====
Sap Id           : 1/1/3:800
IP Address       : 10.1.1.90
Expires          : 171s
Up Time          : 19d 21:59:26
Version          : 2

General Query Interval : 125s
Query Response Interval : 10.0s
Robust Count          : 2
=====
```

PE-20 VPLS 800 IGMP querier status:

```
*A:PE-20# show service id 800 igmp-snooping querier
=====
IGMP Snooping Querier info for service 800
=====
Sap Id           : 1/1/1:800
IP Address       : 10.1.1.90
Expires          : 132s
Up Time          : 19d 22:00:51
Version          : 2

General Query Interval : 125s
Query Response Interval : 10.0s
Robust Count          : 2
=====
```

PE-30 VPLS 800 IGMP querier status:

```
*A:PE-30# show service id 800 igmp-snooping querier
=====
IGMP Snooping Querier info for service 800
=====
Sap Id           : 1/1/2:800
IP Address       : 10.1.1.90
Expires          : 154s
Up Time          : 0d 00:05:50
Version          : 2

General Query Interval : 125s
Query Response Interval : 10.0s
```

Configuration of MCS without IGMP snooping

Robust Count : 2

Because of the placement of the IGMP queriers, as well as the IGMP receiver and the STP forwarding status of SAP, the mFIB of the three PE's is as shown below (when MCS IGMP snooping is not configured).

PE-10 VPLS 800 mFIB:

```
*A:PE-10# show service id 800 mfib
=====
Multicast FIB, Service 800
=====
Source Address  Group Address      Sap/Sdp Id          Svc Id  Fwd/Blk
-----
*               *               sap:1/1/3:800       Local   Fwd
*               239.255.1.1    sap:1/1/2:800       Local   Fwd
*               sap:1/1/3:800       Local   Fwd
-----
```

PE-20 VPLS 800 mFIB:

```
*A:PE-20# show service id 800 mfib
=====
Multicast FIB, Service 800
=====
Source Address  Group Address      Sap/Sdp Id          Svc Id  Fwd/Blk
-----
*               *               sap:1/1/1:800       Local   Fwd
-----
Number of entries: 1
=====
```

PE-30 VPLS 800 mFIB:

```
*A:PE-30>show>service>id>igmp-snooping# show service id 800 mfib
=====
Multicast FIB, Service 800
=====
Source Address  Group Address      Sap/Sdp Id          Svc Id  Fwd/Blk
-----
*               *               sap:1/1/2:800       Local   Fwd
*               239.255.1.1    sap:1/1/3:800       Local   Fwd
*               sap:1/1/2:800       Local   Fwd
-----
```

The mFIB for VPLS 800 on PE-10 shows the following:

- The (*,*) multicast entry is present in the mFIB for SAP 1/1/3:800 because IGMP queries (from PE-90) are received on that SAP (which now is an Mrouter port).
- The (*,239.255.1.1) multicast entry is present in the mFIB for SAP 1/1/2:800 because an IGMP report has been received for that group on that SAP from the IGMP receiver.
- The (*,239.255.1.1) multicast entry is also present in the mFIB for SAP 1/1/3:800 because this is the Mrouter port for the IGMP querier connected to that SAP.

The mFIB for VPLS 800 on PE-20 shows the following:

- The (*,*) multicast entry is present in the mFIB on SAP 1/1/1:800 because IGMP queries (from PE-10) are received on that SAP (which now is an Mrouter port).

The mFIB for VPLS 800 on PE-30 shows the following:

- The (*,*) multicast entry is present in the mFIB for SAP 1/1/2:800 because IGMP queries (from PE-10) are received on that SAP (which now is an Mrouter port).
- The (*,239.255.1.1) multicast entry is present in the mFIB for SAP 1/1/3:800 because an IGMP report has been received for that group on that SAP (connects to the receiver).
- The (*,239.255.1.1) multicast entry is also present in the mFIB for SAP 1/1/2:800 because that SAP is an Mrouter port.

If a link failure occurs between PE-10 and PE-30, the spanning tree converges and SAP 1/1/2:800 of VPLS 800 on PE-20 transitions to the forwarding state (“up”).

```
*A:PE-20# show service id 800 sap
=====
SAP(Summary), Service 800
=====
```

PortId	SvcId	Ing. QoS	Ing. Fltr	Egr. QoS	Egr. Fltr	Adm	Opr
1/1/1:800	800	20	none	1	none	Up	Up
1/1/2:800	800	20	none	1	none	Up	Up

```
-----
```

Immediately after the link failure, while the spanning tree is converging, the mFIBs on PE-10, PE-20 and PE-30 are exactly the same as before the failure.

The mFIB for VPLS 800 on PE-10 looks as follows:

```
*A:PE-10# show service id 800 mfib
=====
Multicast FIB, Service 800
=====
```

Source Address	Group Address	Sap/Sdp Id	Svc Id	Fwd/Blk
----------------	---------------	------------	--------	---------

```
=====
```

Configuration of MCS without IGMP snooping

```
-----
*          *          sap:1/1/3:800          Local    Fwd
*          239.255.1.1  sap:1/1/2:800          Local    Fwd
*                               sap:1/1/3:800          Local    Fwd
-----
```

The mFIB for VPLS 800 on PE-20 looks as follows:

```
*A:PE-20# show service id 800 mfib
=====
Multicast FIB, Service 800
=====
Source Address  Group Address          Sap/Sdp Id          Svc Id  Fwd/Blk
-----
*              *              sap:1/1/1:800          Local    Fwd
-----
Number of entries: 1
=====
```

The mFIB for VPLS 800 on PE-30 looks as follows:

```
*A:PE-30>show>service>id>igmp-snooping# show service id 800 mfib
=====
Multicast FIB, Service 800
=====
Source Address  Group Address          Sap/Sdp Id          Svc Id  Fwd/Blk
-----
*              *              sap:1/1/2:800          Local    Fwd
*              239.255.1.1  sap:1/1/3:800          Local    Fwd
*                               sap:1/1/2:800          Local    Fwd
-----
```

This means that the multicast stream has not recovered yet.

For the multicast stream to reconverge, spanning tree must re-converge and an IGMP query must be received on PE-30 (sourced by the PE-90 querier) via the new Layer 2 forwarding path between the source and the receiver, which now runs via PE-20.

When PE-30 receives an IGMP query on the new Layer 2 forwarding path (via SAP 1/1/1:800) it makes this SAP an Mrouter port and this triggers an IGMP report to be sent across this newly elected Mrouter port to the querier.

The IGMP report results in a change to the mFIBs along the new Layer 2 multicast forwarding path. At that time, the mFIBs in all three PEs will look as shown below.

The mFIB for VPLS 800 on PE-10 looks as follows:

```
*A:PE-10>config>port# show service id 800 mfib
=====
Multicast FIB, Service 800
=====
Source Address  Group Address          Sap/Sdp Id          Svc Id  Fwd/Blk
-----
```



```

-----
*                *                sap:1/1/3:800                Local    Fwd
*                239.255.1.1      sap:1/1/1:800                Local    Fwd
*                                sap:1/1/3:800                Local    Fwd
-----

```

The mFIB for VPLS 800 on PE-20 looks as follows:

```

*A:PE-20# show service id 800 mfib
=====
Multicast FIB, Service 800
=====
Source Address  Group Address          Sap/Sdp Id                Svc Id  Fwd/Blk
-----
*                *                sap:1/1/1:800            Local    Fwd
*                239.255.1.1      sap:1/1/1:800            Local    Fwd
*                                sap:1/1/2:800            Local    Fwd
-----

```

The mFIB for VPLS 800 on PE-30 looks as follows:

```

*A:PE-30>show# show service id 800 mfib
=====
Multicast FIB, Service 800
=====
Source Address  Group Address          Sap/Sdp Id                Svc Id  Fwd/Blk
-----
*                *                sap:1/1/1:800            Local    Fwd
*                239.255.1.1      sap:1/1/1:800            Local    Fwd
*                                sap:1/1/3:800            Local    Fwd
-----

```

The presence of an mFIB entry for group (*,239.255.1.1) on SAP 1/1/1:800 of PE-10 and SAP 1/1/2:800 of PE-20 means that the multicast stream has now recovered.

Because the recovery of multicast streams (IGMP query triggered) depend highly on the IGMP query timer, recovery is slow.

The main purpose of multi-chassis IGMP snooping synchronization is to reduce recovery time.

Configuration of MCS with IGMP Snooping

Figure 12 shows the MCS configuration used for IGMP snooping; the only difference from Figure 11 is that now MCS is configured between PE-10 and PE-20.

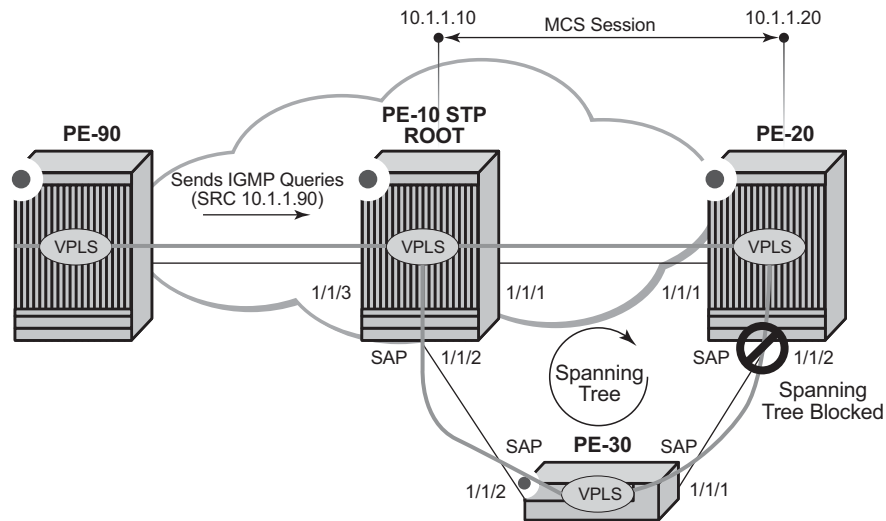


Figure 12: Configuration with MCS for IGMP Snooping

The MCS IGMP snooping configuration on PE-10 is shown below.

```
A:PE-10>config>redundancy# info
-----
multi-chassis
  peer 10.1.1.20 create
  sync
    igmp-snooping
    port 1/1/2 create
      range 800-800 sync-tag "igmp800"
    exit
  no shutdown
exit
no shutdown
exit
exit
```

The MCS IGMP snooping configuration on PE-20 is shown below.

```
*A:PE-20>config>redundancy>multi-chassis# info
-----
peer 10.1.1.10 create
  sync
    igmp-snooping
    port 1/1/2 create
      range 800-800 sync-tag "igmp800"
    exit
  no shutdown
exit
no shutdown
exit
```

This configuration results in IGMP snooping states being synchronized between PE-10 and PE-20 for port 1/1/2 VLAN 800 (VPLS 800) on PE-10 and for port 1/1/2 VLAN 800 (VPLS 800) on PE-20.

Synchronization happens for qtag 800 between these 2 ports (1/1/2 of PE10 and 1/1/2 of PE10) because both are using same sync-tag igmp800. The sync-tag parameter allows to synchronize IGMP contexts between 2 chassis independently from port and Qtag numbering.

The MCS session state can be verified on PE-10 using the following command:

```
*A:PE-10>show>redundancy>multi-chassis# all
=====
Multi-Chassis Peers
=====
```

Peer IP Src IP	Peer Admin Auth	Client	Admin	Oper	State
10.1.1.20	Enabled	MC-Sync:	Enabled	Enabled	inSync
10.1.1.10	None	MC-Ring:	--	--	--
		MC-Endpt:	--	--	--
		MC-Lag:	Disabled	Disabled	--
		MC-IPsec:	--	--	Disabled

```
=====
```

Because the operational state is Enabled and State is inSync, the IGMP states for port 1/1/2 VLAN 800 on PE-20 are now synchronized with port 1/1/2 VLAN 800 on PE-10.

This can be observed on PE-20:

```
*A:PE-20# show redundancy multi-chassis sync peer 10.1.1.10 detail
=====
Multi-chassis Peer Table
=====
Peer
-----
Peer IP Address      : 10.1.1.10
Description          : (Not Specified)
Authentication       : Disabled
```

Configuration of MCS with IGMP Snooping

```
Source IP Address      : 10.1.1.20
Admin State            : Enabled
-----
Sync-status
-----
Client Applications    : IGMP Snooping
Sync Admin State       : Up
Sync Oper State        : Up
Sync Oper Flags        :
DB Sync State          : inSync
Num Entries            : 1
Lcl Deleted Entries    : 0
Alarm Entries          : 0
OMCR Standby Entries   : 0
OMCR Alarm Entries     : 0
Rem Num Entries        : 1
Rem Lcl Deleted Entries : 0
Rem Alarm Entries      : 0
Rem OMCR Standby Entries : 0
Rem OMCR Alarm Entries : 0
=====
..snip ..
```

The contents of the MCS database for the IGMP snooping application can be displayed on PE-20 using the following command:

```
*A:PE-20# tools dump redundancy multi-chassis sync-database application igmp-snooping
detail
```

If no entries are present for an application, no detail will be displayed.

FLAGS LEGEND: ld - local delete; da - delete alarm; pd - pending global delete;
oal - omcr alarmed; ost - omcr standby

```
Peer Ip 10.1.1.10
```

```
Application IGMP Snooping
Sap-id      Client Key
SyncTag      deleteReason code and description
DLen  Flags      timeStamp
-----
1/1/2:800    Group=239.255.1.1
igmp800      10    -- -- -- -- 05/22/2014 13:57:28
0x0
```

The following totals are for:

peer ip ALL, port/lag ALL, sync-tag ALL, application IGMP Snooping

```
Valid Entries:          1
Locally Deleted Entries: 0
Locally Deleted Alarmed Entries: 0
Pending Global Delete Entries: 0
Omcr Alarmed Entries:   0
Omcr Standby Entries:   0
```

Note that an IGMP-snooping MCS entry for group 239.255.1.1 on SAP 1/1/2:800 exists, which means that there is also an IGMP mFIB entry in the data plane on PE-20, as shown in the output below.

```
*A:PE-20# show service id 800 mfib
=====
Multicast FIB, Service
800=====
Source Address  Group Address      Sap/Sdp Id          Svc Id  Fwd/Blk
-----
*               *               sap:1/1/1:800        Local    Fwd
*               239.255.1.1  sap:1/1/1:800        Local    Fwd
*               sap:1/1/2:800        Local    Fwd
-----
Number of entries: 2
=====
```

As well as an mFIB entry for group 239.255.1.1 on SAP 1/1/2:800 being created, PE-20 also automatically sends IGMP reports for that group towards the IGMP querier. This will create an additional mFIB entry for group 239.255.1.1 on SAP 1/1/1:800 on PE-10 (unlike the non-MCS-enabled scenario), as shown in the output below.

```
*A:PE-10# show service id 800 mfib
=====
Multicast FIB, Service 800
=====
Source Address  Group Address      Sap/Sdp Id          Svc Id  Fwd/Blk
-----
*               *               sap:1/1/3:800        Local    Fwd
*               239.255.1.1  sap:1/1/1:800        Local    Fwd
*               sap:1/1/3:800        Local    Fwd
*               sap:1/1/2:800        Local    Fwd
-----
```

This means that the multicast stream not only is forwarded to PE-30 (via SAP 1/1/2:800 on PE-10) but also to PE-20 (via PE-10 SAP 1/1/1:800). However, PE-20 discards the stream as SAP 1/1/2:800 is blocked by the spanning tree.

Configuration of MCS with IGMP Snooping

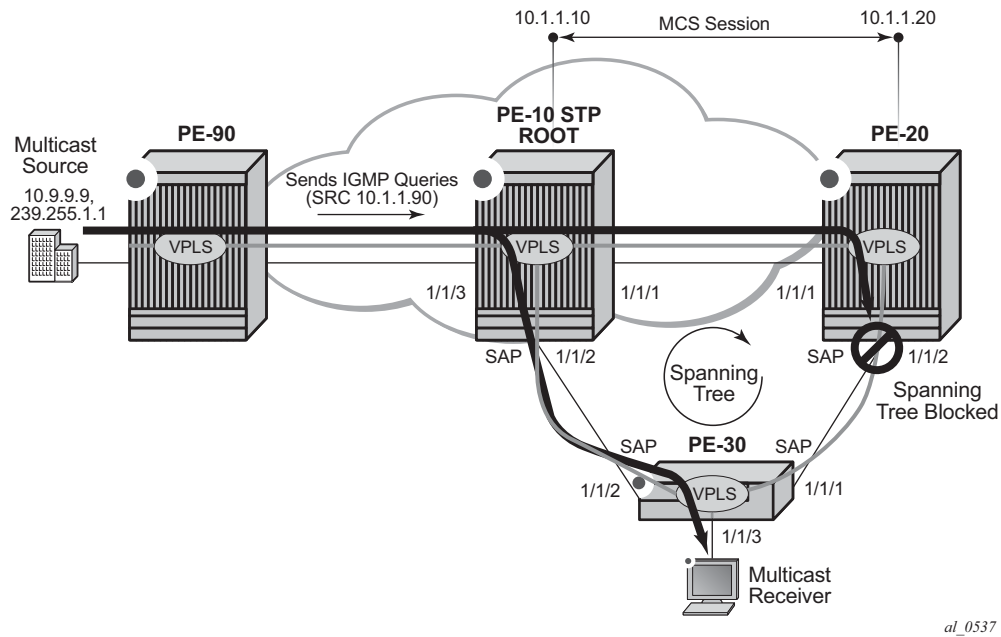


Figure 13: Multicast Stream Forwarded to PE-20

In the first scenario where MCS is not enabled, if port 1/1/2 on PE-10 fails, then SAP 1/1/2:800 on PE-20 will transition to the forwarding state once the spanning tree reconverges. However, it is only after the reception of an IGMP report from PE-30, which can take some seconds, that PE-20 sends the multicast traffic to PE-30.

Enabling MCS for IGMP snooping results in snooping entries being populated in the mFIB of PE-20 (notably on SAP 1/1/2:800), which means that as soon as SAP 1/1/2:800 transitions to the forwarding state, the multicast stream is forwarded immediately to PE-30's multicast receiver without the need for an IGMP query from the IGMP querier and without an IGMP report for the group from PE-30.

Conclusion

Enabling MCS for IGMP snooping speeds up multicast convergence in the event of Layer 2 failures.

Synchronous Ethernet

In This Chapter

This section provides information about Synchronous Ethernet (SyncE).

Topics in this section include:

- [Applicability on page 78](#)
- [Summary on page 79](#)
- [Overview on page 80](#)
- [Configuration on page 87](#)
- [Conclusion on page 94](#)

Applicability

This example is applicable to all of the 7750 SR, 7710 SR and 7450 ESS series, except for the SR-1 and ESS-1, and was tested on release 8.0r7. There are no software pre-requisites for this configuration, however, the hardware requires the use of Synchronous Ethernet capable MDA-XP/CMA-XP or the IMMs.

In addition, Synchronous Ethernet is only supported on optical interfaces. It is not supported on 10/100/1000 base copper interfaces.

Summary

Synchronous Ethernet (SyncE) is the ability to provide PHY-level frequency distribution through an Ethernet port. It is one of the building blocks of Next Generation Networks (NGNs).

Overview

Synchronous Ethernet

Traditionally, Ethernet based networks employ the physical layer transmitter clock to be derived from an inexpensive $\pm 100\text{ppm}$ crystal oscillator and the receiver locks onto it. There is no need for long term frequency stability as the data is packetized and can be buffered. For the same reason there is no need for consistency between the frequencies of different links. However one could elect to derive the physical layer transmitter clock from a high quality frequency reference by replacing the crystal with a frequency source traceable to a primary reference clock. This would not affect the operation of any of the Ethernet layers, for which this change would be transparent. The receiver at the far end of the link would lock onto the physical layer clock of the received signal, and thus itself gain access to a highly accurate and stable frequency reference. Then, in a manner analogous to conventional hierarchical master-slave network synchronization, this receiver could lock the transmission clock of its other ports to this frequency reference and a fully time synchronous network could be established.

The advantage of using SyncE, as compared to methods relying on sending timing information in packets over an unlocked physical layer, is that SyncE is not influenced by impairments introduced by the higher levels of the networking technology (packet loss, packet delay variation). Hence, the frequency accuracy and stability may be expected to exceed those of networks with unsynchronized physical layers. In addition, SyncE was designed to integrate into any existing SONET/SDH synchronization distribution architecture to allow for the easy migration from the traditional to the new synchronous interfaces. SyncE includes the concept of a Hybrid Switch which supports the interworking of synchronization distribution through SONET/SDH and the SyncE interfaces at the same time.

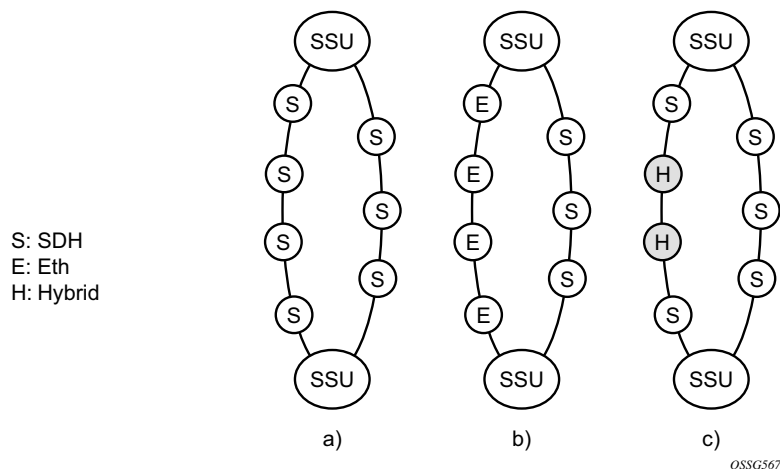


Figure 14: SyncE Hypothetical Reference Network Architecture

Many Tier 1 carriers are looking to migrate their synchronization infrastructure to a familiar and manageable model. In order to enable rapid migration of these networks, SyncE may be the easiest to deploy in order to ensure robust frequency synchronization.

Central Synchronization Sub-System

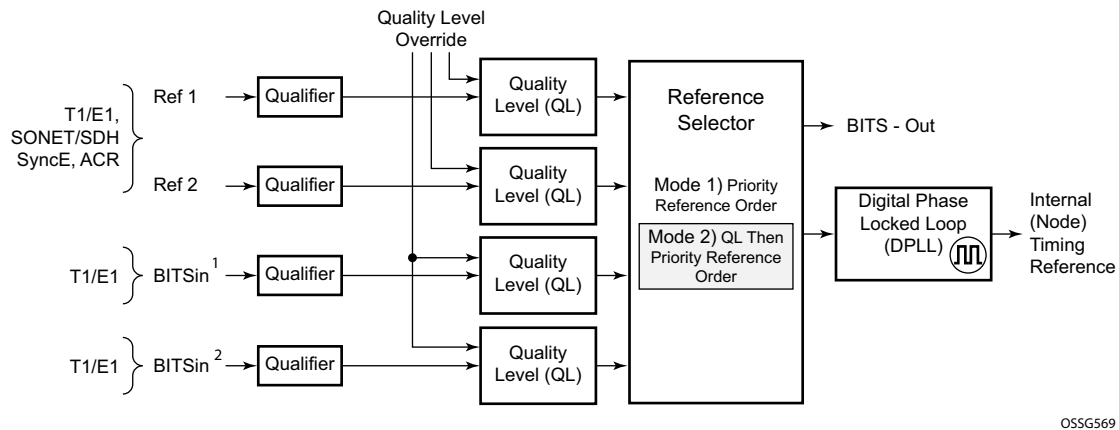


Figure 15: Packet Based Network Timing Infrastructure

Central Synchronization Sub-System

The timing subsystem for the SR/ESS platforms has a central clock located on the Control Processor Module (CPM). The timing subsystem performs many of the duties of the network element clock as defined by Telcordia (GR-1244) and ITU-T G.781.

The system can select from up to four timing inputs to train the local oscillator. The priority order of these references must be specified. This is a simple ordered list of inputs: {BITS [Building Integrated Timing Source], ref1, ref2}. The CPM clock output has the ability to drive the clocking for all line cards in the system. The SR/ESS supports selection of the node reference using Quality Level (QL) indications.

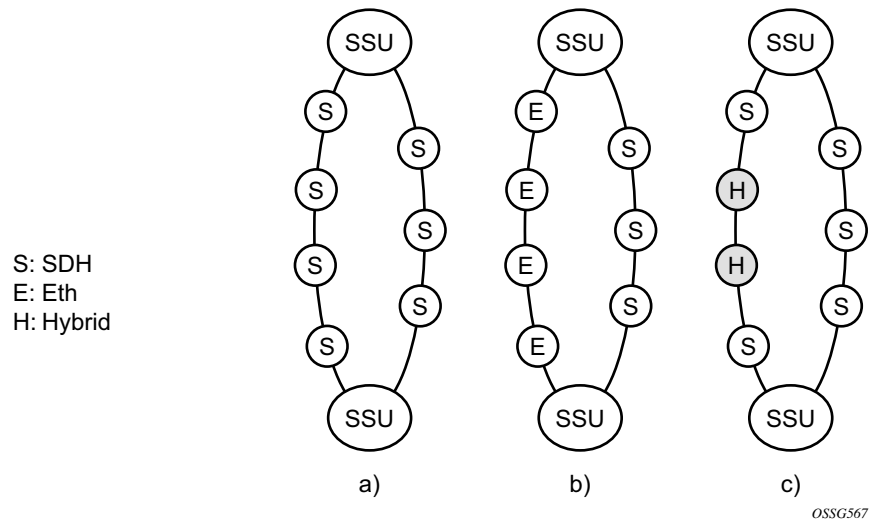


Figure 16: Current 7x50 Timing Sub-System Architecture

¹ BITSin port on Active CPM (7750 SR-7/12, 7450 ESS-7/12) or BITS_1 port on 7750 SR-c4.

² BITSin port on Standby CPM (7750 SR-7/12, 7450 ESS-7/12) or BITS_2 port on 7750 SR-c4.

The recovered clock is able to derive its timing from any of the following:

- OC3/STM1, OC12/STM4, OC48/STM16, OC192/STM64 ports
- T1/E1 CES channel (adaptive clocking)
- SyncE ports
- T1/E1 ports
- BITS port on a Channelized OC3/STM1 CES CMA (7710 SR-c4, 7710 SR-c12, and the 7750 SR-c12)
- BITS port on the CPM or CFM module

On 7750 SR-12 and 7750 SR-7 systems with redundant CPMs, the system has two BITS input ports (one per CPM). On the 7750 SR-c4 systems, there are two BITS input ports on the chassis front plate. These BITS input ports provide redundant synchronization inputs from an external BITS/SSU. Note the 7750 SR-c12 does not support BITS input port redundancy or BITS out.

All settings of the signal characteristics for the BITS input apply to both ports. When the active CPM considers the BITS input as a possible reference, it will consider first the BITS input port on the active CPM followed the BITS input port on the standby CPM in that relative priority order. This relative priority order is in addition to the user definable ref-order. For example, a ref-order of 'bits-ref1-ref2' would actually be BITS in (active CPM) followed by BITS in (standby CPM)

followed by ref1 followed by ref2. When ql-selection is enabled, then the QL of each BITS input port is viewed independently. The higher QL source is chosen.

On the 7750 SR-c4 platform CFM, there are two BITS input ports and two BITS output ports on this one module. These two ports are provided for BITS redundancy for the chassis. All settings of the signal characteristics for the BITS input apply to both ports. This includes the ql-override setting. When the CFM considers the BITS input as a possible reference, it will consider first the BITS input port “bits1” followed the BITS input port “bits2” in that relative priority order. This relative priority order is in addition to the user definable ref-order. For example, a ref-order of ‘bits-ref1-ref2’ would actually be “bits1” followed by “bits2” followed by ref1 followed by ref2. When ql-selection is enabled, then the QL of each BITS input port is viewed independently. The higher QL source is chosen.

The BITS output ports are provided to deliver a unfiltered recovered line clock from a SR/ESS port directly to a dedicated timing device in the facility (BITS or Standalone Synchronization Equipment (SASE) device). The signal selected will be one of ref1 or ref2. It cannot be the BITS input port signal nor can it be the output of the central clock.

When QL selection mode is disabled, then the reversion setting controls when the central clock can re-select a previously failed reference.

Table 1: Revertive, Non-Revertive Timing Reference Switching Operation

Status of Reference A	Status of Reference B	Active Reference Non-revertive Case	Active Reference Revertive Case
OK	OK	A	A
Failed	OK	B	B
OK	OK	B	A
OK	Failed	A	A
OK	OK	A	A
Failed	Failed	holdover	holdover
OK	Failed	A	A
Failed	Failed	holdover	holdover
Failed	OK	B	B
Failed	Failed	holdover	holdover
OK	OK	A or B	A

Synchronization Status Messages (SSM)

SSM provides a mechanism to allow the synchronization distribution network to both determine the quality level of the clock sourcing a given synchronisation trail and to allow a network element to select the best of multiple input synchronization trails. Synchronization Status messages have been defined for various transport protocols including SONET/SDH, T1/E1, and SyncE, for interaction with office clocks, such as BITS or SSUs (synchronisation supply unit) and embedded network element clocks.

SSM allows equipment to autonomously provision and reconfigure (by reference switching) their synchronization references, while helping to avoid the creation of timing loops. These messages are particularly useful to allow synchronization reconfigurations when timing is distributed in both directions around a ring.

In SyncE, the SSM is provided through the Ethernet Synchronization Messaging Channel (ESMC). This mechanism uses Ethernet OAM PDU to exchange the Quality Level values over the SyncE link.

SyncE Chains

Transmission of a reference clock through a chain of Ethernet equipment requires that all of the equipment support SyncE.

A single piece of equipment not capable of SyncE breaks the chain as shown in [Figure 17](#). Ethernet frames will still get through but downstream device will recognize that the signal is out of pull-in range and not use it for reference.

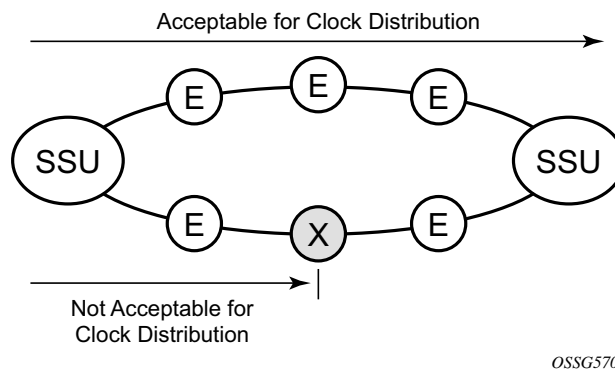


Figure 17: Network Considerations for Ethernet Timing Distribution

Configuration

Configuration 1

The following example shows the configuration options for SyncE when ql-selection mode is disabled. Generally, North American SONET networks do not use the automatic reference selection mechanisms. If SyncE is being added into such a network it would likely have ql-selection set to disabled.

```
A:PE-1>config# card 1 mda 1
A:PE-1>config>card>mda#
    access          + Configure access MDA parameters
    egress          + Configure egress MDA parameters
    [no] hi-bw-mcast-src - Enable/disable allocation of resources for high
                        bandwidth multicast streams
    ingress          + Configure ingress MDA parameters
    [no] mda-type    - Provisions/de-provisions an MDA to/from the device
                        configuration for the slot
    named-pool-mode + Enable/Disable named pool mode
    network          + Configure network MDA parameters
    [no] shutdown    - Administratively shut down an mda
    [no] sync-e      - Enable/Disable Synchronous Ethernet
A:PE-1>config>card>mda# sync-e
*A:PE-1>config>card>mda# info detail
-----
    mda-type m20-1gb-xp-sfp
    sync-e
    named-pool-mode
        ingress
            no named-pool-policy
        exit
        egress
            no named-pool-policy
        exit
    exit
    ingress
        no hsmda-pool-policy
        no scheduler-policy
    exit
    egress
        no hsmda-pool-policy
    exit
    network
        ingress
            pool default
    ...
```

Configuration

```
*A:PE-1>config>system# sync-if-timing
*A:PE-1>config>system>sync-if-timing#
    abort          - Discard the changes that have been made to sync
                    interface timing during a session
    begin          - Switch to edit mode for sync interface timing - use
                    commit to save or abort to discard the changes made in
                    a session
    bits           + Configure parameters for the Building Integrated Timing
                    Supply(BITS)
    commit         - Save the changes made to sync interface timing during a
                    session
[no] ql-selection  - Enable/disable reference selection based on
                    quality-level
[no] ref-order     - Priority order of timing references
    ref1          + Configure parameters for the first timing reference
    ref2          + Configure parameters for the second timing reference
[no] revert        - Revert/do not revert to a higher priority re-validated
                    reference source
*A:PE-1>config>system>sync-if-timing# begin
*A:PE-1>config>system>sync-if-timing# ref-order bits ref1
*A:PE-1>config>system>sync-if-timing# bits input no shutdown
*A:PE-1>config>system>sync-if-timing# bits interface-type ds1 esf
*A:PE-1>config>system>sync-if-timing# revert
*A:PE-1>config>system>sync-if-timing# ref1
*A:PE-1>config>system>sync-if-timing>ref1#
    [no] ql-override - Override the quality level of a timing reference
    [no] shutdown    - Administratively shutdown the timing reference
    [no] source-port - Configure the source port for the first timing reference
*A:PE-1>config>system>sync-if-timing>ref1# source-port 1/1/2
*A:PE-1>config>system>sync-if-timing>ref1# no shutdown
*A:PE-1>config>system>sync-if-timing>ref1# exit
*A:PE-1>config>system>sync-if-timing# commit
*A:PE-1>config>system>sync-if-timing# info detail
-----
    no ql-selection
    ref-order bits ref1 ref2
    ref1
        source-port 1/1/2
        no shutdown
        no ql-override
    exit
    ref2
        shutdown
        no source-port
        no ql-override
    exit
    bits
        interface-type ds1 esf
        no ql-override
        input
            no shutdown
        exit
        output
            shutdown
            line-length 110
        exit
    exit
    revert
```

The following output displays the associated show information.

```
*A:PE-1>show>system# sync-if-timing
=====
System Interface Timing Operational Info
=====
System Status CPM A           : Master Locked
  Reference Input Mode         : Revertive
  Quality Level Selection      : Disabled
  Reference Selected           : ref1
  System Quality Level         : unknown
  Current Frequency Offset (ppm) : -5

Reference Order                : bits ref1 ref2

Reference Mate CPM
  Qualified For Use             : No
  Not Qualified Due To         :      LOS
  Selected For Use             : No
  Not Selected Due To         :      not qualified

Reference Input 1
  Admin Status                 : up
  Rx Quality Level             : unknown
  Quality Level Override       : none
  Qualified For Use            : Yes
  Selected For Use             : Yes
  Source Port                  : 1/1/2

Reference Input 2
  Admin Status                 : down
  Rx Quality Level             : unknown
  Quality Level Override       : none
  Qualified For Use            : No
  Not Qualified Due To         :      disabled
  Selected For Use             : No
  Not Selected Due To         :      disabled
  Source Port                  : None

Reference BITS A
  Input Admin Status           : up
  Rx Quality Level             : failed
  Quality Level Override       : none
  Qualified For Use            : No
  Not Qualified Due To         :      LOS
  Selected For Use             : No
  Not Selected Due To         :      not qualified
  Interface Type               : DS1
  Framing                     : ESF
  Line Coding                   : B8ZS
=====
*A:PE-1>show>system#
```

Configuration 2

The following example shows the configuration options for SyncE when ql-selection mode is enabled.

This is the normal case for European SDH networks.

```
A:PE-1>config# card 1 mda 1
A:PE-1>config>card>mda#
    access          + Configure access MDA parameters
    egress          + Configure egress MDA parameters
[no] hi-bw-mcast-src - Enable/disable allocation of resources for high
                    bandwidth multicast streams
    ingress         + Configure ingress MDA parameters
[no] mda-type       - Provisions/de-provisions an MDA to/from the device
                    configuration for the slot
    named-pool-mode + Enable/Disable named pool mode
    network         + Configure network MDA parameters
[no] shutdown      - Administratively shut down an mda
[no] sync-e        - Enable/Disable Synchronous Ethernet
A:PE-1>config>card>mda# sync-e
A:PE-1>config>card>mda# info detail
-----
mda-type m20-1gb-xp-sfp
sync-e
named-pool-mode
    ingress
        no named-pool-policy
    exit
    egress
        no named-pool-policy
    exit
exit
ingress
    no hsmda-pool-policy
    no scheduler-policy
exit
egress
    no hsmda-pool-policy
exit
network
    ingress
        pool default
...

A:PE-1>config# port 1/1/2 ethernet ssm
A:PE-1>config>port>ethernet>ssm#
A:PE-1>config>port>ethernet>ssm#
[no] code-type      - Set the SSM channel to either use sonet or sdh
[no] shutdown       - Enable/Disable SSM
[no] tx-dus         - Enable/disable always transmit 0xF (dus/dnu) in SSM messaging chan-
nel
A:PE-1>config>port>ethernet>ssm# code-type sdh
*A:PE-1>config>port>ethernet>ssm# no shutdown
*A:PE-1>config>port>ethernet>ssm# info detail
```

```

-----
code-type sdh
no tx-dus
no shutdown
-----

*A:PE-1>config>port>ethernet>ssm#

*A:PE-1>config>system# sync-if-timing
*A:PE-1>config>system>sync-if-timing#
    abort          - Discard the changes that have been made to sync
                    interface timing during a session
    begin          - Switch to edit mode for sync interface timing - use
                    commit to save or abort to discard the changes made in
                    a session
    bits           + Configure parameters for the Building Integrated Timing
                    Supply (BITS)
    commit         - Save the changes made to sync interface timing during a
                    session
    [no] ql-selection - Enable/disable reference selection based on
                    quality-level
    [no] ref-order  - Priority order of timing references
    ref1           + Configure parameters for the first timing reference
    ref2           + Configure parameters for the second timing reference
    [no] revert     - Revert/do not revert to a higher priority re-validated
                    reference source
*A:PE-1>config>system>sync-if-timing# begin
*A:PE-1>config>system>sync-if-timing# ref-order bits ref1
*A:PE-1>config>system>sync-if-timing# ql-selection
*A:PE-1>config>system>sync-if-timing# bits input no shutdown
*A:PE-1>config>system>sync-if-timing# bits interface-type ds1 esf
*A:PE-1>config>system>sync-if-timing# bits ql-override prc
*A:PE-1>config>system>sync-if-timing# revert
*A:PE-1>config>system>sync-if-timing# ref1
*A:PE-1>config>system>sync-if-timing>ref1#
    [no] ql-override - Override the quality level of a timing reference
    [no] shutdown    - Administratively shutdown the timing reference
    [no] source-port - Configure the source port for the first timing reference
*A:PE-1>config>system>sync-if-timing>ref1# source-port 1/1/2
*A:PE-1>config>system>sync-if-timing>ref1# no shutdown
*A:PE-1>config>system>sync-if-timing>ref1# exit
*A:PE-1>config>system>sync-if-timing# commit
*A:PE-1>config>system>sync-if-timing# info detail
-----
ql-selection
ref-order bits ref1 ref2
ref1
    source-port 1/1/2
    no shutdown
    no ql-override
exit
ref2
    shutdown
    no source-port
    no ql-override
exit
bits
    interface-type ds1 esf

```

Configuration

```
        ql-override prc
        input
            no shutdown
        exit
        output
            shutdown
            line-length 110
        exit
    exit
revert
```

The following output displays the associated show information.

```
*A:PE-1>show>system# sync-if-timing
=====
System Interface Timing Operational Info
=====
System Status CPM A           : Master Locked
  Reference Input Mode        : Revertive
  Quality Level Selection     : Enabled
  Reference Selected          : ref1
  System Quality Level        : prc
  Current Frequency Offset (ppm) : -5

Reference Order                : bits ref1 ref2

Reference Mate CPM
  Qualified For Use            : No
    Not Qualified Due To      :      LOS
  Selected For Use            : No
    Not Selected Due To      :      not qualified

Reference Input 1
  Admin Status                : up
  Rx Quality Level            : prc
  Quality Level Override      : none
  Qualified For Use           : Yes
  Selected For Use            : Yes
  Source Port                 : 1/1/2

Reference Input 2
  Admin Status                : down
  Rx Quality Level            : unknown
  Quality Level Override      : none
  Qualified For Use           : No
    Not Qualified Due To      :      disabled
  Selected For Use            : No
    Not Selected Due To      :      disabled
  Source Port                 : None

Reference BITS A
  Input Admin Status          : up
  Rx Quality Level            : failed
  Quality Level Override      : none
  Qualified For Use           : No
    Not Qualified Due To      :      LOS
  Selected For Use            : No
    Not Selected Due To      :      not qualified
```



```
Interface Type      : DS1
Framing             : ESF
Line Coding         : B8ZS
=====
*A:PE-1>show>system#
```

Conclusion

With the world rapidly transitioning to IP/MPLS-based NGNs with Ethernet as the transport medium of choice, there is an increasing need to enhance services and capabilities while still leveraging existing infrastructure, thereby easing the transition while continuing to increase revenue and reduce the Total Cost of Ownership (TCO). In areas such as mobile backhaul, TDM CES etc., these requirements create a need for SONET/SDH-like frequency synchronization capability in the inherently asynchronous Ethernet network.

SyncE, natively supported on the Alcatel-Lucent 7750 SR and 7450 ESS service routers, is an ITU-T standardized PHY-level way of transmitting frequency synchronization across Ethernet packet networks that fulfills that need in a reliable, secure, scalable, efficient, and cost-effective manner. It allows Service Providers to keep existing revenue streams alive and create new ones while simplifying the network design and reducing TCO.

System Management

In This Section

This section provides configuration information for the following topics:

- [Distributed CPU Protection on page 97](#)
- [Event Handling System on page 123](#)

Distributed CPU Protection

In This Chapter

This section describes Distributed CPU Protection (DCP) configurations.

Topics in this section include:

- [Applicability on page 98](#)
- [Overview on page 99](#)
- [Configuration on page 100](#)
- [Conclusion on page 122](#)

Applicability

This Distributed CPU Protection (DCP) configuration example was created using the 7750 SR-c12 platform but is equally applicable to the following platforms: 7750 SR-7/12, 7450 ESS-6/7/12, 7750 SR-c4/c12 and 7950 XRS. DCP is not supported on the 7750 SR-1, 7450 ESS-1 or 7710 SR platforms.

DCP operates on the line cards and requires line cards with the FP2 or greater hardware (for example, IOM3-XP, IMMs and C-XMAs).

The configuration was tested on release 11.0R1.

Overview

SR OS provides several rate limiting mechanisms to protect the CPM/CFM processing resources of the router:

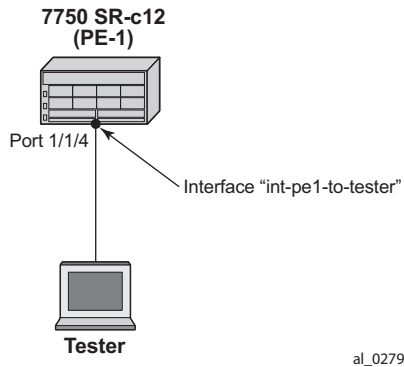
- CPU Protection: A centralized rate limiting function that operates on the CPM to limit traffic destined to the CPUs.
- Distributed CPU Protection: A control traffic rate limiting protection mechanism for the CPM/CFM that operates on the line cards (hence ‘distributed’). CPU protection protects the CPU of the node that it is configured on from a DOS attack by limiting the amount of traffic coming in from one of its ports and destined to the CPM (to be processed by its CPU) using a combination of the configurable limits.

The goal of this example is to familiarize the reader with the configuration and use of Distributed CPU Protection. A simple and controlled setup is used to illustrate how the protection behaves and how to use the tools provided for the feature.

External testing equipment (“tester”) is used to send control traffic of various protocols at various rates to the router in order to exercise DCP. Log events and show routines are examined to explain the indications that the router provides to an operator.

Configuration

The test topology is shown in [Figure 18](#). A Gigabit Ethernet link is used between the Tester and the router.



al_0279

Figure 18: Test Topology

Step 1. The basic configuration of the mda, port, interface and a security event log on the router is shown below.

```
*A:PE-1# configure card 1 mda 1
*A:PE-1>config>card>mda# info
-----
mda-type m5-1gb-sfp-b
no shutdown
-----
*A:PE-1>config>card>mda# exit all
*A:PE-1# configure port 1/1/4
*A:PE-1>config>port# info
-----
ethernet
exit
no shutdown
-----
*A:PE-1>config>port# exit all
*A:PE-1# configure router interface "int-pe1-to-tester"
*A:PE-1>config>router>if# info
-----
address 192.168.10.1/24
port 1/1/4
no shutdown
-----
*A:PE-1>config>router>if# exit all
*A:PE-1# configure log log-id 15
*A:PE-1>config>log>log-id# info
-----
from security
to memory 1024
-----
```


This example was developed on a 7750 SR-c12 platform but it is equally applicable to other platforms such as the 7750 SR-7/12. If other platforms, such as the 7750 SR-7/12 that support centralized CPU Protection, are used to explore Distributed CPU Protection then the centralized CPU Protection should be disabled (for the purposes of this example) so that it does not interfere with reproducing the same results as described below. In a normal production network CPU Protection and DCP are complimentary and can be used together. To disable centralized CPU Protection for the purposes of reproducing the results below please ensure that:

- **protocol-protection** is disabled.
- All rates in all policies (including any default policies) are configure to **max**.

Step 2. In order to activate DCP a policy is created and assigned to the interface.

The first policy that is used in this example is used to simply count protocol packets to see that they are indeed flowing from the tester to the router and being extracted and indentified.

The *dcp-policy-count* policy is configured as follows:

```
*A:PE-1# configure system security dist-cpu-protection
*A:PE-1>config>sys>security>dist-cpu-protection# info
-----
policy "dcp-policy-count" create
  description "Static policers with rate 0 for counting packets"
  static-policer "sp-arp" create
    rate packets 0 within 1
  exit
  static-policer "sp-icmp" create
    rate packets 0 within 1
  exit
  static-policer "sp-igmp" create
    rate packets 0 within 1
  exit
  protocol arp create
    enforcement static "sp-arp"
  exit
  protocol icmp create
    enforcement static "sp-icmp"
  exit
  protocol igmp create
    enforcement static "sp-igmp"
  exit
exit
```

For the *dcp-policy-count* policy configuration:

- The policy contains three static policers: *sp-arp*, *sp-icmp* and *sp-igmp*. These policers are then used by the three configured protocols that are part of the policy: *arp*, *icmp* and *igmp*.
- The list of protocols that are applicable to DCP are as follows: *arp*, *dhcp*, *http-redirect*, *icmp*, *igmp*, *mld*, *ndis*, *pppoe-pppoa*, *all-unspecified*, *mpls-ttl*, *bfd-cpm*, *bgp*, *eth-cfm*, *isis*, *ldp*, *ospf*, *pim* and *rsvp*. The *all-unspecified* protocol is a special “catch-all”. Please see the 7750 SR OS System Management Guide for more details.
- This policy instantiates three permanent (static) policers for every object (for example, interface) that the policy is associated with.
- The three protocols each reference their own static policer so each protocol will be independently rate limited. A single static policer can also be used to rate limit multiple protocols but that capability is not used in this example.
- The rate is set to 0 which means all packets will be considered as non-conformant to the policer. This configuration is used to provide counters of protocol packets. The DCP counters provide the count of packets exceeding the policing parameters since the given policer was previously declared as conformant or newly instantiated. A rate of zero ensures that the policer will never be declared as conformant and hence will never reset the counters.
- The exceed-action is not configured and takes the default value of *none*. The *log-events* parameter is not configured and is enabled by default. That means the policer will notify the operator when the first packet arrives but will not discard or mark any packets.

Step 3. Assign the *dcp-policy-count* to the interface:

```
*A:PE-1# configure router interface "int-pe1-to-tester"
*A:PE-1>config>router>if# dist-cpu-protection "dcp-policy-count"
```

Step 4. Examine some log and status on the router to get a baseline (no traffic is flowing from the tester to the router at this point). Notice that the cpu utilization is fairly low with an overall Idle of 96% and no task groups at more than 5% capacity usage. Future example output from this show routine will be snipped to only show relevant and interesting lines.

```
*A:PE-1# show system cpu
=====
CPU Utilization (Sample period: 1 second)
=====
```

Name	CPU Time (uSec)	CPU Usage	Capacity Usage
BFD	0	0.00%	0.00%
BGP	28,779	0.32%	0.47%
BGP PE-CE	0	0.00%	0.00%
CFLOWD	7,384	0.08%	0.38%
Cards & Ports	65,941	0.73%	5.35%
DHCP Server	55	~0.00%	~0.00%
ICC	1,195	0.01%	0.06%
IGMP/MLD	1,883	0.02%	0.12%

IMSI Db Appl	120	~0.00%	~0.00%
IOM	132,522	1.47%	3.11%
IP Stack	7,666	0.08%	0.39%
IS-IS	1,415	0.01%	0.07%
ISA	11,988	0.13%	0.43%
LDP	496	~0.00%	0.04%
Logging	185	~0.00%	0.01%
MBUF	0	0.00%	0.00%
MPLS/RSVP	6,219	0.06%	0.48%
MSCP	0	0.00%	0.00%
MSDP	0	0.00%	0.00%
Management	4,077	0.04%	0.13%
OAM	10,311	0.11%	0.44%
OSPF	661	~0.00%	0.05%
PIM	0	0.00%	0.00%
RIP	0	0.00%	0.00%
RTM/Policies	0	0.00%	0.00%
Redundancy	7,641	0.08%	0.51%
SNMP Daemon	0	0.00%	0.00%
Services	3,965	0.04%	0.09%
Stats	0	0.00%	0.00%
Subscriber Mgmt	7,437	0.08%	0.44%
System	57,081	0.63%	3.49%
Traffic Eng	0	0.00%	0.00%
VRRP	1,918	0.02%	0.09%
WEB Redirect	77	~0.00%	~0.00%

Total	8,965,427	100.00%	
Idle	8,605,657	95.98%	
Usage	359,770	4.01%	
Busiest Core Utilization	134,481	13.49%	
=====			

The DCP feature is reporting no violations for interfaces on card 1.

```
*A:PE-1# tools dump security dist-cpu-protection violators enforcement interface card 1
=====
Distributed Cpu Protection Current Interface Enforcer Policer Violators
=====
Interface                               Policer/Protocol                        Hld Rem
-----
Violators on Slot-1 Fp-1
-----
[S]-Static [D]-Dynamic [M]-Monitor
=====
```

There are no security log events.

```
*A:PE-1# show log log-id 15
=====
Event Log 15
=====
Description : (Not Specified)
Memory Log contents [size=1024 next event=1 (not wrapped)]
```

The detailed DCP status for the interface shows all three policers are currently in the conform state.

```
*A:PE-1# show router interface "int-pel-to-tester" dist-cpu-protection
=====
Interface "int-pel-to-tester" (Router: Base)
=====
Distributed CPU Protection Policy : dcp-policy-count
-----
Statistics/Policer-State Information
=====
-----
Static Policer
-----
Policer-Name      : sp-arp
Card/FP           : 1/1
Protocols Mapped  : arp
Exceed-Count      : 0
Detec. Time Remain : 0 seconds
Policer-State     : Conform
Hold-Down Remain. : none

Policer-Name      : sp-icmp
Card/FP           : 1/1
Protocols Mapped  : icmp
Exceed-Count      : 0
Detec. Time Remain : 0 seconds
Policer-State     : Conform
Hold-Down Remain. : none

Policer-Name      : sp-igmp
Card/FP           : 1/1
Protocols Mapped  : igmp
Exceed-Count      : 0
Detec. Time Remain : 0 seconds
Policer-State     : Conform
Hold-Down Remain. : none
-----
-----
Local-Monitoring Policer
-----
No entries found
-----
-----
Dynamic-Policer (Protocol)
-----
No entries found
-----
=====
```

Step 5. Configure the tester to send ARP, ICMP and IGMP traffic to the router using the following rates:

- ARP: 2 packets per second (pps)
- ICMP: 4 pps
- IGMP: 8 pps

Here are some tips for how to configure the tester to send protocol packets that will be recognized by the router:

- ARP:
 - Set the MAC destination address to FF-FF-FF-FF-FF-FF
 - Use an ARP Request format
- ICMP:
 - Use an icmp type of 8 (echo request, such as **ping**).
 - Set the MAC destination address equal to the MAC address of the receiving port. The MAC address of port 1/1/4 can be seen in the output of show port 1/1/4 as the Configured Address.
 - Set the IP destination address to 192.168.10.1 and the IP source address to 192.168.10.2.
- IGMP:
 - Set the MAC destination address equal to the MAC address of the receiving port. The MAC address of port 1/1/4 can be seen in the output of show port 1/1/4 as the Configured Address.
 - Set the IP destination address to 224.0.0.2 and the IP source address to 0.0.0.0.
 - Set the IGMP version to 2, make the IGMP message type a Membership Query to Group 0.

Also ensure that the tester interleaves the three streams of protocol packets such that it schedules them independently in an interleaved fashion, not serially.

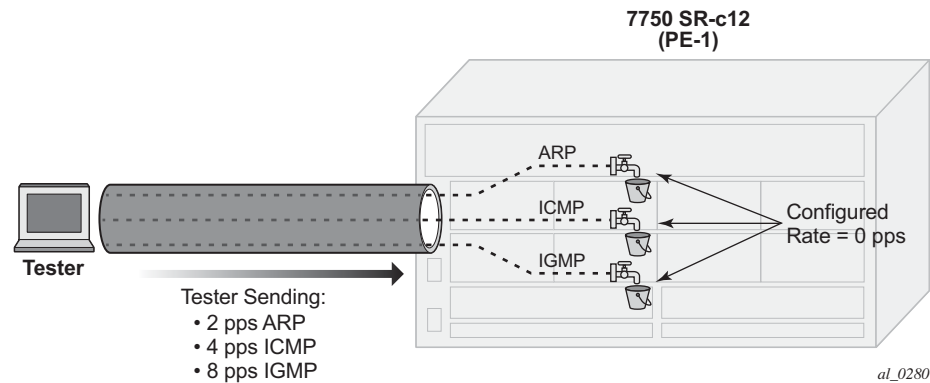


Figure 19: Count Traffic with DCP Policy Count

Step 6. Notice that DCP now reports some violations of the policy against the interface.

```
*A:PE-1# tools dump security dist-cpu-protection violators enforcement interface card 1
=====
Distributed Cpu Protection Current Interface Enforcer Policer Violators
=====
Interface                               Policer/Protocol                         Hld Rem
-----
Violators on Slot-1 Fp-1
-----
int-pe1-to-tester                       sp-arp                                  [S] none
int-pe1-to-tester                       sp-icmp                                 [S] none
int-pe1-to-tester                       sp-igmp                                 [S] none
-----
[S]-Static [D]-Dynamic [M]-Monitor
=====
```

After a few seconds the DCP exceed-count values can be seen incrementing.

Note the following details:

- Exceed-Count is non-zero. This will continue incrementing and will never reset since the rate configured in the DCP policy is zero.
- The Policer-State is Exceed. The policers have detected that the protocol is non-conformant to the configured rate.
- Detec. Time Remain stays at 29 seconds. This countdown timer is automatically reset to 30 seconds every time a policer is detected as non-conformant (which will be continually when the rate is set to 0 and packets of that protocol are being received).

```

*A:PE-1# show router interface "int-pel-to-tester" dist-cpu-protection
=====
Interface "int-pel-to-tester" (Router: Base)
=====
Distributed CPU Protection Policy : dcp-policy-count
-----
Statistics/Policer-State Information
=====
-----
Static Policer
-----
Policer-Name      : sp-arp
Card/FP           : 1/1
Protocols Mapped  : arp
Exceed-Count      : 72
Detec. Time Remain : 29 seconds
Policer-State     : Exceed
Hold-Down Remain. : none

Policer-Name      : sp-icmp
Card/FP           : 1/1
Protocols Mapped  : icmp
Exceed-Count      : 144
Detec. Time Remain : 29 seconds
Policer-State     : Exceed
Hold-Down Remain. : none

Policer-Name      : sp-igmp
Card/FP           : 1/1
Protocols Mapped  : igmp
Exceed-Count      : 290
Detec. Time Remain : 29 seconds
Policer-State     : Exceed
Hold-Down Remain. : none
-----
[snip]

```

Step 7. Keep the tester running.

Now a DCP policy that enforces protocol rates using static policers will be applied to the interface. First, the policy is created:

```

*A:PE-1# configure system security dist-cpu-protection
*A:PE-1>config>sys>security>dist-cpu-protection# policy "dcp-static-policy-1" create
description "Static policers for arp, icmp and igmp"
static-policer "sp-arp" create
    rate packets 10 within 1
    exceed-action discard
exit
static-policer "sp-icmp" create
    rate packets 20 within 1
    exceed-action discard
exit
static-policer "sp-igmp" create
    rate packets 10 within 1
    exceed-action discard
exit
protocol arp create
    enforcement static "sp-arp"
exit
protocol icmp create
    enforcement static "sp-icmp"
exit
protocol igmp create

```

Configuration

```
        enforcement static "sp-igmp"
    exit
exit
```

For the dcp-static-policy-1 policy configuration, note that a few parameters are different than in the previously created dcp-policy-count policy:

- The rates are set to low (but non-zero) values.
- The exceed-action is configured such that packets are dropped once the rate is exceeded.

Now assign the policy to the test interface:

```
*A:PE-1# configure router interface "int-pe1-to-tester"
*A:PE-1>config>router>if# dist-cpu-protection "dcp-static-policy-1"
*A:PE-1>config>router>if# exit all
*A:PE-1# show system security dist-cpu-protection policy "dcp-static-policy-1" association
=====
Distributed CPU Protection Policy
=====
Policy Name : dcp-static-policy-1
Description : Static policers for arp, icmp and igmp

-----
Associations
-----

SAP associations
-----
None

Managed SAP associations
-----
None

Interface associations
-----
Router-Name : Base
    int-pe1-to-tester
-----
Number of interfaces : 1
=====
```


Step 8. Increase the rate of IGMP packets that the tester is sending to 1000pps (keep ARP and ICMP at 2pps and 4pps).

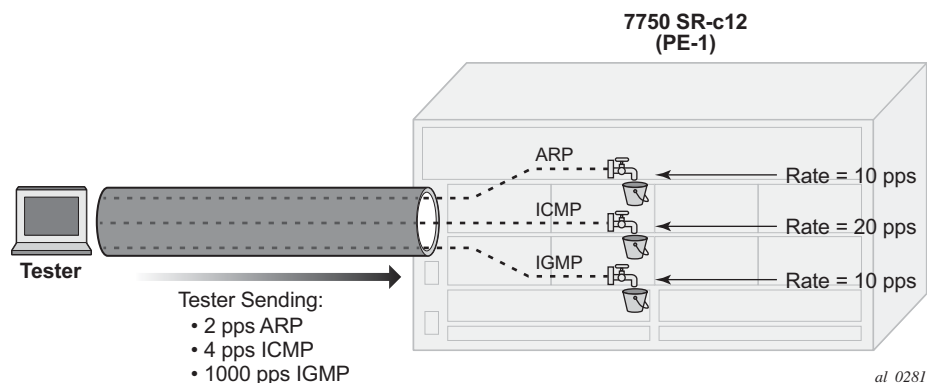


Figure 20: Limit Traffic with dcp-static-policy-1

Step 9. Notice that the system has identified a violation of the DCP rates for the igmp policer.

```
*A:PE-1# tools dump security dist-cpu-protection violators enforcement interface card 1
=====
Distributed Cpu Protection Current Interface Enforcer Policer Violators
=====
Interface                               Policer/Protocol                        Hld Rem
-----
Violators on Slot-1 Fp-1
-----
int-pe1-to-tester                       sp-igmp                                [S] none
-----
[S]-Static [D]-Dynamic [M]-Monitor
=====
```

After a few minutes the violation will be indicated as a log event. This delay is due to the design of DCP. In order to support large scale operation of DCP, and also to avoid overload conditions, a polling process is used to monitor state changes in the policers and to gather violations. This means there can be a delay between when an event occurs in the data plane and when the relevant state change or event notification occurs towards an operator, but in the meantime the policers are still operating and protecting the control plane.

```
*A:PE-1# show log log-id 15
=====
Event Log 15
=====
Description : (Not Specified)
```

Configuration

```
Memory Log contents [size=1024 next event=11 (not wrapped)]

10 2013/04/18 17:31:54.58 EDT WARNING: SECURITY #2066 Base DCPUPROT
"Non conformant network if "int-pel-to-tester" on fp 1/1 detected at 04/18/2013 17:31:33.
Policy "dcp-static-policy-1". Policer="sp-igmp"(static). Excd count=135"
... [snip] ...
```

The status of DCP on the interface also shows the igmp policer as being in an Exceed state:

```
*A:PE-1# show router interface "int-pel-to-tester" dist-cpu-protection
=====
Interface "int-pel-to-tester" (Router: Base)
=====
Distributed CPU Protection Policy : dcp-static-policy-1
-----
Statistics/Policer-State Information
=====
-----
Static Policer
-----
Policer-Name      : sp-arp
Card/FP           : 1/1
Protocols Mapped  : arp
Exceed-Count      : 0
Detec. Time Remain : 0 seconds
Policer-State     : Conform
Hold-Down Remain. : none

Policer-Name      : sp-icmp
Card/FP           : 1/1
Protocols Mapped  : icmp
Exceed-Count      : 0
Detec. Time Remain : 0 seconds
Policer-State     : Conform
Hold-Down Remain. : none

Policer-Name      : sp-igmp
Card/FP           : 1/1
Protocols Mapped  : igmp
Exceed-Count      : 19031
Detec. Time Remain : 29 seconds
Policer-State     : Exceed
Hold-Down Remain. : none
-----
...[snip]...
```

The CPU utilization of the IGMP task group is not impacted since DCP is discarding packets that are non-conformant to the configure rate.

```
*A:PE-1# show system cpu
=====
CPU Utilization (Sample period: 1 second)
=====
Name                                CPU Time      CPU Usage      Capacity
                                (uSec)                               Usage
-----
BFD                                0             0.00%          0.00%
...[snip]...
IGMP/MLD                          1,883         0.02%          0.12%
IMSI Db Appl                       120           ~0.00%         ~0.00%
IOM                             132,522       1.47%          3.11%
IP Stack                          7,666         0.08%          0.39%
```

```

IS-IS                1,415                0.01%                0.07%
ISA                  11,988                0.13%                0.43%
LDP                   496                ~0.00%                0.04%
...[snip]...
WEB Redirect          77                ~0.00%                ~0.00%
-----
Total                8,965,427            100.00%
  Idle              8,605,657            95.98%
  Usage              359,770             4.01%
Busiest Core Utilization 134,481            13.49%
=====

```

Step 10. Remove the DCP policy from the interface and see the CPU utilization goes up for the IGMP task group.

```

*A:PE-1# configure router interface "int-pel-to-tester"
*A:PE-1>config>router>if# no dist-cpu-protection
*A:PE-1>config>router>if# /show system cpu
=====
CPU Utilization (Sample period: 1 second)
=====
Name                        CPU Time      CPU Usage      Capacity
                        (uSec)                Usage
-----
BFD                          0              0.00%          0.00%
...[snip]...
IGMP/MLD                    82,142          0.91%          8.14%
IMSI Db Appl                  98             ~0.00%          ~0.00%
IOM                       129,851          1.45%          3.15%
IP Stack                    196,549          2.19%          19.35%
IS-IS                       1,484            0.01%          0.07%
ISA                       11,765            0.13%          0.42%
LDP                         449             ~0.00%          0.04%
...[snip]...
WEB Redirect                 102             ~0.00%          0.01%
-----
Total                      8,948,806      100.00%
  Idle                    8,259,903      92.30%
  Usage                    688,903        7.69%
Busiest Core Utilization 210,435        21.16%
=====

```

Step 11. Increase the rate of IGMP traffic from the tester to 5000 pps. See the CPU utilization increase further.

```

*A:PE-1# show system cpu
=====
CPU Utilization (Sample period: 1 second)
=====
Name                        CPU Time      CPU Usage      Capacity
                        (uSec)                Usage
-----
BFD                          0              0.00%          0.00%
...[snip]...
IGMP/MLD                    417,124          4.65%          41.78%
IMSI Db Appl                  82             ~0.00%          ~0.00%
IOM                       133,029          1.48%          2.92%

```

Configuration

```
IP Stack                935,491          10.43%          93.45%
IS-IS                   1,343           0.01%           0.06%
ISA                    12,350           0.13%           0.45%
LDP                     394            ~0.00%           0.03%
...[snip]...
WEB Redirect           116            ~0.00%           0.01%
-----
Total                  8,966,128        100.00%
  Idle                 6,972,962        77.77%
  Usage                1,993,166        22.22%
Busiest Core Utilization 484,748          48.65%
=====
```

Step 12. Reinstall the DCP policy to the interface and see the CPU utilization drop.

```
*A:PE-1# configure router interface "int-pe1-to-tester"
*A:PE-1>config>router>if# dist-cpu-protection "dcp-static-policy-1"
*A:PE-1>config>router>if# exit all
*A:PE-1# show system cpu
=====
CPU Utilization (Sample period: 1 second)
=====
Name                      CPU Time      CPU Usage      Capacity
                          (uSec)                          Usage
-----
BFD                        0             0.00%          0.00%
...[snip]...
IGMP/MLD                   2,058         0.02%          0.10%
IMSI Db Appl                48            ~0.00%         ~0.00%
IOM                       135,148        1.50%          3.04%
IP Stack                   7,851         0.08%          0.47%
IS-IS                     1,398         0.01%          0.07%
ISA                       11,730        0.13%          0.43%
LDP                        299           ~0.00%          0.02%
...[snip]...
WEB Redirect               71            ~0.00%         ~0.00%
-----
Total                     8,975,262      100.00%
  Idle                     8,611,593      95.94%
  Usage                    363,669         4.05%
Busiest Core Utilization 136,669         13.70%
=====
```

Step 13. Stop the tester from sending packets, wait a few minutes and then note the status of the system.

There are no longer any violations of any enforcement policers on any interfaces on card 1.

```
*A:PE-1# tools dump security dist-cpu-protection violators enforcement interface card 1
=====
Distributed Cpu Protection Current Interface Enforcer Policer Violators
=====
Interface                      Policers/Protocol                      Hld Rem
-----
Violators on Slot-1 Fp-1
```

```
-----
[S]-Static [D]-Dynamic [M]-Monitor
-----
=====
```

The IGMP policer is indicated as conformant in the log events.

```
*A:PE-1# show log log-id 15
=====
Event Log 15
=====
Description : (Not Specified)
Memory Log contents [size=1024 next event=7 (not wrapped)]

...[snip]...

12 2013/04/18 17:42:12.43 EDT WARNING: SECURITY #2072 Base DCPUPROT
"Network_if "int-pel-to-tester" on fp 1/1 newly conformant at 04/18/2013 17:41:57:27. Pol-
icy "dcp-static-policy-1". Policer="sp-igmp"(static). Excd count=316418"

...[snip]...
```

The interface DCP details show all policers as conformant.

```
*A:PE-1# show router interface "int-pel-to-tester" dist-cpu-protection
=====
Interface "int-pel-to-tester" (Router: Base)
=====
Distributed CPU Protection Policy : dcp-static-policy-1
-----
Statistics/Policer-State Information
=====
-----
Static Policer
-----
Policer-Name      : sp-arp
Card/FP           : 1/1
Protocols Mapped  : arp
Exceed-Count      : 0
Detec. Time Remain : 0 seconds
Policer-State     : Conform
Hold-Down Remain. : none

Policer-Name      : sp-icmp
Card/FP           : 1/1
Protocols Mapped  : icmp
Exceed-Count      : 0
Detec. Time Remain : 0 seconds
Policer-State     : Conform
Hold-Down Remain. : none

Policer-Name      : sp-igmp
Card/FP           : 1/1
Protocols Mapped  : igmp
Exceed-Count      : 0
Detec. Time Remain : 0 seconds
Policer-State     : Conform
Hold-Down Remain. : none
-----
...[snip]...
```

An optional hold-down can be used in the configuration of the exceed-action of the policers in order to apply the exceed-action for a defined period (even if the policer goes conformant again during that period). The hold-down could be used, for example, to discard all packets associated with a policer for one hour after a violation is detected. An “indefinite” period is also supported which enforces discard or marking until the operator clears the policer with the **tools perform security dist-cpu-protection release-hold-down** command.

Step 14. The next scenario explored in this example is the use of DCP dynamic enforcement.

In order to use dynamic enforcement policers, a number of dynamic policers must be allocated to the DCP pool for the particular card being used.

```
*A:PE-1# configure card 1 fp dist-cpu-protection
*A:PE-1>config>card>fp>d-cpu-prot# info
-----
dynamic-enforcement-policer-pool 1000
-----
```

The number allocated should be greater than the maximum number of dynamic policers expected to be in use on the card at one time. A conservative (large) number could be selected at first, and then the following show command can give data to help tune the pool to a smaller size over time:

```
*A:PE-1# show card 1 fp 1 dist-cpu-protection
=====
Card : 1 Forwarding Plane (FP) : 1
=====
Dynamic Enforcement Policer Pool : 1000
-----
Statistics Information
-----
Dynamic-Policers Currently In Use      : 0
Hi-WaterMark Hit Count                 : 0
Hi-WaterMark Hit Time                  : 04/20/2013 08:16:24 UTC
Dynamic-Policers Allocation Fail Count : 0
=====
```

If the dynamic-enforcement-policer-pool is too small then when a local-monitoring-policer detects violating traffic, the dynamic enforcement policers will not be able to be instantiated. A log event will warn the operator when the pool is nearly exhausted.

A sample dynamic enforcement policy is created as follows:

```
*A:PE-1# configure system security dist-cpu-protection
*A:PE-1>config>sys>security>dist-cpu-protection# policy "dcp-dynamic-policy-1" create
description "Dynamic policing policy"
local-monitoring-policer "local-mon" create
description "Monitor for arp, icmp, igmp
and all-unspecified"
rate packets 100 within 10
exit
```

```
protocol arp create
    enforcement dynamic "local-mon"
    dynamic-parameters
        rate packets 20 within 10
        exceed-action discard
    exit
exit
protocol icmp create
    enforcement dynamic "local-mon"
    dynamic-parameters
        rate packets 20 within 10
        exceed-action discard
    exit
exit
protocol igmp create
    enforcement dynamic "local-mon"
    dynamic-parameters
        rate packets 20 within 10
        exceed-action discard
    exit
exit
protocol all-unspecified create
    enforcement dynamic "local-mon"
    dynamic-parameters
        rate packets 100 within 10
        exceed-action discard
    exit
exit
```

For the *dcp-dynamic-policy-1* policy configuration:

- The policy contains no static policers. Per-protocol enforcement policers will be instantiated dynamically but only if triggered by a violation of the local-monitoring-policer.
- A local-monitoring-policer is configured for the policy. The configured rate determines the rate of arriving protocol packets at which the policy will trigger the automatic instantiation of dynamic per-protocol policers for the interface.
- Four protocols are configured and they are all associated with the local-monitoring-policer. The all-unspecified protocol will include all other extracted control packets on the interface.
- Each protocol has its own configured dynamic rates that will be used by the dynamic enforcement policers if they are instantiated. Note these rates are lower than previous scenarios (the **within** parameter is 10 seconds instead of 1 second).
- When this DCP policy is associated with an interface, only a single policer (the local-monitoring-policer) will be instantiated (statically/permanently). The per-protocol dynamic policers are only instantiated when there is a violation of the local-monitoring-policer.

The policy is then associated with the interface:

```
*A:PE-1# configure router interface "int-pe1-to-tester"  
*A:PE-1>config>router>if# dist-cpu-protection "dcp-dynamic-policy-1"
```

Step 15. Configure the tester to send:

- 1pps of ARP
- 4pps of ICMP
- 1000pps of IGMP

Start the tester.

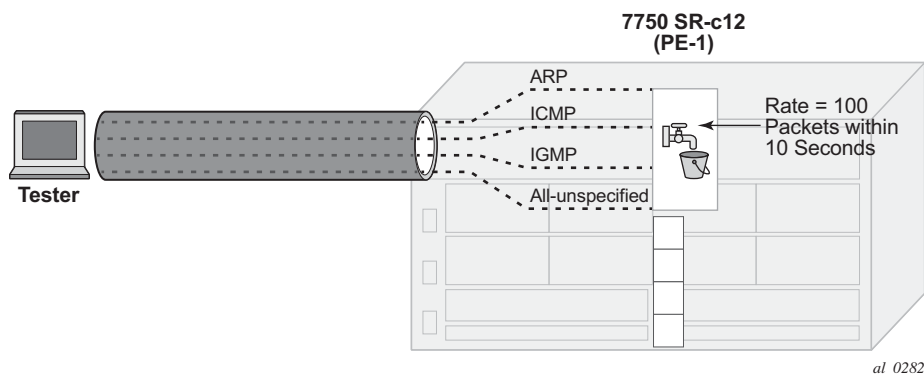


Figure 21: Dynamic Policing – Local Monitor

In [Figure 21](#), the dynamic policers have not been instantiated yet.

Step 16. The local-monitoring-policer will become non-conforming since the aggregate arrival rate of arp+icmp+igmp+all-unspecified packets is greater than the configured local-monitoring-policer rate of 100 packets within 10 seconds. Dynamic enforcement policers will then be instantiated.

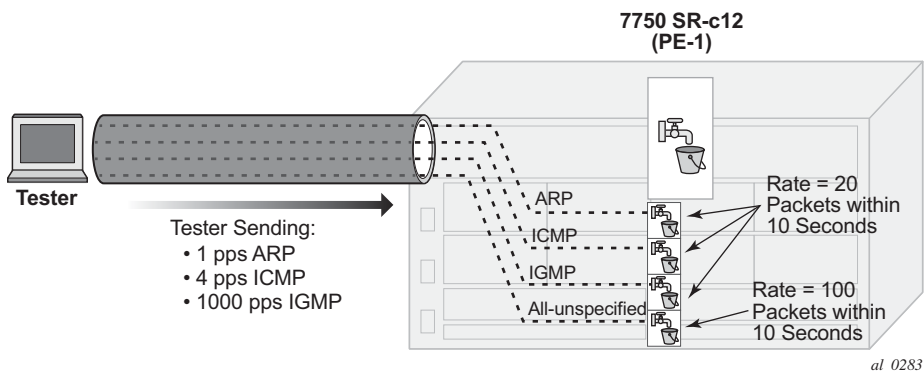


Figure 22: Dynamic Policers Instantiated

The ICMP and IGMP dynamic policers will see violations since their dynamic rates are being exceeded.

```
*A:PE-1# tools dump security dist-cpu-protection violators enforcement interface card 1
=====
Distributed Cpu Protection Current Interface Enforcer Policer Violators
=====
Interface                               Policer/Protocol                       Hld Rem
-----
Violators on Slot-1 Fp-1
-----
int-pel-to-tester                       icmp                                   [D] none
int-pel-to-tester                       igmp                                   [D] none
-----
[S]-Static [D]-Dynamic [M]-Monitor
=====
```

The arp and all-unspecified dynamic policers were instantiated but will be counting down their detection time (if this show command is issued within 30 seconds of the attack starting).

```
*A:PE-1# show router interface "int-pel-to-tester" dist-cpu-protection
=====
Interface "int-pel-to-tester" (Router: Base)
=====
Distributed CPU Protection Policy : dcp-dynamic-policy-1
-----
Statistics/Policer-State Information
=====
Static Policer
-----
No entries found
-----
Local-Monitoring Policer
-----
Policer-Name       : local-mon
Card/FP            : 1/1
Policer-State      : Exceed
Protocols Mapped   : arp, icmp, igmp, all-unspecified
Exceed-Count       : 1097
All Dyn-Plcr Alloc. : True
-----
Dynamic-Policer (Protocol)
-----
Protocol (Dyn-Plcr) : arp
Card/FP             : 1/1
Protocol-State       : Conform
Exceed-Count        : 0
Detec. Time Remain  : 5 seconds
Hold-Down Remain.   : none
Dyn-Policer Alloc.  : True
-----
Protocol (Dyn-Plcr) : icmp
Card/FP             : 1/1
Protocol-State       : Exceed
Exceed-Count        : 31
```

```

Detec. Time Remain : 28 seconds      Hold-Down Remain. : none
Dyn-Policer Alloc. : True

Protocol (Dyn-Plcr) : igmp
Card/FP             : 1/1             Protocol-State    : Exceed
Exceed-Count        : 23867
Detec. Time Remain  : 29 seconds      Hold-Down Remain. : none
Dyn-Policer Alloc. : True

Protocol (Dyn-Plcr) : all-unspecified
Card/FP             : 1/1             Protocol-State    : Conform
Exceed-Count        : 0
Detec. Time Remain  : 5 seconds       Hold-Down Remain. : none
Dyn-Policer Alloc. : True

```

After 30 seconds have passed, the “Detec. Time Remain” for arp and all-unspecified will simply read 0 (zero).

After a few minutes the log events will be collected indicating a non-conformance was seen.

```

*A:PE-1# show log log-id 15
=====
Event Log 15
=====
Description : (Not Specified)
Memory Log contents [size=1024  next event=3  (not wrapped)]

2 2013/04/20 08:56:59.37 EDT WARNING: SECURITY #2067 Base DCPUPROT
"Non conformant network_if "int-pe1-to-tester" on fp 1/1 detected at 04/20/2013 08:52:28.
Policy "dcp-dynamic-policy-1". Policер="icmp" (dynamic). Excd count=2"

1 2013/04/20 08:56:59.37 EDT WARNING: SECURITY #2067 Base DCPUPROT
"Non conformant network_if "int-pe1-to-tester" on fp 1/1 detected at 04/20/2013 08:52:22.
Policy "dcp-dynamic-policy-1". Policер="igmp" (dynamic). Excd count=27"

```

Step 17. Stop the tester.

The dynamic policer detection timers will start counting down since they are no longer seeing violating packets.

```

*A:PE-1# show router interface "int-pe1-to-tester" dist-cpu-protection
=====
Interface "int-pe1-to-tester" (Router: Base)
=====
Distributed CPU Protection Policy : dcp-dynamic-policy-1
-----
Statistics/Policer-State Information
=====
-----
Static Policер
-----
No entries found
-----
-----

```

Configuration

```
Local-Monitoring Policer
-----
Policer-Name       : local-mon
Card/FP            : 1/1                Policer-State      : Exceed
Protocols Mapped   : arp, icmp, igmp, all-unspecified
Exceed-Count       : 1097
All Dyn-Plcr Alloc. : True
-----

Dynamic-Policer (Protocol)
-----
Protocol(Dyn-Plcr) : arp
Card/FP            : 1/1                Protocol-State      : Conform
Exceed-Count       : 0
Detec. Time Remain : 0 seconds          Hold-Down Remain.  : none
Dyn-Policer Alloc. : True

Protocol(Dyn-Plcr) : icmp
Card/FP            : 1/1                Protocol-State      : Exceed
Exceed-Count       : 511
Detec. Time Remain : 14 seconds          Hold-Down Remain.  : none
Dyn-Policer Alloc. : True

Protocol(Dyn-Plcr) : igmp
Card/FP            : 1/1                Protocol-State      : Exceed
Exceed-Count       : 345550
Detec. Time Remain : 18 seconds          Hold-Down Remain.  : none
Dyn-Policer Alloc. : True

Protocol(Dyn-Plcr) : all-unspecified
Card/FP            : 1/1                Protocol-State      : Conform
Exceed-Count       : 0
Detec. Time Remain : 0 seconds          Hold-Down Remain.  : none
Dyn-Policer Alloc. : True
-----
=====
```

After 30 seconds there are no more violators.

```
*A:PE-1# tools dump security dist-cpu-protection violators enforcement interface card 1
=====
Distributed Cpu Protection Current Interface Enforcer Policer Violators
=====
Interface                Policer/Protocol                Hld Rem
-----
Violators on Slot-1 Fp-1
-----
[S]-Static [D]-Dynamic [M]-Monitor
=====
```

The dynamic policer pool Hi-WaterMark for card 1 fp 1 shows 4 since the highest number of dynamic policers allocated at any one time on the card/fp was 4.

```
*A:PE-1# show card 1 fp 1 dist-cpu-protection
=====
Card : 1 Forwarding Plane(FP) : 1
=====
Dynamic Enforcement Policer Pool : 1000
-----
-----
Statistics Information
-----
Dynamic-Policers Currently In Use      : 0
Hi-WaterMark Hit Count                 : 4
Hi-WaterMark Hit Time                  : 04/20/2013 08:52:22 UTC
Dynamic-Policers Allocation Fail Count : 0
-----
=====
```

A few minutes later the log events indicate that the flood has ended.

```
*A:PE-1# show log log-id 15
=====
Event Log 15
=====
Description : (Not Specified)
Memory Log contents [size=1024 next event=5 (not wrapped)]

4 2013/04/20 09:01:59.39 EDT WARNING: SECURITY #2073 Base DCPUPROT
"Network_if "int-pe1-to-tester" on fp 1/1 newly conformant at 04/20/2013 08:58:39. Policy
"dcp-dynamic-policy-1". Policer="igmp"(dynamic). Excd count=345550"

3 2013/04/20 09:01:59.39 EDT WARNING: SECURITY #2073 Base DCPUPROT
"Network_if "int-pe1-to-tester" on fp 1/1 newly conformant at 04/20/2013 08:58:35. Policy
"dcp-dynamic-policy-1". Policer="icmp"(dynamic). Excd count=511"
```

Conclusion

Distributed CPU Protection (DCP) offers a powerful rate limiting function for control protocol traffic that is extracted from the data path and sent to the CPM.

This example has demonstrated how to configure DCP on an interface and what indications SR OS provides to the operator during a potential attack or misconfiguration.

DCP can also be deployed in scenarios where per-SAP-per-protocol rate limiting is useful, such as for subscriber management in a subscriber per-vlan scenario. A DCP policy can be assigned to an MSAP policy on a Broadband Network Gateway, for example, to limit traffic related to certain protocols and to discard certain protocols. When deployed in a subscriber management scenario, DCP can help isolate SAPs (subscribers) from each other and even isolate protocols from each other within an individual SAP (subscriber). Many of the same concepts introduced in this example are applicable when DCP is deployed in a subscriber management application.

Event Handling System

In This Chapter

This section provides information about Event Handling Systems (EHS).

Topics in this section include:

- [Applicability on page 124](#)
- [Overview on page 125](#)
- [Configuration on page 126](#)
- [Conclusion on page 138](#)

Applicability

This feature is applicable to 7750 SR-7/12, 7750 SR-a4/8, 7750 SR-12E, 7450 ESS-7/12, XRS-20/16c, and 7750-c4/12 with no hardware constraints.

The configuration was tested on SR OS release 13.0.R3.

Overview

The Event Handling System (EHS) is a tool available in SR OS that allows operators to configure user-defined actions on the router in reaction to an event. The event is referred to as the trigger, and can be all or part of any event message generated by the event-control framework. The user-defined action is controlled by the script-control function. This script-control function references one or more scripts that are able to execute any command available in CLI when the trigger event occurs.

This feature allows for customized automated event management based on specific operator requirements.

The diagram illustrates a network topology with the following components and connections:

- CE-1** (Core Edge 1): IP 172.16.1.4/29, connected to **PE-1**.
- PE-1** (Provider Edge 1): IP 192.0.2.1, connected to **PE-2** and **PE-3**. It contains a **VRRP** component.
- PE-2** (Provider Edge 2): IP 192.0.2.2, connected to **PE-1** and **PE-4**. It contains an **IES** component.
- PE-3** (Provider Edge 3): IP 192.0.2.3, connected to **PE-1** and **PE-5**. It contains an **IES** component.
- PE-4** (Provider Edge 4): IP 192.0.2.4, connected to **PE-2** and **PE-6**.
- PE-5** (Provider Edge 5): IP 192.0.2.5, connected to **PE-3** and **PE-6**.
- PE-6** (Provider Edge 6): IP 192.0.2.6, connected to **PE-4** and **PE-5**.

Network segments and interfaces are labeled with IP addresses and interface identifiers (e.g., .1, .2). A dashed line indicates a VRRP configuration between **PE-1** and **PE-2**.

PE-1 has a connected CE router, CE-1, indexed into a VPLS service. The VPLS has spoke-SDPs to an IES instance on both PE-2 and PE-3, which provide a redundant default gateway to CE-1 using the Virtual Router Redundancy Protocol (VRRP). The subnet used for this redundant gateway connectivity between PE-2 and PE-3 is 172.16.1.0/29. The configuration at PE-3 is shown in the following output. The configuration at PE-2 is similar; the exception being IP addressing and VRRP priority, which is 254.

Page 126

```

        no shutdown
    exit
exit
no shutdown
exit

```

The objective of this configuration example is to ensure that both upstream and downstream traffic are always routed through the same PE router. That is, if PE-3 is VRRP Master, it will attract upstream traffic from CE-1 using the VRRP virtual IP/MAC, but PE-3 should also be the transit PE for downstream traffic destined toward CE-1. When both upstream and downstream traffic transit through the same PE router, it can simplify troubleshooting, QoS configuration, and reconciliation of ingress/egress statistics.

In normal operation, PE-2 is the VRRP Master and advertises the BGP prefix 172.16.1.0/29 with a LOCAL-PREF of 100 (default value). Similarly, PE-3 is the VRRP Backup and advertises the BGP prefix 172.16.1.0/29 with a LOCAL-PREF of 50, using a BGP export policy (route-policy). Therefore, upstream and downstream traffic normally transit through PE-2.

```

*A:PE-3# show router vrrp instance
=====
VRRP Instances
=====
Interface Name          VR Id Own Adm  State      Base Pri  Msg Int
                        IP      Opr  Pol Id      InUse Pri  Inh Int
-----
redundant-interface     1      No  Up   Backup    253      1
                        IPv4      Up   n/a      253      No
    Backup Addr: 172.16.1.1
-----
Instances : 1
=====

*A:PE-3# show router bgp routes 172.16.1.0/29 hunt | match expression "Net-
work|Nexthop|To|Local Pref"
Network      : 172.16.1.0/29
Nexthop      : 192.0.2.3
To           : 192.0.2.6
Res. Nexthop : n/a
Local Pref. : 50
Interface Name : NotAvailable

```

When PE-3 transitions from Backup to Master, it must modify its LOCAL-PREF attribute for prefix 172.16.1.0/29 to a value of 150 to attract downstream traffic destined toward CE-1. Similarly, when PE-3 reverts to Backup, it must advertise the prefix with a LOCAL-PREF of 50.

Script Control

The first step in configuring event handling is to configure a script containing the CLI commands to be executed when the event is triggered. This script can be stored locally on the compact flash, or it can be stored off-node at a defined remote URL, where it can be accessed using FTP or TFTP. When the script is stored locally on the compact flash and the router is equipped with redundant CPMs, the script must be manually saved on the same compact flash on both CPMs, because it is not synchronized automatically.

The first requirement is to modify the LOCAL-PREF of the prefix 172.16.1.0/29 to 150 on transition to VRRP Master. The script, which in this example is held locally on CF1:/, therefore contains the following commands (where the policy-statement, redundant-interface, is the name of the export policy used to advertise the 172.16.1.0/29 prefix):

```
*A:PE-3# file type cf1:/vrrp-master.txt
File: vrrp-master.txt
-----
exit all
config router policy-options
begin
policy-statement redundant-interface
entry 10
action accept
local-preference 150
exit
exit
exit
commit
exit all
```

Note that the script file does not support variables or conditions. Output modifiers such as “[match” and “>” redirect are not supported. If these modifiers are present, the script will continue to run but the pipe/match or redirect will not work. Similarly, there is no syntax checking when the script file is created; instead, the script will fail with a command error. Also, transactional CLI (candidate edit) cannot be used in the script, and will fail with a command error.

Within the system>script-control context, the script is assigned a name and reference is made to its location. It is then put in the no shutdown state. When the script has been defined, a script-policy is configured that calls the previously configured script. The script-policy also specifies a location and filename for a results file that records the successful or unsuccessful conclusion of each script run and each command executed during that run. Each time the script is run, the results are recorded in a file with the name specified for results, followed by an underscore and the date and time that the script was run. A results file must be specified in order for the script to successfully run. The results file can be on the local compact flash, or a remote URL can be specified. As with the script, the script-policy must also be put in the no shutdown state.

```
configure
  system
    script-control
```

```

script "vrrp-master-script"
    location "cfl:/vrrp-master.txt"
    no shutdown
exit
script-policy "vrrp-master-policy"
    results "cfl:/script-results.txt"
    script "vrrp-master-script"
    max-completed 4
    expire-time 3600
    lifetime forever
    no shutdown
exit
exit

```

The optional lifetime command specifies the maximum time that the script may run. The max-completed command specifies the maximum number of script run history status entries to be retained. An optional expire-time command specifies the maximum time that the system keeps the run history status (default is 1 h). The system maintains the script run history table, which has a maximum size of 255 entries. Entries are removed from this table when the max-completed or expire-time thresholds are crossed. If the table reaches the maximum value, subsequent script launch requests are not run until older run history entries expire (due to expire-time), or entries are manually cleared. To manually clear entries, the following command is used:

```
clear system script-control script-policy completed <script-policy-name>
```

The script run history status information can be viewed using the following command:

```

*A:PE-3# show system script-control script-policy "vrrp-master-policy"
=====
Script-policy Information
=====
Script-policy           : vrrp-master-policy
Script-policy Owner     : TiMOS CLI
Administrative status    : enabled
Operational status      : enabled
Script                  : vrrp-master-script
Script owner            : TiMOS CLI
Script source location   : cfl:/vrrp-master.txt
Script results location  : cfl:/script-results.txt
Max running allowed      : 1
Max completed run histories : 4
Max lifetime allowed     : 248d 13:13:56 (21474836 seconds)
Completed run histories  : 1
Executing run histories   : 0
Initializing run histories : 0
Max time run history saved : 0d 01:00:00 (3600 seconds)
Script start error       : N/A
Last change              : 2015/06/09 16:35:07
Max row expire time      : never
=====
Script Run History Status Information
-----
Script Run #4
-----

```

Event Handler

```
Start time      : 2015/06/09 16:36:09      End time       : 2015/06/09 16:36:19
Elapsed time    : 0d 00:00:10             Lifetime      : 0d 00:00:00
State          : terminated                Run exit code  : noError
Result time     : 2015/06/09 16:36:19      Keep history   : 0d 00:59:24
Error time      : never
Results file    : cfl:/script-results.txt_20150609-153607.281824.out
Run exit       : Success
Error          : N/A
```

=====

Event Handler

The second step in configuring event handling is to assign actions to be performed as a result of the event-trigger. These actions are typically one or more configured scripts defined as entries in an action-list. In the following output, the event-handler is assigned the name event-handler-1, and the action-list consists of a single entry. This entry calls the previously configured script-policy vrrp-master-policy (which in turn references the previously defined script vrrp-master-script). If multiple actions are required based on a single event-trigger, they can be configured in the action-list with subsequent entries, which are run in sequence (up to 1500 action-list entries are supported).

For this example, only a single entry is required; therefore, there is a one to one relationship between the event-handler and the action-list entry. Both the entry within the action-list and the handler should be put in the no shutdown state.

```
configure
  log
    event-handling
      handler "event-handler-1"
        action-list
          entry 10
            script-policy "vrrp-master-policy"
            no shutdown
          exit
        exit
      no shutdown
    exit
  exit
```

Event Trigger

The final step in configuring event handling is to configure the event-trigger. The event-trigger defines the event that triggers the running of the script. The event-trigger is based on any event generated by the event-control framework, and can match against the application and event number (event_id). Log filters can also be used to match against specific events using the subject and/or message fields. Regular expressions can be used where required. Note that EHS will not

use any message that is suppressed through event-control configuration, or any event message that is throttled.

The general format for an event in an event log is as follows:

```
nnnn YYYY/MM/DD HH:MM:SS.SS Zone <severity>:<application> # <event_id> <router-name> <subject> description
```

Where:

nnnn	The log entry sequence number
YYYY/MM/DD	The UTC date stamp for the log entry:
	YYYY - Year
	MM - Month
	DD - Date
HH:MM:SS.SS	The UTC time stamp for the event
	HH - Hours (24 hour format)
	MM - Minutes
	SS.SS - Seconds
Zone	Timezone
<severity>	The severity level name of the event
<application>	The application generating the log message
<event_id>	The application's event ID number for the event
<subject>	The subject/affected object for the event
<message>	A textual description of the event

In the example, the following event message is generated when PE-3 becomes VRRP Master:

```
2 2015/06/10 08:23:37.54 UTC MINOR: VRRP #2001 Base Becoming Master
"VRRP virtual router instance 1 on interface redundant-interface (primary address
172.16.1.3) changed state to master"
```

Therefore, the event-trigger configuration is based on an application of VRRP and an event number of 2001 (vrrpNewMaster). In the following output, vrrp 2001 is configured as the event. The trigger-entry is defined as 1, and in this example there is only one trigger event. Up to 1500 trigger-entries can be included, each of which can act as a potential trigger event. The trigger-entry also references the previously configured event-handler-1. (Recall that the event-handler references the script-control, which in turn references the script that should be run.)

Finally, there is a reference to log-filter 1. Without more explicit filtering, event handling will be triggered on any event with the application of VRRP and event number 2001. There may be multiple VRRP instances running on this router, but the requirement is that event handling should only be triggered when the VRRP instance running on redundant-interface transitions to Master at PE-3. Therefore, log-filter 1 is used to define a more explicit match using the message field, which contains an explicit reference to the interface. Both the trigger-entry and the event-handler should be put in the no shutdown state.

```
configure
  log
    filter 1
```

Event Trigger

```
        default-action drop
        entry 10
            action forward
            match
                message eq pattern "interface redundant-interface (primary
                                address 172.16.1.3) changed state to master"
            exit
        exit
    exit
exit

configure
log
    event-trigger
        event "vrrp" 2001
            trigger-entry 1
                event-handler "event-handler-1"
                log-filter 1
                no shutdown
            exit
        no shutdown
    exit
exit
exit
```

The configuration of the example event handling for the failure event (PE-3 transitions to VRRP Master) is now complete. By shutting down the spoke-SDP between PE-1 and PE-2, it is possible to simulate a failure event where the VRRP message path is broken. Therefore, three events are generated.

- The first indicates that PE-3 has become VRRP Master for the interface named redundant-interface.
- The second indicates that a script file has initiated an attempt to execute CLI commands contained in script file vrrp-master.txt.
- The third indicates that the attempt to execute those CLI commands was successful.

```
15 2015/06/10 09:13:59.03 UTC MINOR: VRRP #2001 Base Becoming Master
"VRRP virtual router instance 1 on interface redundant-interface (primary address
172.16.1.3) changed state to master"
```

```
16 2015/06/10 09:14:01.64 UTC MAJOR: SYSTEM #2052 Base CLI 'exec'
"A CLI user has initiated an 'exec' operation to process the commands in the SROS CLI file
cf1:\vrrp-master.txt"
```

```
17 2015/06/10 09:14:10.73 UTC MAJOR: SYSTEM #2053 Base CLI 'exec'
"The CLI user initiated 'exec' operation to process the commands in the SROS CLI
file cf1:\vrrp-master.txt has completed with the result of success"
```


An example of the results file configured in the script-policy is shown as follows. A successful script run shows the commands contained in the script, followed by an indication that the commands were executed.

```
*A:PE-3# file type script-results.txt_20150608-160703.402470.out
File: script-results.txt_20150608-160703.402470.out
-----
*A:PE-3# exit all
*A:PE-3# config router policy-options
*A:PE-3# begin
*A:PE-3# policy-statement redundant-interface
*A:PE-3# entry 10
*A:PE-3# action accept
*A:PE-3# local-preference 150
*A:PE-3# exit
*A:PE-3# exit
*A:PE-3# exit
*A:PE-3# commit
*A:PE-3# exit all
Executed 12 lines in 8.2 seconds from file cf1:\vrrp-master.txt
```

The following outputs confirm that PE-3 is VRRP Master, and that the LOCAL-PREF attribute for prefix 172.16.1.0/29 has changed to a value of 150. The result of this action is that PE-3 will now be the transit router for both upstream and downstream traffic.

```
*A:PE-3# show router vrrp instance
=====
VRRP Instances
=====
Interface Name          VR Id Own Adm  State      Base Pri  Msg Int
                        IP      Opr  Pol Id      InUse Pri  Inh Int
-----
redundant-interface     1      No  Up   Master    253      1
                        IPv4      Up   n/a      253      No
    Backup Addr: 172.16.1.1
-----
Instances : 1
=====

*A:PE-3# show router bgp routes 172.16.1.0/29 hunt | match expression "Net-
work|Nexthop|To|Local Pref"
Network      : 172.16.1.0/29
Nexthop      : 192.0.2.3
To           : 192.0.2.6
Res. Nexthop : n/a
Local Pref. : 150
Interface Name : NotAvailable
```

The event-handler indicates that the referenced script was triggered and run using the command shown in the following output. The Action-List Entry Execution Statistics window provides statistics on the number of times an action (script) was queued to run, and the number of times an error was experienced, both during launch and due to a non-operational admin status. The remainder of the fields in the output are self-explanatory.

Event Trigger

```
*A:PE-3# show log event-handling handler "event-handler-1"
=====
Event Handling System - Handlers
=====
Handler          : event-handler-1
=====
Description       : (Not Specified)
Admin State       : up                      Oper State : up
-----
Handler Action-List Entry
-----
Entry-id         : 10
Description       : (Not Specified)
Admin State       : up                      Oper State : up
Script
  Policy Name     : vrrp-master-policy
  Policy Owner    : TiMOS CLI
  Last Exec       : 06/10/2015 15:41:52
-----
Handler Action-List Entry Execution Statistics
  Enqueued       : 8
  Err Launch     : 0
  Err Adm Status : 0
  Total          : 8
=====
```

The example includes an event-trigger and script to meet the requirements of a fail-forward where PE-3 becomes VRRP Master. Now, configuration is needed for when PE-3 reverts to VRRP Backup. Without another event-trigger and script, PE-3 will continue to advertise the prefix 172.16.1.0/29 with a LOCAL-PREF of 150 and upstream/downstream traffic will be asymmetric through PE-1/PE-3 respectively.

As before, a script is required. Because PE-2 advertises the prefix with a LOCAL-PREF of 100 (default), PE-3 needs to advertise the same prefix with a lower value (50 in the following output), so that PE-2 is the preferred next hop.

```
*A:PE-3# file type cf1:/vrrp-backup.txt
File: vrrp-backup.txt
-----
exit all
config router policy-options
begin
policy-statement redundant-interface
entry 10
action accept
local-preference 50
exit
exit
exit
commit
exit all
```

The script must then be configured within the script-control context, and subsequently referenced in a script-policy as vrrp-backup-policy.

```
configure
  system
    script-control
      script "vrrp-backup-script"
        location "cfl:/vrrp-backup.txt"
        no shutdown
      exit
    script-policy "vrrp-backup-policy"
      results "cfl:/script-revert-results.txt"
      script "vrrp-backup-script"
      max-completed 4
      lifetime forever
      no shutdown
    exit
  exit
```

The event-handler acts as the interface between the configured script-policy and event-trigger. Therefore, a second event-handler is configured with an action-list consisting of a single entry referencing the newly configured vrrp-backup-policy.

```
configure
  log
    event-handling
      handler "event-handler-2"
        action-list
          entry 10
            script-policy "vrrp-backup-policy"
            no shutdown
          exit
        exit
      no shutdown
    exit
  exit
```

Finally, the event-trigger is configured. To revert to VRRP Backup, the application is VRRP and the event number is 2006 (tmnxVrrpBecameBackup). The configuration is filtered on the message field, as before, using log-filter 2, so that it is specific to the interface named redundant-interface.

```
configure
  log
    filter 2
      default-action drop
      entry 10
        action forward
        match
          message eq pattern "interface redundant-interface changed
                               state to backup"
        exit
      exit
    exit
```

Event Trigger

```
configure
  log
    event-trigger
      event "vrrp" 2006
        trigger-entry 1
          event-handler "event-handler-2"
          log-filter 2
          no shutdown
        exit
      no shutdown
    exit
  exit
exit
```

The configuration of the example event handling for the revertive failure event (PE-3 transitions to VRRP Backup) is now complete. By re-enabling the spoke-SDP between PE-1 and PE-2, the VRRP message path is restored, and PE-2 again becomes the VRRP Master. As before, three events are generated. The first indicates that PE-3 has become VRRP Backup for the interface named redundant-interface. The second indicates that a script file has initiated an attempt to execute CLI commands contained in script file vrrp-backup.txt. The third indicates that the attempt to execute those CLI commands was successful.

```
7 2015/06/11 16:17:53.61 UTC MINOR: VRRP #2006 Base Becoming Backup
"VRRP virtual router instance 1 on interface redundant-interface changed state to backup -
current master is 172.16.1.2"

8 2015/06/11 16:17:56.17 UTC MAJOR: SYSTEM #2052 Base CLI 'exec'
"A CLI user has initiated an 'exec' operation to process the commands in the SR OS CLI file
cfl:\vrrp-backup.txt"

9 2015/06/11 16:18:04.64 UTC MAJOR: SYSTEM #2053 Base CLI 'exec'
"The CLI user initiated 'exec' operation to process the commands in the SROS CLI
file cfl:\vrrp-backup.txt has completed with the result of success"
```

The following outputs confirm that PE-3 is VRRP Backup, and that the LOCAL-PREF attribute for prefix 172.16.1.0/29 has changed to a value of 50. The result of this action is that PE-2 will now be the transit router for both upstream and downstream traffic.

```
*A:PE-3# show router vrrp instance
=====
VRRP Instances
=====
```

Interface Name	VR Id	Own	Adm	State	Base Pri	Msg Int
	IP		Opr	Pol Id	InUse Pri	Inh Int
redundant-interface	1	No	Up	Backup	253	1
	IPv4		Up	n/a	253	No
Backup Addr: 172.16.1.1						

```
-----
Instances : 1
=====
```

```
*A:PE-3# show router bgp routes 172.16.1.0/29 hunt | match expression "Net-  
work|Nexthop|To|Local Pref"  
Network      : 172.16.1.0/29  
Nexthop      : 192.0.2.3  
To           : 192.0.2.6  
Res. Nexthop : n/a  
Local Pref. : 50                                Interface Name : NotAvailable
```

Conclusion

EHS allows operators to configure user-defined actions on the router when an event occurs. The event trigger can be anything that is generated by the event-control framework, and explicit filtering is possible using regular expressions. A user-defined action typically runs a script that allows any CLI commands to be executed. Multiple actions are permitted, running multiple scripts if required.

Interface Configuration

In This Section

This section provides interface configuration information for the following topics:

- [Multi-Chassis APS and Pseudowire Redundancy Interworking on page 141](#)
- [Multi-Chassis LAG and Pseudowire Redundancy Interworking on page 163](#)

Multi-Chassis APS and Pseudowire Redundancy Interworking

In This Chapter

This section describes multi-chassis APS and pseudowire redundancy interworking.

Topics in this section include:

- [Applicability on page 142](#)
- [Overview on page 143](#)
- [Configuration on page 146](#)
- [Conclusion on page 162](#)

Applicability

Multi-Chassis Automatic Protection Switching (MC-APS) is supported on 7x50 platforms. The configuration in this chapter was tested on release 13.0.R4 and includes the use of the ATM ports. Refer to the Release Notes for information about support of ATM (and other) MDAs on various platforms as well as MC-APS restrictions.

Overview

MC-APS

MC-APS is an extension to the APS feature to provide not only link redundancy but also node level redundancy. It can protect against nodal failure by configuring the working circuit of an APS group on one node while configuring the protect circuit of the same APS group on a different node.

The two nodes connect to each other with an IP link that is used to establish a signaling path between them. The relevant APS groups in both the working and protection routers must have the same group ID and working circuit, and the protect circuit must have compatible configurations (such as the same speed, framing, and port-type). Signaling is provided using the direct connection between the two service routers. A heartbeat protocol can be used to add robustness to the interaction between the two routers.

Signaling functionality includes support for:

- APS group matching between service routers.
- Verification that one side is configured as a working circuit and the other side is configured as the protect circuit. In case of a mismatch, a trap (incompatible-neighbor) is generated.
- Change in working circuit status is sent from the working router to keep the protection router in sync.
- Protection router, based on K1/K2 byte data, member circuit status, and external request, selects the active circuit and informs the working router to activate or de-activate the working circuit.

Pseudowire Redundancy

Pseudowire (PW) redundancy provides the ability to protect a pseudowire with a pre-provisioned pseudowire and to switch traffic over to the secondary standby pseudowire in case of a SAP and/or network failure condition. Normally, pseudowires are redundant by the virtue of the SDP redundancy mechanism. For instance, if the SDP is an RSVP LSP and is protected by a secondary standby path and/or by Fast-Reroute paths, the pseudowire is also protected.

However, there are a few of applications in which SDP redundancy does not protect the end-to-end pseudowire path when there are two different destination 7x50 PE nodes for the same VLL service. The main use case is the provisioning of dual-homing of a CPE or access node to two 7x50 PE nodes located in different POPs. The other use case is the provisioning of a pair of active and standby BRAS nodes, or active and standby links to the same BRAS node, to provide service resiliency to broadband service subscribers.

Network Topology

The setup in this section contains two access nodes and 4 PE nodes. The access nodes can be any ATM switches that support 1+1 bi-directional APS. [Figure 24](#) shows the physical topology of the setup. [Figure 25](#) shows the use of both MC-APS in the access network and pseudowire redundancy in the core network to provide a resilient end-to-end VLL service.

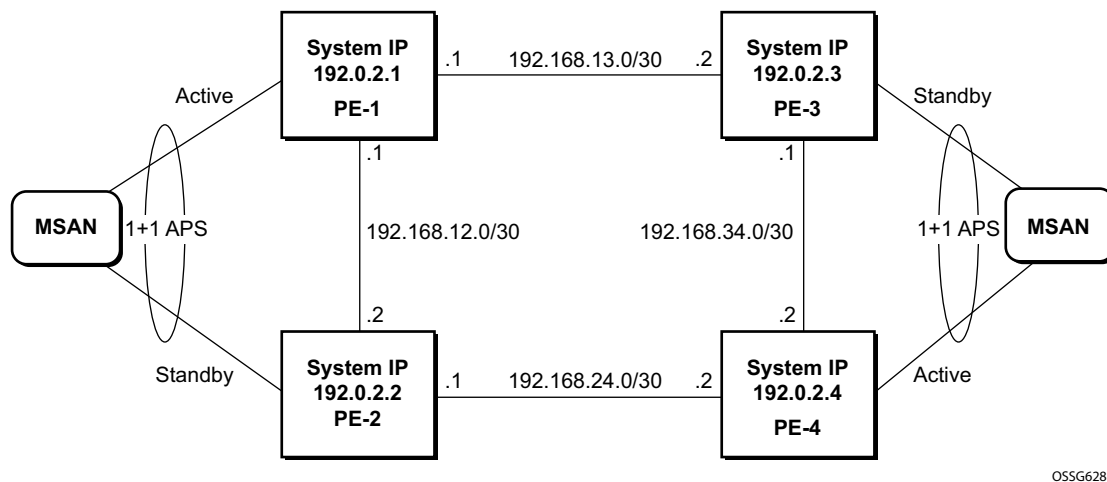
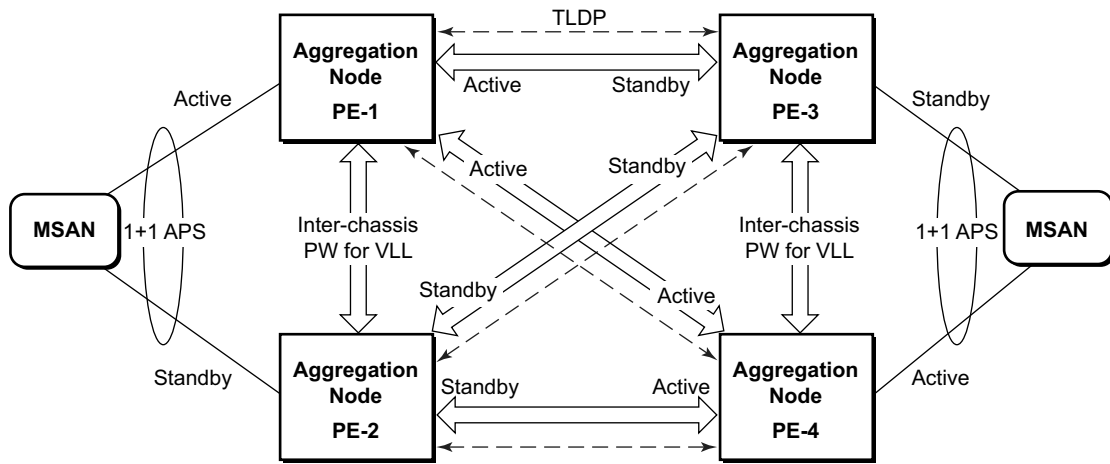
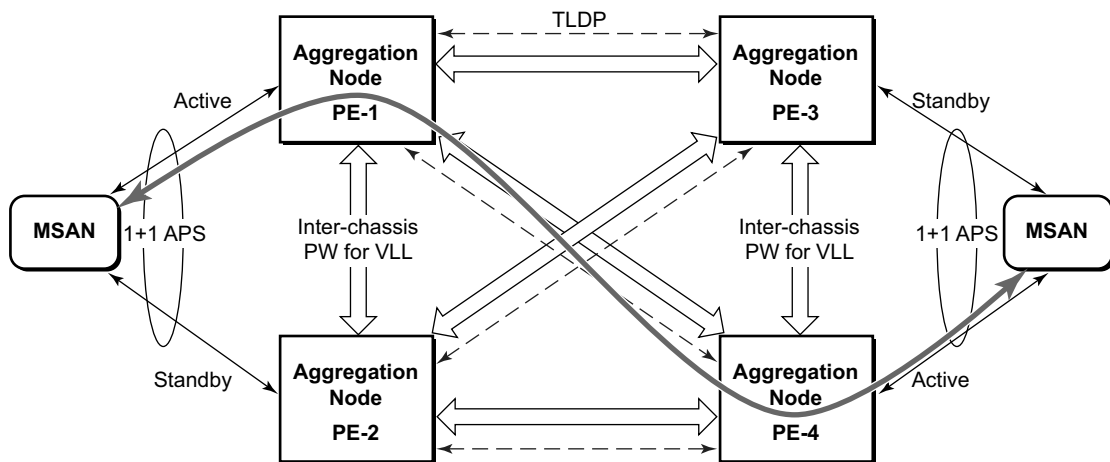


Figure 24: MC-APS Network Topology



OSSG629



OSSG630

Figure 25: Access Node and Network Resilience

Configuration

The following configuration should be completed on the PEs before configuring MC-APS:

- Cards, MDAs and ports
- Interfaces
- IGP configured and converged
- MPLS
- SDPs configured between all PE routers

For the IGP, OSPF or IS-IS can be used. MPLS or GRE can be used for the transport tunnels. For MPLS, LDP or RSVP protocols can be used for signaling MPLS labels. In this example OSPF and LDP are used. The following commands can be used to check if OSPF has converged and to make sure the SDPs are up (for example, on PE-1):

```
*A:PE-1# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type   Proto   Age      Pref
  Next Hop[Interface Name]                        Metric
-----
192.0.2.1/32                                     Local  Local   00h02m12s  0
      system
192.0.2.2/32                                     Remote  OSPF    00h01m17s  10
      192.168.12.2
192.0.2.3/32                                     Remote  OSPF    00h01m05s  10
      192.168.13.2
192.0.2.4/32                                     Remote  OSPF    00h01m08s  10
      192.168.12.2
192.168.12.0/30                                  Local  Local   00h02m13s  0
      int-PE-1-PE-2
192.168.13.0/30                                  Local  Local   00h02m12s  0
      int-PE-1-PE-3
192.168.24.0/30                                  Remote  OSPF    00h01m17s  10
      192.168.12.2
192.168.34.0/30                                  Remote  OSPF    00h01m05s  10
      192.168.13.2
-----
No. of Routes: 8
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====
*A:PE-1#
*A:PE-1# show service sdp
=====
Services: Service Destination Points
=====
SdpId  AdmMTU  OprMTU  Far End      Adm  Opr      Del    LSP    Sig
-----
```

12	0	1556	192.0.2.2	Up	Up	MPLS	L	TLDP
13	0	1556	192.0.2.3	Up	Up	MPLS	L	TLDP
14	0	1556	192.0.2.4	Up	Up	MPLS	L	TLDP

Number of SDPs : 3

Legend: R = RSVP, L = LDP, B = BGP, M = MPLS-TP, n/a = Not Applicable
 I = SR-ISIS, O = SR-OSPF

=====

*A:PE-1#

Step 1. APS configuration on MSANs

The access nodes can be any ATM switches that support 1+1 bi-directional APS. Here is an example on 7670RSP (Routing Switching Platform).

```
Alcatel[RW]> configure
Alcatel[RW]> port 1-6-1-1
Alcatel[RW]> options protection type 1+1
Alcatel[RW]> options protection switching bidirect
Alcatel[RW]> options protection
```

(Standby)

#	Type	Status	Name
1-6-1-1	STM1_IR8	OK	

Protection Group Contains:

```
Protection Port      : 1-6-1-1      (Standby)
Working Port        : 1-5-1-1
Protection Type      : 1+1
Switching Type      : Non-Revertive
Switching Mode      : Bi-directional
Wait-To-Restore Timer : 5 minute(s)
```

Step 2. MC-APS configuration on PE-1 and PE-2

Assuming the link between MSAN and PE-1 is working circuit and the link between MSAN and PE-2 is protection circuit.

Configure APS on the PE-1 port. Specify the system IP address of neighbor node (PE-2) and working-circuit.

```
*A:PE-1# configure port 1/2/1 sonet-sdh
back
no shutdown

*A:PE-1# configure port aps-1
aps
neighbor 192.0.2.2
working-circuit 1/2/1
exit
sonet-sdh
path
atm
exit
no shutdown
```

Configuration

```
exit
exit
no shutdown
```

Configure APS on the PE-2 port. Specify the system IP address of neighbor node (PE-1) and protect-circuit instead of working-circuit.

```
*A:PE-2# configure port 1/2/1 sonet-sdh
back
no shutdown
```

```
*A:PE-2# configure port aps-1
aps
neighbor 192.0.2.1
protect-circuit 1/2/1
exit
sonet-sdh
path
atm
exit
no shutdown
exit
exit
no shutdown
```

The following parameters can be configured under APS optionally.

- advertise-interval — This command specifies the time interval, in 100s of milliseconds, between 'I am operational' messages sent by both protect and working circuits to their neighbor for multi-chassis APS.
- hold-time — This command specifies how much time can pass, in 100s of milliseconds, without receiving an advertise packet from the neighbor before the multi-chassis signaling link is considered not operational.
- revert-time — This command configures the revert-time timer to determine how long to wait before switching back to the working circuit after that circuit has been restored into service.
- switching-mode — This command configures the switching mode for the APS port which can be bi-directional or uni-directional.

Step 3. Verify the APS status on PE-1.

```
*A:PE-1# show port aps-1
=====
SONET/SDH Interface
=====
Description      : APS Group
Interface        : aps-1
Speed            : oc3
Admin Status     : up
Oper Status      : up
```


Multi-Chassis APS and Pseudowire Redundancy

```

Physical Link      : Yes                      Loopback Mode      : none
Single Fiber Mode  : No
Clock Source       : node                     Framing           : sonet
Last State Change  : 09/08/2015 13:45:58      Port IfIndex       : 1358987264
Configured Address : 02:15:ff:00:02:49
Hardware Address   : 02:15:ff:00:02:49
Last Cleared Time   : N/A
J0 String          : 0x01                     Section Trace Mode  : byte
Rx S1 Byte         : 0x00 (stu)                Rx K1/K2 Byte      : 0x00/0x00
Tx S1 Byte         : 0x0a (st3)                Tx DUS/DNU         : Disabled
Rx J0 String (Hex) : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Cfg Alarm          : loc lrdi lb2er-sf slof slos
Alarm Status       :
BER SD Threshold   : 6                        BER SF Threshold    : 3
Hold time up       : 500 milliseconds          Reset On Path Down  : Disabled
Hold time down     : 200 milliseconds

```

Transceiver Data

Transceiver Status : not-equipped

Port Statistics

```

=====
Input                                     Output
-----
Packets                                0                                0
Discards                              0                                0
Unknown Proto Discards                 0
=====

```

*A:PE-1#

Step 4. Verify the MC-APS status and parameters on PE-1 and PE-2

*A:PE-1# show aps detail

APS Group: aps-1

```

=====
Description      : APS Group
Group Id         : 1                      Active Circuit      : 1/2/1
Admin Status     : Up                     Oper Status        : Up
Working Circuit   : 1/2/1                 Protection Circuit   : N/A
Switching-mode    : Bi-directional        Switching-arch     : 1+1(sig-only)
Annex B          : No
Revertive-mode    : Non-revertive          Revert-time (min)   :
Rx K1/K2 byte     : N/A
Tx K1/K2 byte     : N/A
Current APS Status : OK
Multi-Chassis APS : Yes
Neighbor         : 192.0.2.2
Control link state : Up
Advertise Interval : 1000 msec             Hold Time           : 3000 msec
Mode mismatch Cnt : 0                     Channel mismatch Cnt : 0
PSB failure Cnt   : 0                     FEPL failure Cnt    : 0
=====

```

APS Working Circuit - 1/2/1

```

-----
Admin Status      : Up                     Oper Status        : Up
Current APS Status : OK                     No. of Switchovers : 0
Last Switchover    : None                   Switchover seconds : 0

```

Configuration

```
Signal Degrade Cnt : 0                Signal Failure Cnt : 0
Last Switch Cmd    : N/A              Last Exercise Result : N/A
Tx L-AIS           : None
```

```
=====
*A:PE-1#
```

Detailed parameters of the APS configuration on PE-1 can be verified, shown above. The admin/oper status of APS group 1 shows up/up. K1/K2 byte shows N/A as APS 1+1 exchanges that information through the protection circuit.

The admin/oper status of the working circuit (the link between MSAN and PE-1) is up/up.

```
*A:PE-2# show aps detail
```

```
=====
APS Group: aps-1
```

```
=====
Description      : APS Group
Group Id         : 1
Admin Status     : Up
Working Circuit   : N/A
Switching-mode   : Bi-directional
Annex B          : No
Revertive-mode   : Non-revertive
Rx K1/K2 byte    : 0x00/0x05 (No-Req on Protect)
Tx K1/K2 byte    : 0x00/0x05 (No-Req on Protect)
Current APS Status : OK
Multi-Chassis APS : Yes
Neighbor         : 192.0.2.1
Control link state : Up
Advertise Interval : 1000 msec
Mode mismatch Cnt : 0
PSB failure Cnt  : 0
Active Circuit    : N/A
Oper Status      : Up
Protection Circuit : 1/2/1
Switching-arch   : 1+1 (sig-only)
Revert-time (min) :
Hold Time        : 3000 msec
Channel mismatch Cnt : 0
FEPL failure Cnt : 1
```

```
-----
APS Working Circuit - Neighbor
```

```
-----
Admin Status      : N/A
Current APS Status : OK
Last Switchover   : None
Signal Degrade Cnt : 0
Last Switch Cmd   : No Cmd
Tx L-AIS          : None
Oper Status       : N/A
No. of Switchovers : 0
Switchover seconds : 0
Signal Failure Cnt : 1
Last Exercise Result : Unknown
```

```
-----
APS Protection Circuit - 1/2/1
```

```
-----
Admin Status      : Up
Current APS Status : OK
Last Switchover   : None
Signal Degrade Cnt : 0
Last Switch Cmd   : No Cmd
Tx L-AIS          : None
Oper Status       : Up
No. of Switchovers : 0
Switchover seconds : 0
Signal Failure Cnt : 0
Last Exercise Result : Unknown
```

```
=====
*A:PE-2#
```

Detailed parameters of the APS configuration on PE-2 can be verified, as above. The admin/oper status of APS group 1 shows up/up. Both Rx and Tx of the K1/K2 byte are in the status of 0x00/0x05 (No-Req on Protect) as there is no failure or force-switchover request.

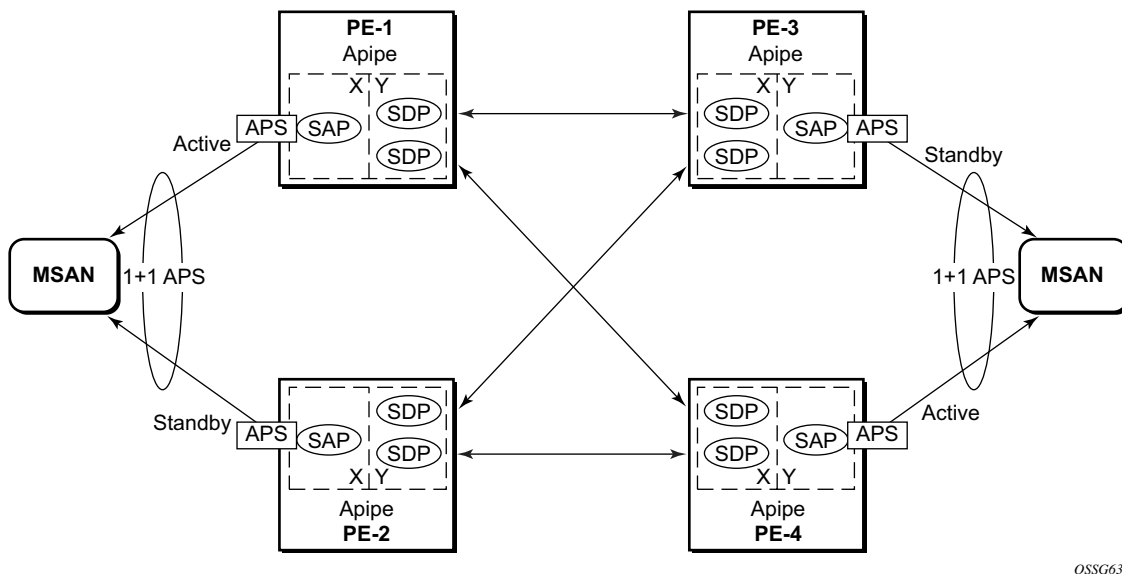
The admin/oper status of the protection circuit (the link between MSAN and PE-2) is up/up.

Step 5. MC-APS configuration on PE-3 and PE-4

The MC-APS configuration on PE-3 and PE-4 is similar to the configuration on PE-1 and PE-2. Configure the working circuit on PE-4 and the protection circuit on PE-3.

Step 6. Pseudowire configuration

Configure an Apipe service on every PE and create endpoints X and Y. Associate the SAPs and spoke SDPs with the endpoints, as shown in [Figure 26](#).



OSSG631

Figure 26: Association of SAPs/SDPs and Endpoints

```
*A:PE-1>config>service>apipe# info
-----
endpoint "x" create
exit
endpoint "y" create
exit
sap aps-1:0/32 endpoint "x" create
exit
spoke-sdp 13:1 endpoint "y" create
```

Configuration

```
exit
spoke-sdp 14:1 endpoint "y" create
exit
no shutdown
-----
*A:PE-1>config>service>apipe#
```

Syntax `aps-1:0/32` above specifies the APS group and VPI/VCI of the ATM circuit (`aps-id:vpi/vci`).

Likewise, an Apipe service, with endpoints, SAPs and spoke SDPs must be configured on the other PE routers.

Step 7. Pseudowire verification

```
*A:PE-1# show service service-using
=====
Services
=====
ServiceId      Type      Adm  Opr  CustomerId  Service Name
-----
1              Apipe     Up   Up   1           _tmnx_InternalIesService
2147483648     IES       Up   Down 1       _tmnx_InternalIesService
2147483649     intVpls   Up   Down 1       _tmnx_InternalVplsService
-----
Matching Services : 3
-----

*A:PE-1#
*A:PE-2# show service service-using
=====
Services
=====
ServiceId      Type      Adm  Opr  CustomerId  Service Name
-----
1              Apipe     Up   Down 1       _tmnx_InternalIesService
2147483648     IES       Up   Down 1       _tmnx_InternalIesService
2147483649     intVpls   Up   Down 1       _tmnx_InternalVplsService
-----
Matching Services : 3
-----

*A:PE-2#
*A:PE-3# show service service-using
=====
Services
=====
ServiceId      Type      Adm  Opr  CustomerId  Service Name
-----
1              Apipe     Up   Down 1       _tmnx_InternalIesService
2147483648     IES       Up   Down 1       _tmnx_InternalIesService
2147483649     intVpls   Up   Down 1       _tmnx_InternalVplsService
-----
Matching Services : 3
-----
=====
```

```

*A:PE-3#
*A:PE-4# show service service-using
=====
Services
=====
ServiceId      Type      Adm  Opr  CustomerId Service Name
-----
1              Apipe     Up   Up   1
2147483648     IES       Up   Down 1      _tmnx_InternalIesService
2147483649     intVpls   Up   Down 1      _tmnx_InternalVplsService
-----
Matching Services : 3
-----
=====
*A:PE-4#

```

Note that only the Apipe services on PE-1 and PE-4 show as up but they are down on PE-2 and PE-3 as the APS configuration on these nodes is in protection status.

Step 8. Verify SDP status

An example on PE-2:

```

*A:PE-2# show service id 1 sdp 23:1 detail
=====
Service Destination Point (Sdp Id : 23:1) Details
=====
-----
Sdp Id 23:1  -(192.0.2.3)
-----
Description      : (Not Specified)
SDP Id           : 23:1                               Type           : Spoke
Spoke Descr      : (Not Specified)
Split Horiz Grp  : (Not Specified)
VC Type          : AAL5SDU                             VC Tag          : 0
Admin Path MTU   : 0                                   Oper Path MTU   : 1556
Delivery         : MPLS
Far End          : 192.0.2.3
Tunnel Far End   : 192.0.2.3                           LSP Types       : LDP

Admin State      : Up                                   Oper State      : Up
Acct. Pol        : None                                Collect Stats   : Disabled
Ingress Label    : 131067                              Egress Label    : 131066
Ingr Mac Fltr-Id : n/a                                  Egr Mac Fltr-Id : n/a
Ingr IP Fltr-Id  : n/a                                  Egr IP Fltr-Id  : n/a
Admin ControlWord : Preferred                          Oper ControlWord : True
Admin BW(Kbps)   : 0                                   Oper BW(Kbps)   : 0
BFD Template     : None
BFD-Enabled      : no                                  BFD-Encap       : ipv4
Last Status Change : 09/08/2015 13:48:30              Signaling       : TLDP
Last Mgmt Change  : 09/08/2015 13:48:22
Endpoint         : Y                                   Precedence      : 4
PW Status Sig     : Enabled
Class Fwding State : Down
Flags            : None
Local Pw Bits     : lacIngressFault lacEgressFault pwFwdingStandby

```

Configuration

```
Peer Pw Bits      : lacIngressFault lacEgressFault pwFwdingStandby
Peer Fault Ip     : None
Peer Vccv CV Bits : lspPing bfdFaultDet
Peer Vccv CC Bits : pwe3ControlWord mplsRouterAlertLabel

Ingress Qos Policy : (none)          Egress Qos Policy : (none)
Ingress FP QGrp    : (none)          Egress Port QGrp   : (none)
Ing FP QGrp Inst   : (none)          Egr Port QGrp Inst: (none)

KeepAlive Information :
Admin State         : Disabled        Oper State          : Disabled
Hello Time          : 10              Hello Msg Len       : 0
Max Drop Count      : 3              Hold Down Time      : 10

---snip---
-----
Number of SDPs : 1
-----
=====
*A:PE-2#
```

Peer Pw Bits shows the status of the pseudowire on the peer node. In this example, both the local node (PE-2) as the remote node (PE-3) are sending the lacIngressFault lacEgressFault and pwFwdingStandby flags. This is because the Apipe service on these nodes is down because the MC-APS is in protection status.

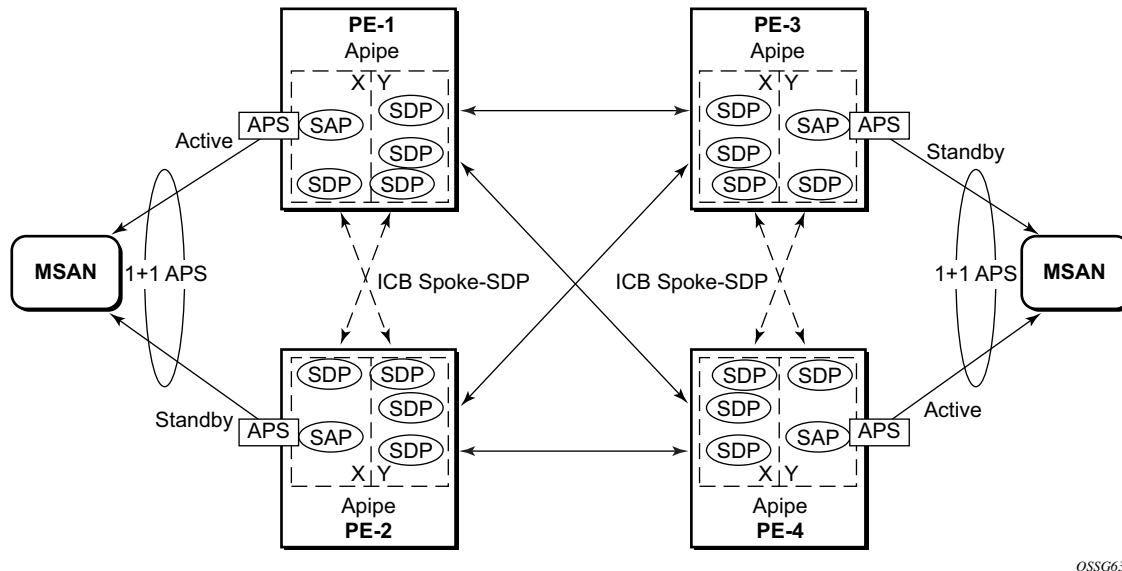
In case of failure, the access link can be protected by MC-APS. An MPLS network failure can be protected by pseudowire redundancy. Node failure can be protected by the combination of MC-APS and pseudowire redundancy.

Step 9. Inter-Chassis Backup (ICB) pseudowire configuration.

Configuring Inter-Chassis Backup (ICB) is optional. It can reduce traffic impact by forwarding traffic on ICB spoke SDPs during MC-APS switchover. The ICB spoke SDP cannot be added to the endpoint if the SAP is not part of an MC-APS (or MC-LAG) instance. Conversely, a SAP which is not part of a MC-APS (or MC-LAG) instance cannot be added to an endpoint which already has an ICB spoke SDP. Forwarding between ICBs is blocked on the same node. The user has to explicitly indicate the spoke SDP is actually an ICB at creation time. Figure 5 shows some setup examples where ICBs are required.

Note that after configuring ICB spoke SDPs the Apipe will be in admin/oper up/up status on all PE routers.

Configure ICB SDPs and associate them to endpoints is shown in [Figure 27](#).



OSSG632

Figure 27: ICB Spoke SDPs and Association with the Endpoints

Two ICB spoke SDPs must be configured in the Apipe service on each PE router, one in each endpoint. The same SDP IDs can be used for the ICBs since the far-end will be the same. However, the vc-id must be different. The ICB spoke SDPs must cross, meaning one end should be associated with endpoint X and the other end (on the other PE) should be associated with endpoint Y.

An ICB is always the last forwarding resort. Only one spoke SDP will be forwarding. If there is an ICB and an MC-APS SAP in an endpoint, the ICB will only forward if the SAP goes down. If an ICB resides in an endpoint together with other spoke SDPs the ICB will only forward if there is no other active spoke SDP.

The following shows the additional configuration for ICB on each PE:

```
*A:PE-1# configure service
  apipe 1
    spoke-sdp 12:1 endpoint "X" icb create
    exit
    spoke-sdp 12:2 endpoint "Y" icb create
    exit

*A:PE-2# configure service
  apipe 1
    spoke-sdp 21:1 endpoint "Y" icb create
    exit
    spoke-sdp 21:2 endpoint "X" icb create
    exit
```

Configuration

```
*A:PE-3# configure service
  apipe 1
    spoke-sdp 34:1 endpoint "X" icb create
    exit
    spoke-sdp 34:2 endpoint "Y" icb create
    exit

*A:PE-4# configure service
  apipe 1
    spoke-sdp 43:1 endpoint "Y" icb create
    exit
    spoke-sdp 43:2 endpoint "X" icb create
    exit
```

Step 10. Verification of active objects for each endpoint

The following command shows which objects are configured for each endpoint and which is the active object at this moment:

```
*A:PE-1# show service id 1 endpoint
=====
Service 1 endpoints
=====
Endpoint name           : X
Description              : (Not Specified)
Creation Origin          : manual
Revert time             : 0
Act Hold Delay           : 0
Tx Active                : aps-1:0/32
Tx Active Up Time       : 0d 00:02:58
Revert Time Count Down  : N/A
Tx Active Change Count   : 1
Last Tx Active Change    : 09/08/2015 13:48:15
-----
Members
-----
SAP      : aps-1:0/32                               Oper Status: Up
Spoke-sdp: 12:1 Prec:4 (icb)                         Oper Status: Up
=====
Endpoint name           : Y
Description              : (Not Specified)
Creation Origin          : manual
Revert time             : 0
Act Hold Delay           : 0
Tx Active (SDP)          : 14:1
Tx Active Up Time       : 0d 00:02:35
Revert Time Count Down  : N/A
Tx Active Change Count   : 2
Last Tx Active Change    : 09/08/2015 13:48:38
-----
Members
-----
Spoke-sdp: 12:2 Prec:4 (icb)                         Oper Status: Up
Spoke-sdp: 13:1 Prec:4                               Oper Status: Up
Spoke-sdp: 14:1 Prec:4                               Oper Status: Up
```



```
=====
=====
*A:PE-1#
```

Note that on PE-1 both the SAP and the spoke SDP 14:1 are active. The other objects do not forward traffic.

Step 11. Other types of setups

The following figures show other setups that combine MC-APS and pseudowire redundancy.

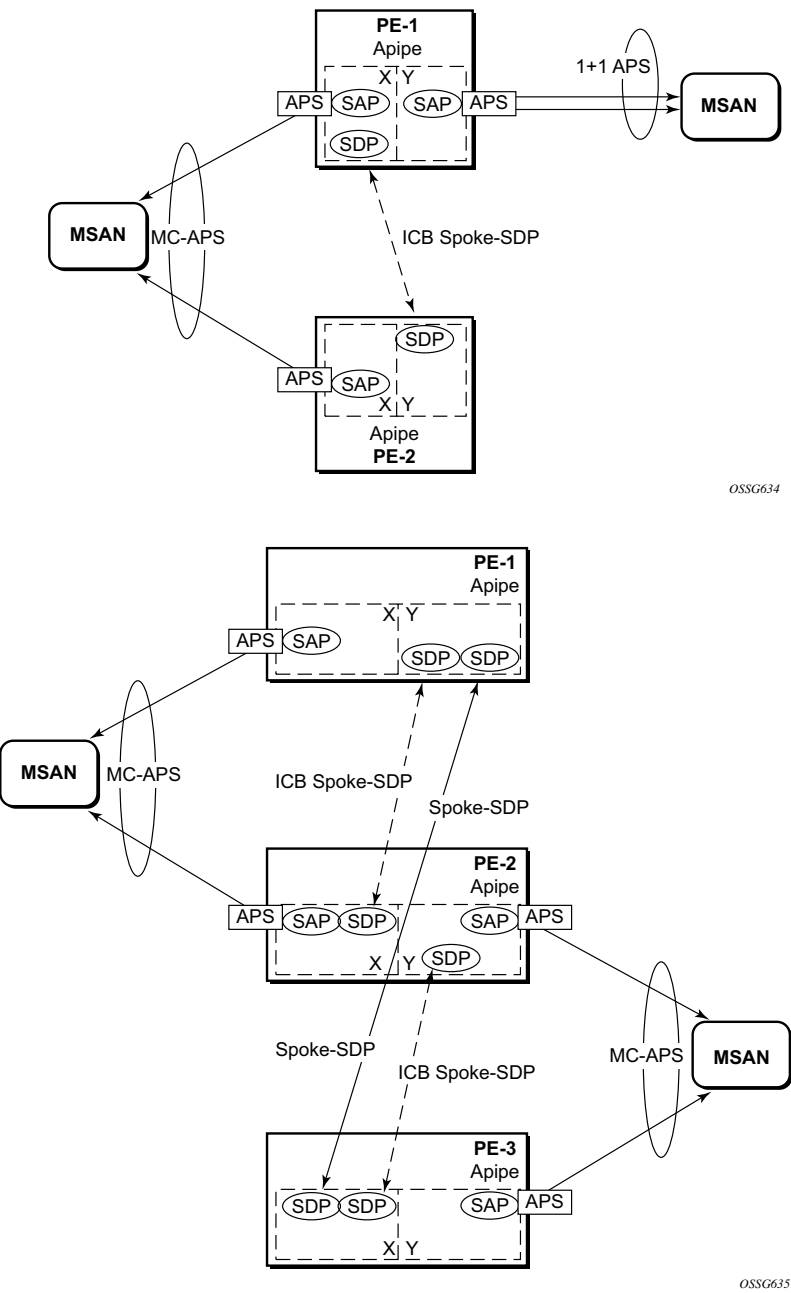
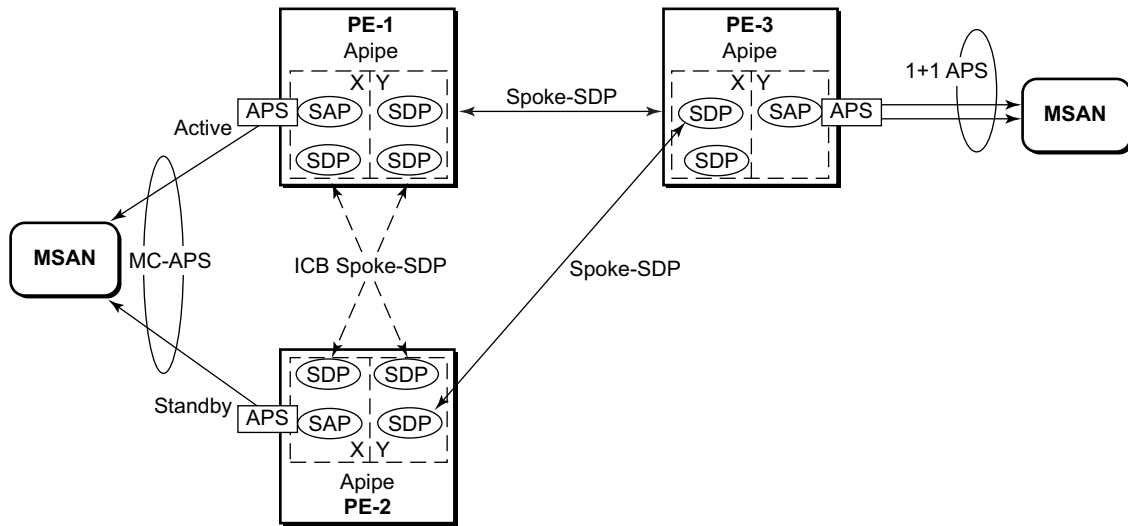
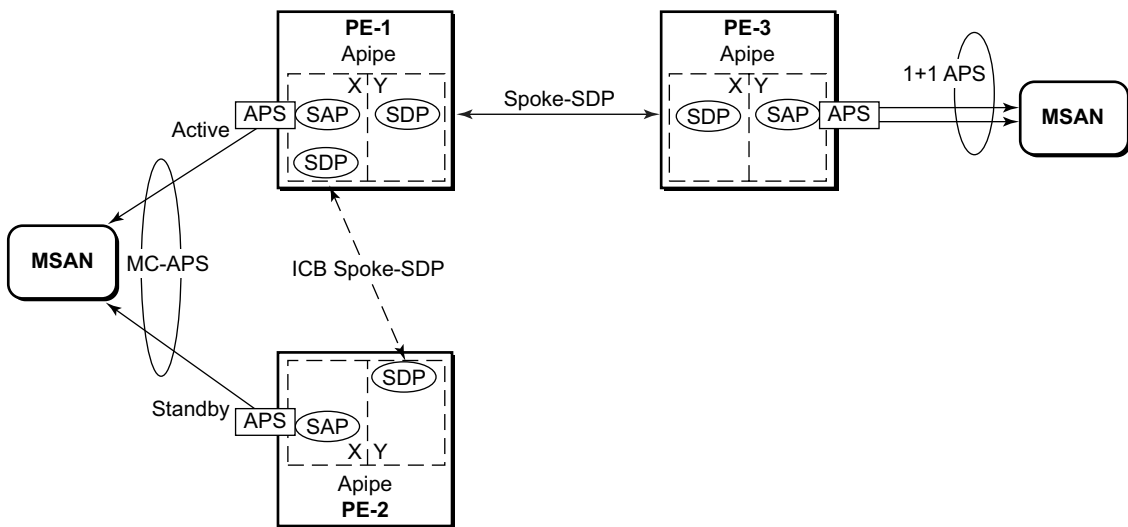


Figure 28: Additional Setup Example 1

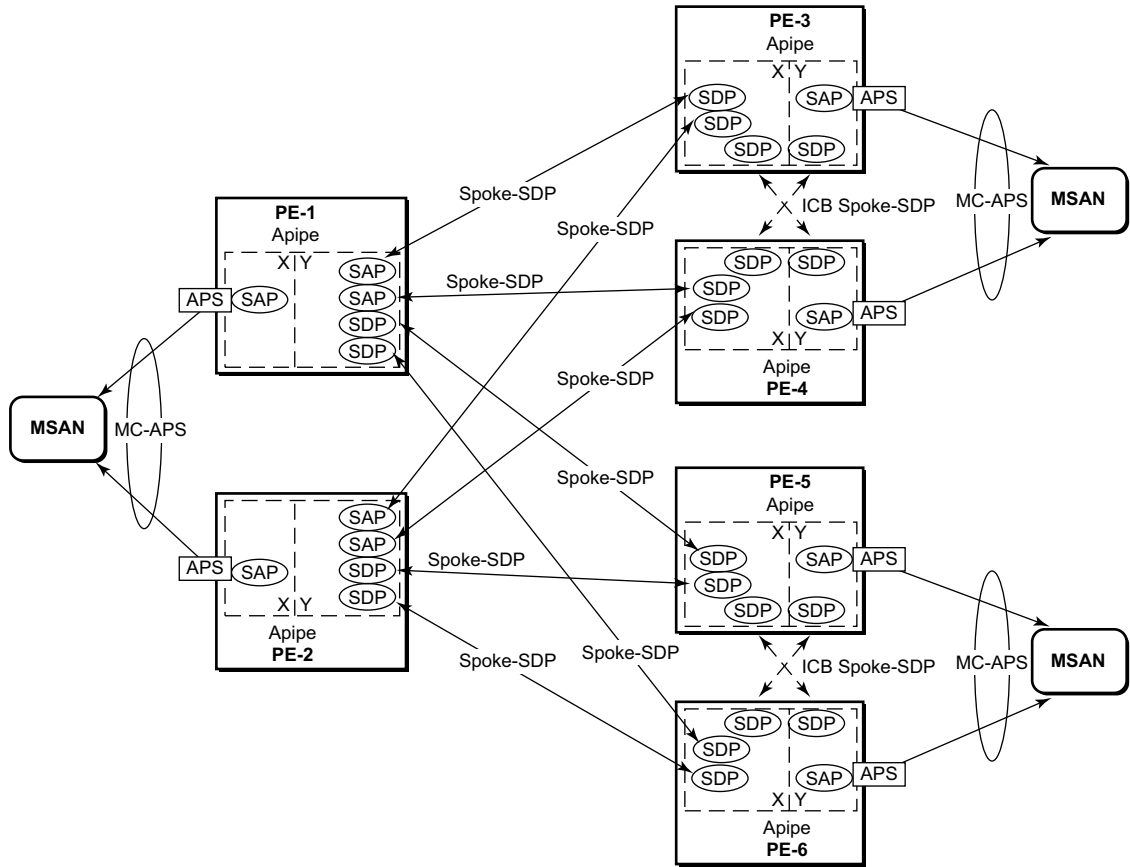


OSSG636



OSSG637

Figure 29: Additional Setup Example 2 (Part 1)



OSSG638

Figure 30: Additional Setup Example 2 (Part 2)

Forced Switchover

MC-APS convergence can be forced with the **tools perform aps** command:

```
*A:PE-1# tools perform aps force
  - force <aps-id> {protect|working} [number <number>]

<aps-id>                : aps-<group-id>
                        aps                - keyword
                        group-id           - [1..64]
<protect|working>       : keyword
<number>                : [1-2]
```

After the forced switchover it is important to clear the forced switchover:

```
*A:PE-1# tools perform aps clear
  - clear <aps-id> {protect|working} [number <number>]

<aps-id>                : aps-<group-id>
                        aps                - keyword
                        group-id           - [1..64]
<protect|working>       : protect|working
<number>                : [1-2]
```

Conclusion

In addition to Multi-Chassis LAG, Multi-Chassis APS provides a solution for both network redundancy and access node redundancy. It supports ATM VLL and Ethernet VLL with ATM SAP. Access links and PE nodes are protected by APS and the MPLS network is protected by pseudowire redundancy/FRR. With this feature, Alcatel-Lucent can provide resilient end-to-end solutions.

Multi-Chassis LAG and Pseudowire Redundancy Interworking

In This Chapter

This section provides information about Multi-Chassis Link Aggregation (MC-LAG) and pseudowire redundancy interworking.

Topics in this section include:

- [Applicability on page 164](#)
- [Overview on page 165](#)
- [Configuration on page 168](#)
- [Conclusion on page 186](#)

Applicability

This feature is supported on all 7x50 platforms. MC-LAG is supported only on Ethernet MDAs, and this only for access ports (the given LAG group must be in access mode). The configuration was tested on release 13.0.R4.

Overview

MC-LAG

MC-LAG is an extension to the LAG feature to provide not only link redundancy but also node-level redundancy. This feature provides an Alcatel-Lucent added value solution which is not defined in any IEEE standard.

A proprietary messaging system between redundant-pair nodes supports coordinating the LAG switchover.

Multi-chassis LAG supports LAG switchover coordination: one node connected to two redundant-pair peer nodes with the LAG. During the LACP negotiation, the redundant-pair peer nodes act like a single node using active/stand-by signaling to ensure that only links of one peer node are used at a time.

Pseudowire Redundancy

Pseudowire (PW) redundancy provides the ability to protect a pseudowire with a pre-provisioned pseudowire and to switch traffic over to the secondary standby pseudowire in case of a SAP and/or network failure condition. Normally, pseudowires are redundant by the virtue of the SDP redundancy mechanism. For instance, if the SDP is an RSVP LSP and is protected by a secondary standby path and/or by Fast-Reroute paths, the pseudowire is also protected.

However, there are a few applications in which SDP redundancy does not protect the end-to-end pseudowire path when there are two different destination 7x50 PE nodes for the same VLL service. The main use case is the provision of dual-homing of a CPE or access node to two 7x50 PE nodes located in different POPs. The other use case is the provisioning of a pair of active and standby BRAS nodes, or active and standby links to the same BRAS node, to provide service resiliency to broadband service subscribers.



Figure 32 shows the use of both MC-LAG in the access network and pseudowire redundancy in the core network to provide a resilient end-to-end VLL service between CE-5 and CE-6.

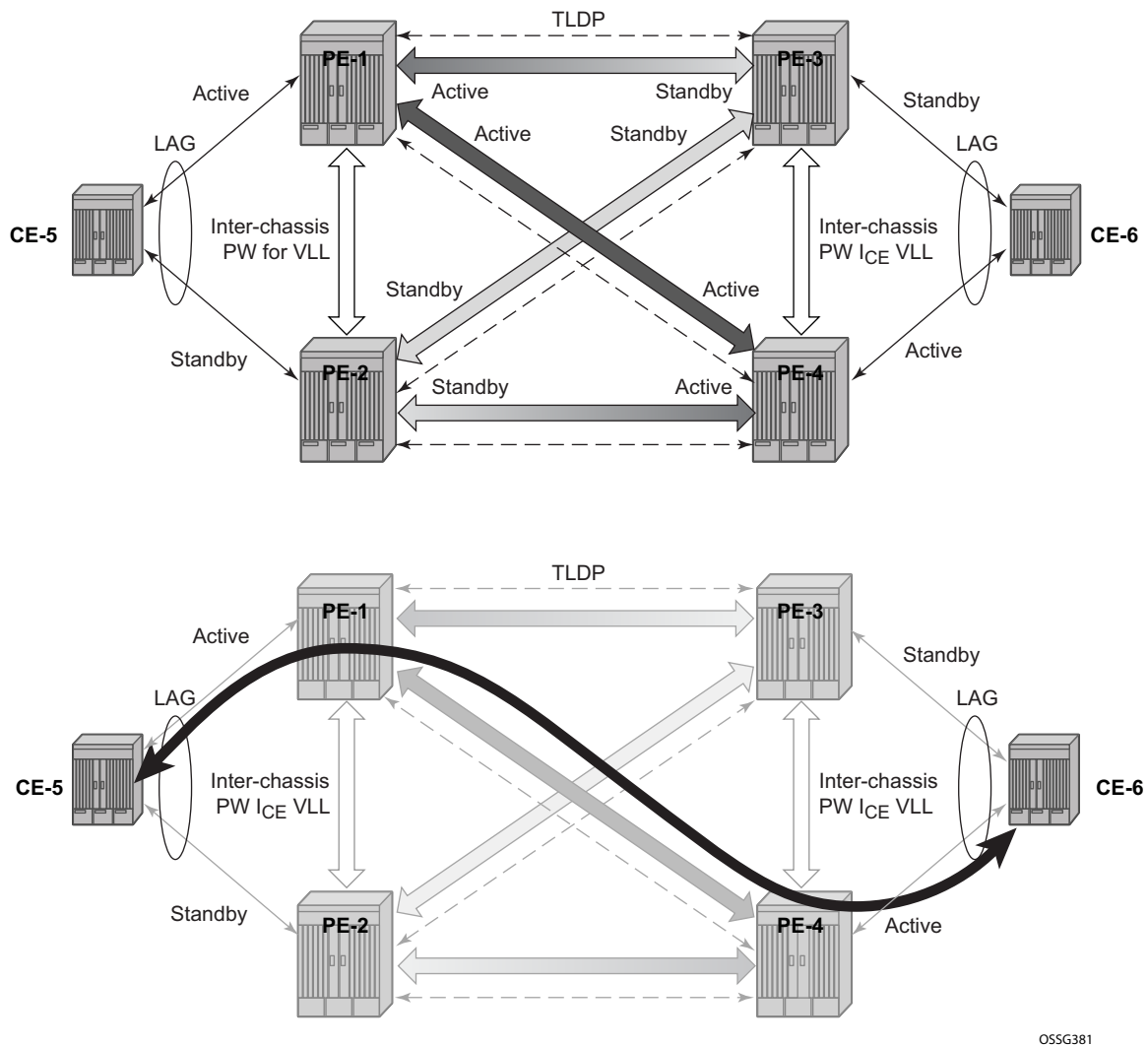


Figure 32: Network Resiliency

Note in [Figure 32](#) that when an SDP is in standby it sends the pseudowire status bit `pwFwdingStandby` to its peer.

Configuration

It is assumed that the following base configuration has been implemented on the PEs:

- Cards, MDAs and ports
- Interfaces
- IGP configured and converged
- MPLS
- SDPs configured between all PE routers

Note that either OSPF and IS-IS can be used as the IGP. Both LDP or RSVP can be used for signaling the transport MPLS labels. Alternatively, GRE can be used for the transport tunnels.

It does not matter if the SDPs are using LDP, RSVP or GRE. In this example OSPF and LDP are used.

The following commands can be used to check if OSPF has converged and to make sure the SDPs are up (for example, on PE-1):

```
*A:PE-1# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type   Proto   Age           Pref
  Next Hop[Interface Name]                        Metric
-----
192.0.2.1/32                                     Local  Local   01h10m55s    0
    system                                         0
192.0.2.2/32                                     Remote OSPF    01h10m01s   10
    192.168.12.2                                   100
192.0.2.3/32                                     Remote OSPF    01h09m55s   10
    192.168.13.2                                   100
192.0.2.4/32                                     Remote OSPF    01h09m05s   10
    192.168.12.2                                   200
192.168.12.0/30                                  Local  Local   01h10m55s    0
    int-PE-1-PE-2                                  0
192.168.13.0/30                                  Local  Local   01h10m55s    0
    int-PE-1-PE-3                                  0
192.168.24.0/30                                  Remote OSPF    01h10m01s   10
    192.168.12.2                                   200
192.168.34.0/30                                  Remote OSPF    01h09m55s   10
    192.168.13.2                                   200
-----
No. of Routes: 8
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====
*A:PE-1#
```

```

*A:PE-1# show service sdp
=====
Services: Service Destination Points
=====
SdpId  AdmMTU  OprMTU  Far End          Adm  Opr          Del    LSP    Sig
-----
12      0        1556    192.0.2.2        Up   Up           MPLS   L      TLDP
13      0        1556    192.0.2.3        Up   Up           MPLS   L      TLDP
14      0        1556    192.0.2.4        Up   Up           MPLS   L      TLDP
-----
Number of SDPs : 3
-----
Legend: R = RSVP, L = LDP, B = BGP, M = MPLS-TP, n/a = Not Applicable
        I = SR-ISIS, O = SR-OSPF
=====
*A:PE-1#

```

Step 1. MC-LAG configuration.

LAG configuration on CEs. Note that this is only included for completeness of the example, any CE device could be used.

Auto-negotiation must be switched off or set to limited on all ports that will be included into the LAG. in order to guarantee a specific port speed¹.

Configure LACP on the LAG. At least one side of the LAG must be configured in **active** mode.

```
*A:CE-5# configure port 1/1/[1..4] ethernet autonegotiate limited
*A:CE-5# configure port 1/1/[1..4] no shutdown
*A:CE-5# configure lag 1
    port 1/1/1 1/1/2 1/1/3 1/1/4
    lacp active
    no shutdown
```

Step 1.1 LAG configuration on PEs.

The PE ports connected to the CEs must be configured as access ports since they will be used in the redundant pseudowire service. The LAG must also be configured in mode access.

Note that the LAG encapsulation type (null | dot1q | qinq) must match the port encapsulation type of the LAG members.

Auto-negotiation must be switched off or configured to limited.

Configure LACP on the LAG. At least 1 side of the LAG (PE or CE) must be configured in **active** mode.

```
*A:PE-1# configure port 1/1/[3..4] ethernet mode access
*A:PE-1# configure port 1/1/[3..4] ethernet autonegotiate limited
*A:PE-1# configure port 1/1/[3..4] no shutdown

*A:PE-1# configure lag 1
    mode access
    port 1/1/3 1/1/4
    lacp active
    no shutdown
```

1. Note that disabling autonegotiation on Gigabit ports is not allowed as the IEEE 802.3 specification for Gigabit Ethernet requires autonegotiation be enabled for far end fault detection.

Step 1.2 MC-LAG configuration on PE1 and PE2

The redundant PEs must act as 1 virtual node toward the CE. They have to be able to communicate the same LACP parameters to the CE side.

The following parameters uniquely identify a LAG instance:

- lacp-key
- system-id
- system-priority

These three parameters must be configured with the same value on both redundant PEs.

Configure multi-chassis redundancy with a peering session (which operates by an IP connection using UDP destination port 1025) toward the redundant PE system address and enable MC-LAG redundancy. The peering session can be configured with MD5 authentication.

```
*A:PE-1# configure redundancy
      multi-chassis
        peer 192.0.2.2 create
          authentication-key "441d0/0RgDhHgzyWpOCTK9zbKjv4GZ/z" hash2
          mc-lag
            lag 1 lacp-key 1 system-id 00:00:00:00:00:01 system-priority 100
            no shutdown
          exit
        no shutdown
      exit
    exit
```

```
*A:PE-2# configure redundancy
      multi-chassis
        peer 192.0.2.1 create
          authentication-key "441d0/0RgDg2CA0JlyzVNQBoRc327b1j" hash2
          mc-lag
            lag 1 lacp-key 1 system-id 00:00:00:00:00:01 system-priority 100
            no shutdown
          exit
        no shutdown
      exit
    exit
```

Step 1.3 MC-LAG verification.

Verify MC peers showing that the authentication and admin state are enabled.

```
*A:PE-1# show redundancy multi-chassis sync
=====
Multi-chassis Peer Table
=====
Peer
-----
Peer IP Address      : 192.0.2.2
Description          : (Not Specified)
Authentication      : Enabled
Source IP Address    : 192.0.2.1
Admin State        : Enabled
Warm standby         : No
Remote warm standby  : No
-----
Sync: Not-configured
-----
=====
*A:PE-1#
```

Step 1.4 Verify MC-LAG peer status and LAG parameters.

```
*A:PE-1# show redundancy multi-chassis mc-lag peer 192.0.2.2
=====
Multi-Chassis MC-Lag Peer 192.0.2.2
=====
Last State chg   : 09/03/2015 07:40:53
Admin State      : Up                               Oper State      : Up
KeepAlive        : 10 deci-seconds                 Hold On Ngbr Failure : 3
-----
Lag Id Lacp      Remote Source Oper   System Id          Sys   Last State Changed
      Key        Lag Id MacLSB MacLSB          Prio
-----
1      1         1      Def    n/a      00:00:00:00:00:01  100   09/03/2015 07:40:54
-----
Number of LAGs : 1
=====
*A:PE-1#
```

Note that there is a fixed keepalive timer of 1 second. The **hold-on-neighbor-failure multiplier** command indicates the interval that the standby node will wait for packets from the active node before assuming a redundant-neighbor failure. The **hold-on-neighbor-failure multiplier** command is configurable in the **config>redundancy>multi-chassis>peer>mc-lag** context. The standby node will also assume a redundant-neighbor failure when there is no route available to the redundant-neighbor.

```
*A:PE-1# configure redundancy
      multi-chassis
        peer 192.0.2.2
          mc-lag
```



```
hold-on-neighbor-failure 10
```

In this example the *lag-id* is 1 on both redundant PEs. This is not mandatory. If the *lag-id* on PE-2 is, for example 2, the following should be configured on PE-1:

```
*A:PE-1# configure redundancy
      multi-chassis
        peer 192.0.2.2
          mc-lag
            lag 1 remote-lag 2 lacp-key 1 system-id 00:00:00:00:00:01 system-pri
              ority 100
```

Step 1.5 Verify MC-LAG

```
*A:PE-1# show lag 1
=====
Lag Data
=====
Lag-id      Adm      Opr      Weighted Threshold Up-Count MC Act/Stdby
-----
1           up       up       No         0         2       active
=====
*A:PE-1#
A:PE-2# show lag 1
=====
Lag Data
=====
Lag-id      Adm      Opr      Weighted Threshold Up-Count MC Act/Stdby
-----
1           up       down     No         0         0       standby
=====
A:PE-2#
```

In this case the LAG on PE-1 is active (operationally up) whereas the LAG on PE-2 is standby (operationally down).

The selection criteria by default is highest number of links and priority. In this example the number of links and the priority of the links is the same on both redundant PEs. Whichever PE's LAG gets the operational status **up** first will be the active.

LAG ports of one PE could be preferred over the other PE by configuring port priority (for example, the following command lowers the priority of the LAG ports on PE-1, thus giving this LAG higher preference). The default priority is 32768.

```
*A:PE-1# configure lag 1 port 1/1/3 1/1/4 priority 10
```

Note that the selection criteria can be configured as highest-count, highest-weight or best-port (the default is highest count).

```
*A:PE1# configure lag 1 selection-criteria
- selection-criteria [best-port|highest-count|highest-weight] [slave-to-partner] [sub-
```

Configuration

```
group-hold-time <hold-time>]
- no selection-criteria

<best-port|highest*> : keywords
<slave-to-partner>   : keyword
<hold-time>          : [0..2000] tenths of a second | infinite
```

If highest-weight is configured, the sum of the weights of the LAG members is considered. The weight of an individual LAG member is calculated as priority 65535 (the default is 32768).

Step 1.6 Verify detailed MC-LAG status on PE-1

```
*A:PE-1# show lag 1 detail
=====
LAG Details
=====
Description          : N/A
-----
Details
-----
Lag-id               : 1                      Mode                : access
Adm                   : up                     Opr                  : up
Thres. Exceeded Cnt   : 2                      Port Threshold       : 0
Thres. Last Cleared   : 09/03/2015 07:42:51    Threshold Action      : down
Dynamic Cost          : false                   Encap Type            : null
Configured Address    : 4a:c4:ff:00:01:41       Lag-IfIndex           : 1342177281
Hardware Address      : 4a:c4:ff:00:01:41       Adapt Qos (access)   : distribute
Hold-time Down        : 0.0 sec                 Port Type             : standard
Per-Link-Hash         : disabled
Include-Egr-Hash-Cfg : disabled                 Forced                : -
Per FP Ing Queuing    : disabled                 Per FP Egr Queuing    : disabled
Per FP SAP Instance   : disabled
LACP                  : enabled                  Mode                   : active
LACP Transmit Intvl   : fast                     LACP xmit stdby       : enabled
Selection Criteria     : highest-count            Slave-to-partner       : disabled
MUX control           : coupled
Subgrp hold time      : 0.0 sec                   Remaining time        : 0.0 sec
Subgrp selected       : 1                         Subgrp candidate       : -
Subgrp count          : 1
System Id             : 4a:c4:ff:00:00:00        System Priority        : 32768
Admin Key              : 32768                     Oper Key               : 1
Prtr System Id        : 4a:c8:ff:00:00:00        Prtr System Priority   : 32768
Prtr Oper Key         : 32768
Standby Signaling     : lacp
Port weight speed     : 0 gbps                     Number/Weight Up      : 2
Weight Threshold      : 0                         Threshold Action       : down

MC Peer Address       : 192.0.2.2                MC Peer Lag-id        : 1
MC System Id          : 00:00:00:00:00:01         MC System Priority     : 100
MC Admin Key          : 1                         MC Active/Standby     : active
MC Lacp ID in use     : true                      MC extended timeout    : false
MC Selection Logic     : local master decided
MC Config Mismatch    : no mismatch
-----
```

Port-id	Adm	Act/Stdby	Opr	Primary	Sub-group	Forced	Prio
1/1/3	up	active	up	yes	1	-	10
1/1/4	up	active	up		1	-	10

Port-id	Role	Exp	Def	Dist	Col	Syn	Aggr	Timeout	Activity
1/1/3	actor	No	No	Yes	Yes	Yes	Yes	Yes	Yes
1/1/3	partner	No	No	Yes	Yes	Yes	Yes	Yes	Yes
1/1/4	actor	No	No	Yes	Yes	Yes	Yes	Yes	Yes
1/1/4	partner	No	No	Yes	Yes	Yes	Yes	Yes	Yes

After changing the LAG port priorities the LAG on PE-1 is in up/up state and the ports are in up/active/up status. This show command also displays actor and partner bits set in the LACP messages.

Step 1.7 MC-LAG configuration on PE-3 and PE-4.

The MC-LAG configuration on PE-3 and PE-4 is similar to the configuration on PE-1 and PE-2. In this case the priority of the LAG port on PE-4 is lowered to obtain the behavior in [Figure 32](#) where LAG on PE-1 and PE-4 is active.

Step 1.8 Pseudowire configuration.

Configure an Epipe service on every PE and create endpoints **x** and **y** (the endpoint names can be any text string). Traffic can only be forwarded between two endpoints, for example, it is not possible for objects associated with the same endpoint to forward traffic to each other.

Associate the SAPs and spoke SDPs with the endpoints as shown in [Figure 33](#).

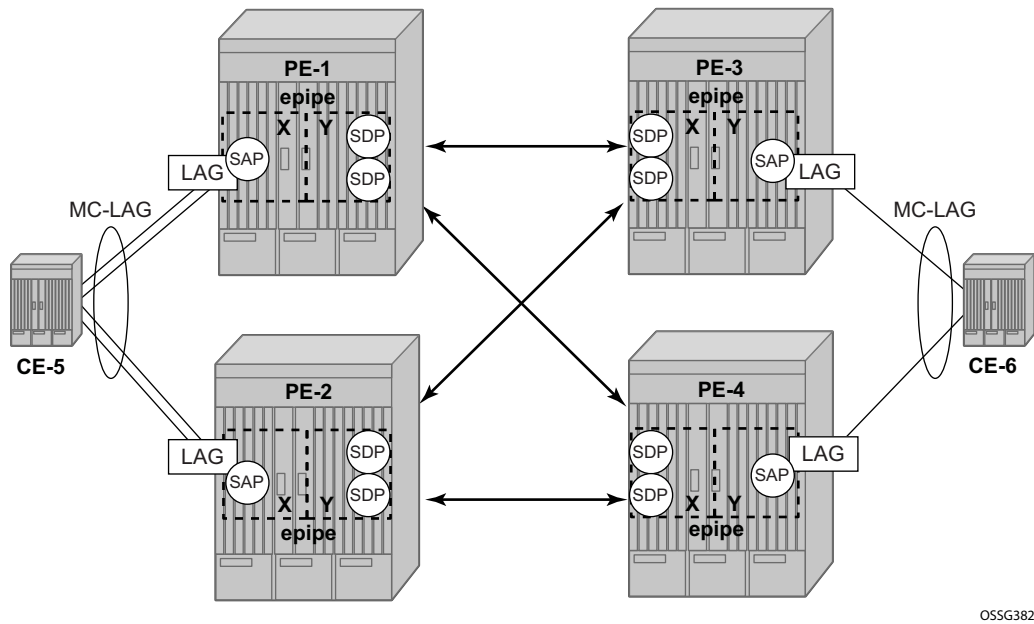


Figure 33: Association of SAPs/SDPs and Endpoints

```
*A:PE-1# configure service
    epipe 1 customer 1 create
        endpoint "X" create
        exit
        endpoint "Y" create
        exit
        sap lag-1 endpoint "X" create
        exit
        spoke-sdp 13:1 endpoint "Y" create
        exit
        spoke-sdp 14:1 endpoint "Y" create
        exit
        no shutdown
    exit
```

Likewise, an Epipe service, endpoints, SAPs and spoke SDPs must be configured on the other PE routers.

Step 1.9 Pseudowire verification.

```

*A:PE-1# show service service-using
=====
Services
=====
ServiceId      Type      Adm  Opr  CustomerId Service Name
-----
1              Epipe     Up   Up   1
2147483648     IES       Up   Down 1      _tmnx_InternalIesService
2147483649     intVpls   Up   Down 1      _tmnx_InternalVplsService
-----
Matching Services : 3
-----

*A:PE-1#
*A:PE-2# show service service-using
=====
Services
=====
ServiceId      Type      Adm  Opr  CustomerId Service Name
-----
1              Epipe     Up   Down 1
2147483648     IES       Up   Down 1      _tmnx_InternalIesService
2147483649     intVpls   Up   Down 1      _tmnx_InternalVplsService
-----
Matching Services : 3
-----

*A:PE-2#
*A:PE-3# show service service-using
=====
Services
=====
ServiceId      Type      Adm  Opr  CustomerId Service Name
-----
1              Epipe     Up   Down 1
2147483648     IES       Up   Down 1      _tmnx_InternalIesService
2147483649     intVpls   Up   Down 1      _tmnx_InternalVplsService
-----
Matching Services : 3
-----

*A:PE-3#
*A:PE-4# show service service-using
=====
Services
=====
ServiceId      Type      Adm  Opr  CustomerId Service Name
-----
1              Epipe     Up   Up   1
2147483648     IES       Up   Down 1      _tmnx_InternalIesService
2147483649     intVpls   Up   Down 1      _tmnx_InternalVplsService
-----
Matching Services : 3
-----

*A:PE-4#

```

The Epipe service on PE-2 and PE-3 is down and up on PE-1 and PE-4. This reflects the standby behavior shown in [Figure 32](#). Note that after configuring ICB spoke SDPs (described later in this document) the Epipe will be in up/up status on all PE routers.

Step 1.10 Verify SDP status

Local pseudowire bits indicate the status of the pseudowire on the PE node. These pseudowire bits will be sent to the peer. Peer pseudowire bits indicate the status of the pseudowire on the peer, as sent by the peer. Here is an example taken on PE-2:

```
*A:PE-2# show service id 1 sdp 23:1 detail
=====
Service Destination Point (Sdp Id : 23:1) Details
=====
-----
Sdp Id 23:1 - (192.0.2.3)
-----
Description      : (Not Specified)
SDP Id           : 23:1                               Type           : Spoke
Spoke Descr      : (Not Specified)
VC Type          : Ether                               VC Tag          : n/a
Admin Path MTU   : 0                                   Oper Path MTU   : 1556
Delivery         : MPLS
Far End          : 192.0.2.3
Tunnel Far End   : 192.0.2.3                           LSP Types       : LDP
Hash Label       : Disabled                             Hash Lbl Sig Cap : Disabled
Oper Hash Label  : Disabled

Admin State      : Up                                   Oper State       : Up
Acct. Pol        : None                                Collect Stats    : Disabled
Ingress Label    : 262138                               Egress Label     : 262138
Ingr Mac Fltr-Id : n/a                                   Egr Mac Fltr-Id  : n/a
Ingr IP Fltr-Id  : n/a                                   Egr IP Fltr-Id   : n/a
Ingr IPv6 Fltr-Id : n/a                                Egr IPv6 Fltr-Id : n/a
Admin ControlWord : Not Preferred                       Oper ControlWord  : False
Admin BW(Kbps)   : 0                                   Oper BW(Kbps)    : 0
BFD Template     : None
BFD-Enabled      : no                                   BFD-Encap        : ipv4
Last Status Change : 09/03/2015 07:45:49                Signaling         : TLDP
Last Mgmt Change  : 09/03/2015 07:45:39
Endpoint         : Y                                   Precedence        : 4
PW Status Sig     : Enabled
Force Vlan-Vc     : Disabled                           Force Qinq-Vc     : Disabled
Class Fwding State : Down
Flags             : None
Local Pw Bits     : lacIngressFault lacEgressFault pwFwdingStandby
Peer Pw Bits      : lacIngressFault lacEgressFault pwFwdingStandby
Peer Fault Ip     : None
Peer Vccv CV Bits : lspPing bfdFaultDet
Peer Vccv CC Bits : mplsRouterAlertLabel

---snip---
```

```
-----
Number of SDPs : 1
-----
=====
*A:PE-2#
```

In this example, the remote side of the SDP is sending lacIngressFault lacEgressFault pwFwdingStandby flags. This is because the Epipe service on PE-3 is down because the MC-LAG is in standby/down status.

Link and node protection can be tested. The access links are protected by the MC-LAG, the PE routers are protected by the combination of MC-LAG/pseudowire redundancy. The SDPs can be protected by FRR in the case of RSVP-TE or LDP.

Revertive behavior is expected when different MC-LAG port priorities are configured or if the number of MC-LAG ports is different on the MC-LAG peers: convergence takes place when the active PE fails and convergence takes place again when that PE is online again.

In case of revertive behavior, MC-LAG convergence might take less time than the setup of the spoke SDPs, thus creating a temporary blackhole. To avoid this situation, it is best to configure **hold-time up** on the LAG ports. In that case the ports are kept in a down state for a configured period of time after the node has rebooted. This is done to ensure that the SDPs are operationally up when the MC-LAG convergence takes place. The **hold-time up** is expressed in seconds.

```
*A:PE-1# configure port 1/1/3 ethernet hold-time up 50
*A:PE-1# configure port 1/1/4 ethernet hold-time up 50
```

Step 1.11 Inter-Chassis Backup (ICB) pseudowire configuration.

Note that in this setup the configuration of ICBs is optional. It can be used to speed up convergence by forwarding in-flight packets during MC-LAG transition. [Figure 35](#) shows some setup examples where ICBs are required. ICBs cannot be configured at endpoints where the other object is a standard SAP, only MC-LAG SAPs and pseudowires are allowed with ICBs.

Configure ICB SDPs and associate them to endpoints like shown in [Figure 34](#).

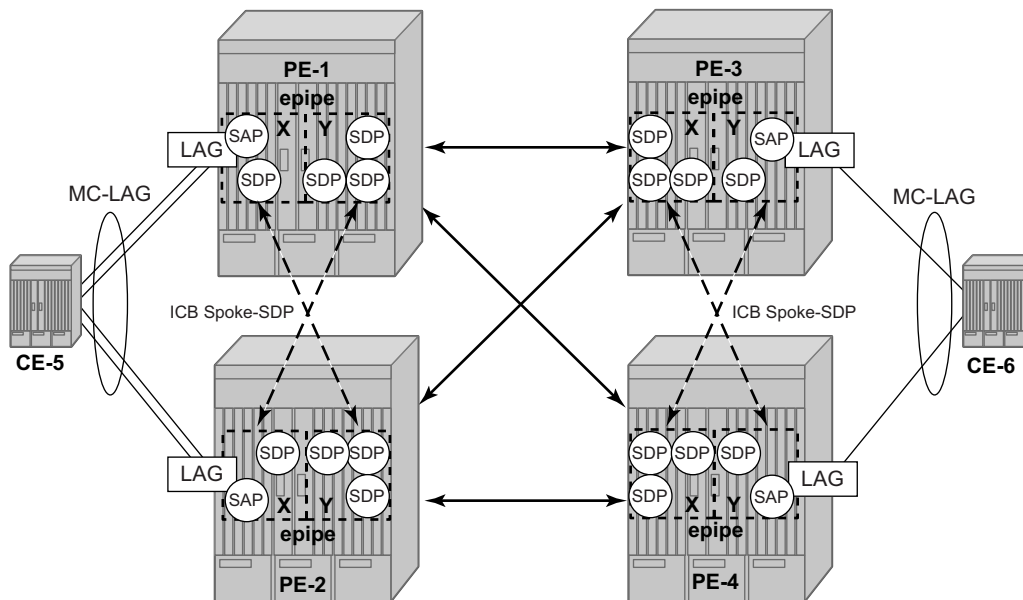


Figure 34: ICB Spoke SDPs and Their Association with the Endpoints

Two ICB spoke SDPs must be configured in the Epipe service on each PE router, one in each endpoint. Different SDP IDs can be used for the ICBs (as opposed to the regular pseudowires) but this is not necessary since the far-end will be the same. The *vc-id* must be different however.

The ICB spoke SDPs must cross, one end should be associated with endpoint **x** and the other end (on the other PE) should be associated with endpoint **y**. Note that after configuring the ICB spoke SDPs the Epipe service will be up/up on all four PE routers.

Only one spoke SDPs will be forwarding. If there is an ICB and a MC-LAG SAP in an endpoint the ICB will only forward if the SAP goes down. If an ICB resides in an endpoint together with other spoke SDPs the ICB will only forward if there is no other active spoke SDP.

The following output shows the additional Epipe service configuration on each PE:

```
*A:PE-1# configure service
  epipe 1
    spoke-sdp 12:1 endpoint "X" icb create
  exit
  spoke-sdp 12:2 endpoint "Y" icb create
  exit

*A:PE-2# configure service
  epipe 1
    spoke-sdp 21:1 endpoint "Y" icb create
  exit
  spoke-sdp 21:2 endpoint "X" icb create
  exit
```



```

exit

*A:PE-3# configure service
  epipe 1
    spoke-sdp 34:1 endpoint "X" icb create
    exit
    spoke-sdp 34:2 endpoint "Y" icb create
    exit

*A:PE-4# configure service
  epipe 1
    spoke-sdp 43:1 endpoint "Y" icb create
    exit
    spoke-sdp 43:2 endpoint "X" icb create
    exit

```

Step 1.12 Verification of active objects for each endpoint.

The following command shows which objects are configured for each endpoint and which is the active object at this moment:

```

*A:PE-1# show service id 1 endpoint
=====
Service 1 endpoints
=====
Endpoint name      : X
Description        : (Not Specified)
Creation Origin    : manual
Revert time        : 0
Act Hold Delay     : 0
Standby Signaling Master : false
Standby Signaling Slave  : false
Tx Active          : lag-1
Tx Active Up Time   : 0d 00:04:27
Revert Time Count Down : N/A
Tx Active Change Count : 1
Last Tx Active Change : 09/03/2015 07:44:51
-----
Members
-----
SAP      : lag-1                               Oper Status: Up
Spoke-sdp: 12:1 Prec:4 (icb)                   Oper Status: Up
=====
Endpoint name      : Y
Description        : (Not Specified)
Creation Origin    : manual
Revert time        : 0
Act Hold Delay     : 0
Standby Signaling Master : false
Standby Signaling Slave  : false
Tx Active (SDP)    : 14:1
Tx Active Up Time   : 0d 00:03:47

```

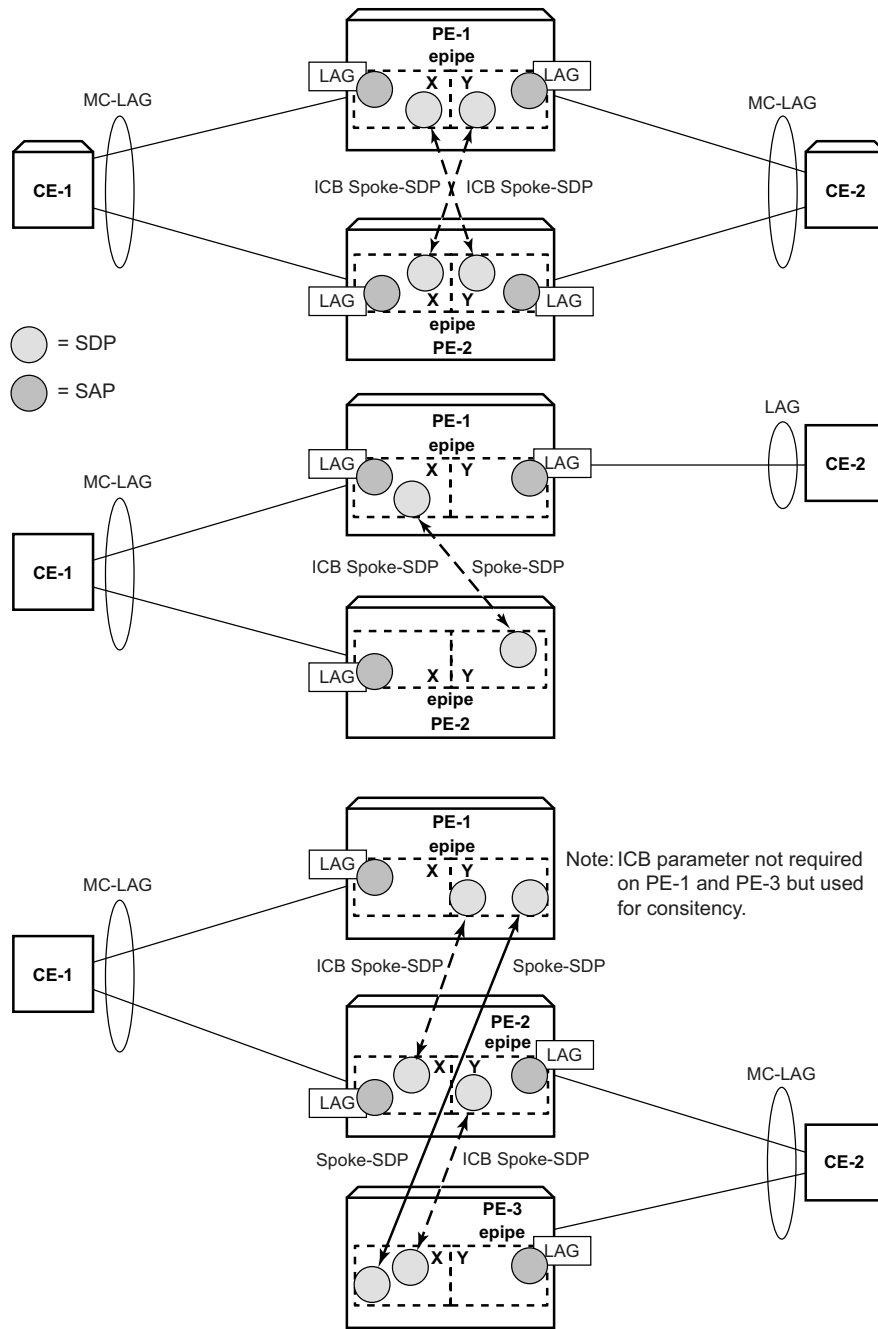
Configuration

```
Revert Time Count Down      : N/A
Tx Active Change Count      : 2
Last Tx Active Change       : 09/03/2015 07:45:31
-----
Members
-----
Spoke-sdp: 12:2 Prec:4 (icb)           Oper Status: Up
Spoke-sdp: 13:1 Prec:4                 Oper Status: Up
Spoke-sdp: 14:1 Prec:4                 Oper Status: Up
=====
=====
*A:PE-1#
```

Note that on PE-1 the SAP and the spoke SDP 14:1 are active. The other objects do not forward traffic.

Step 1.13 Other types of setups.

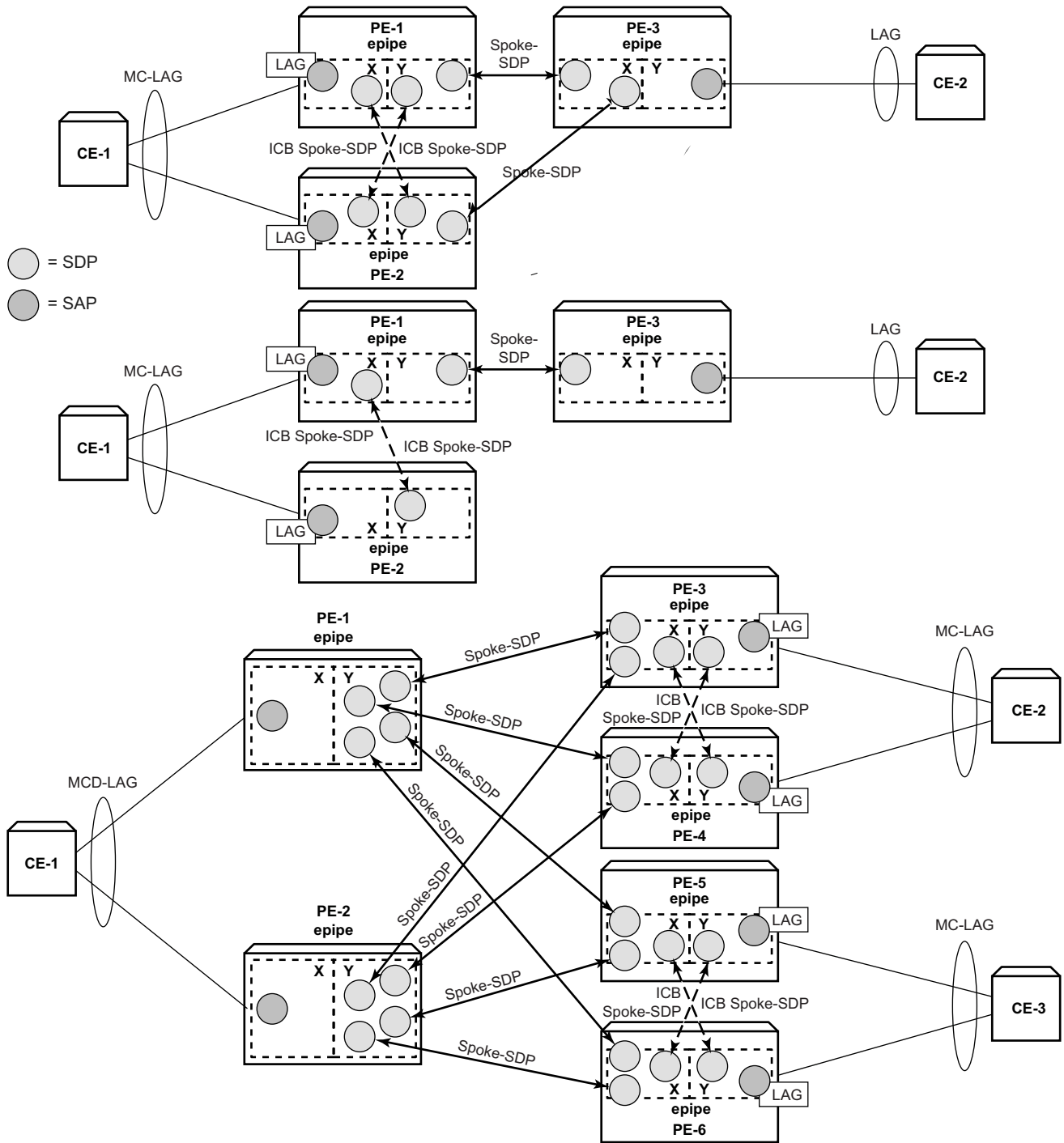
[Figure 35](#) and [Figure 36](#) shows other setups that combine MC-LAG and pseudowire redundancy.



OSSG384

Figure 35: Additional Setup Example 1

Configuration



OSSG386

Figure 36: Additional Setup Example 2

MC-LAG in VPLS Services

MC-LAG can also be configured in VPLS services. When the MC-LAG converges the PE that moves to standby state for the MC-LAG will send out an LDP address withdrawal message to all peers configured in the VPLS service. Both types of SDPs (spoke and mesh) support this feature. The PE peers will then flush all the MAC addresses learned through the PE that sent the LDP MAC address withdrawal message.

Since a VPLS service is a multipoint service, pseudowire redundancy is not required. The MC-LAG redundancy configuration is identical.

Forced Switchover

MC-LAG convergence can be forced with **tools perform lag** command:

```
*A:PE-1# tools perform lag force
- force all-mc {active|standby}
- force lag-id <lag-id> [sub-group <sub-group-id>] {active|standby}
- force peer-mc <peer-ip-address> {active|standby}

<lag-id>           : [1..800]
<sub-group-id>     : [1..16]
<all-mc>           : keyword
<peer-ip-address>  : a.b.c.d
<active|standby>   : keywords

*A:PE-1# tools perform lag force lag-id 1 standby
*A:PE-1# show lag 1
=====
Lag Data
=====
Lag-id      Adm      Opr      Port-Threshold  Up-Link-Count  MC Act/Stdby
-----
1           up       down     0               0              standby
=====
*A:PE-1#
```

After the forced switchover it is important to clear the forced switchover:

```
*A:PE-1# tools perform lag clear-force
- clear-force all-mc
- clear-force lag-id <lag-id> [sub-group <sub-group-id>]
- clear-force peer-mc <ip-address>

<lag-id>           : [1..800]
<sub-group-id>     : [1..16]
<all-mc>           : keyword
<ip-address>       : a.b.c.d

*A:PE-1# tools perform lag clear-force lag-id 1
```

Conclusion

MC-LAG is an Alcatel-Lucent added value redundancy feature that offers fast access link convergence in Epipe and VPLS services for CE devices that support standard LACP. PE node convergence for VPLS services is enhanced by using LDP address withdrawal messages to flush the FDB on the PE peers. PE node convergence for Epipes is guaranteed by using pseudowire redundancy.

Router Configuration

In This Section

This section provides configuration information for the following topics:

- [Aggregate Route Indirect Next-Hop Option on page 189](#)
- [Bi-Directional Forwarding Detection on page 201](#)
- [LFA Policies Using OSPF as IGP on page 249](#)

Aggregate Route Indirect Next-Hop Option

In This Chapter

This section provides information about aggregate route indirect next-hop option configurations.

Topics in this section include:

- [Applicability on page 190](#)
- [Overview on page 191](#)
- [Configuration on page 192](#)
- [Conclusion on page 199](#)

Applicability

This section is applicable to all of the 7950 XRS series, 7750 SR series (SR-7, SR-12, SR-c4 and SR-c12), 7710 SR, as well as 7450 ESS (ESS-6, ESS-7 and ESS-12) series in mixed mode. This configuration is supported by all IOM/IMMs and supported on all chassis types as long as the protocol is supported.

The configuration was tested in release 11.0R1.

Overview

The 7x50s have for many releases supported the ability to configure IPv4 and IPv6 aggregate routes. A configured aggregate route that has the best preference for the prefix is added to the routing table (activated) when it has at least one contributing route and removed when there are no longer any more contributing routes. A contributing route is any route installed in the forwarding table that is a more-specific match of the aggregate. (10.16.12.0/24 is a contributing route to the aggregate route 10.16.12.0/22, but for this same aggregate 10.16.12.0/22 and 10.0.0.0/8 are not contributing routes).

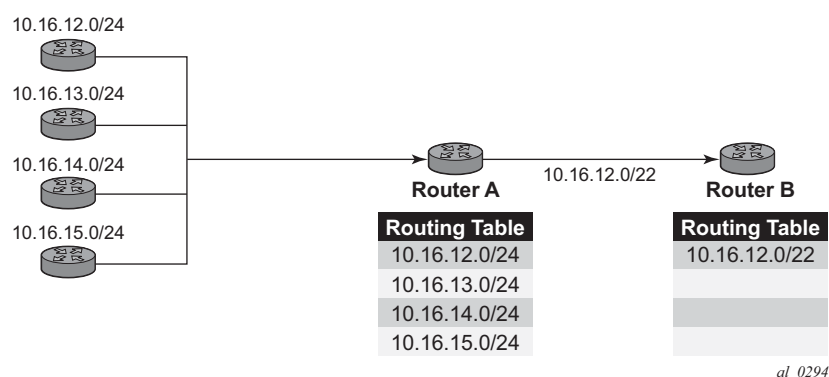


Figure 37: Aggregate Routes

In [Figure 37](#), Router A could choose to advertise all the four routes or one aggregate route. By aggregating the four routes, fewer updates are sent on the link between routers A and B, router B needs to maintain a smaller routing table resulting in better convergence and router B saves on computational resources by evaluating fewer entries in its routing table.

Different network operators have different requirements for how to forward a packet that matches an aggregate route but not any of the more-specific routes in the forwarding table that activated it. In general, there are three different options:

1. The packet can be forwarded according to the next-most specific route, ignoring the aggregate route. This can lead to routing loops in some topologies.
2. The packet can be discarded.
3. The packet can be forwarded towards an indirect next-hop address that is configured by the operator. The indirect next-hop could be the address of a threat management server that analyzes the packets it receives for security threats. This option requires the aggregate route to be installed in the forwarding table with a resolved next-hop interface determined from a route lookup of the indirect next-hop address.

Configuration

The test topology is shown in [Figure 38](#).

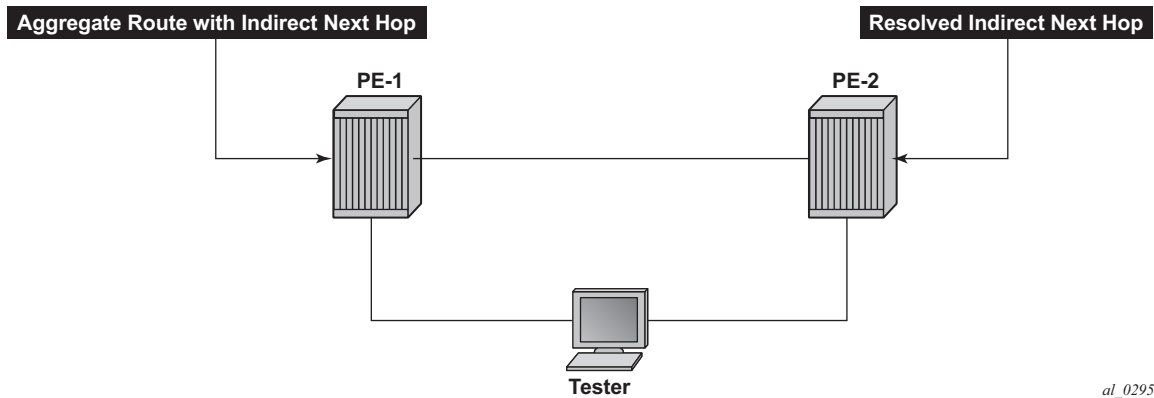


Figure 38: Test topology

This feature adds a new **indirect** keyword and associated IP address parameter to the existing aggregate command in these configuration contexts:

```
config>router
config>service>vprn
```

The aggregate route configuration commands are shown below.

```
config
  router
    aggregate ip-prefix/ip-prefix-length [summary-only] [as-set] [aggregator as
      -number: ip-address] [community comm-id] [black-hole | indirect ip-address]

config
  service
    vprn
    aggregate ip-prefix/ip-prefix-length [summary-only] [as-set] [aggregator as
      -number: ip-address] [community comm-id] [black-hole | indirect ip-address]
```

Parameters:

- **indirect** — This indicates that the aggregate route has an indirect address. Note from this syntax that the indirect option is mutually exclusive with the black-hole option. To change the next-hop type of an aggregate route (for example from black-hole to indirect) the route must be deleted and then re-added with the new next-hop type (however other configuration attributes can generally be changed dynamically).

- `<ip-address>` — Installing an aggregate route with an indirect next-hop is supported for both IPv4 and IPv6 prefixes, however if the aggregate prefix is IPv6 the indirect next-hop must be an IPv6 address and if the aggregate prefix is IPv4 the indirect next-hop must be an IPv4 address.

An indirect next-hop address of an aggregate route may be resolved by any of the following route types:

- Direct/local route
- Static route with regular next-hop, black-hole next-hop or an indirect next-hop
- OSPFv2 or RIP IPv4 route (applicable only to IPv4 aggregate routes)
- LDP shortcut route (applicable only to IPv4 aggregate routes)
- OSPFv2 or IS-IS shortcut route (IPv4 route with an LDP/RSVP or RSVP tunnel next-hop) (applicable only to IPv4 aggregate routes)
- OSPFv3 or IS-IS route
- BGP route resolved by an IGP route
- BGP route resolved by a BGP route
- BGP labeled-IPv4 route resolved by an LDP or RSVP tunnel (applicable only to IPv4 aggregate routes)
- 6PE route resolved by an LDP tunnel or static route with black-hole next-hop (applicable only to IPv6 aggregate routes)
- BGP-VPN route resolved by a BGP labeled-IPv4 route, LDP tunnel, or RSVP tunnel (applicable only to aggregate routes configured in a VPRN context)

If an indirect next-hop is not resolved, the aggregate route will show up as black-hole.

Step 1. Configure the aggregate route.

Command: **aggregate**

Syntax **aggregate** *ip-prefix/ip-prefix-length* [**summary-only**] [**as-set**] [**aggregator** *as-number: ip-address*] [**community** *comm-id*] [**black-hole** | **indirect** *ip-address*]
no aggregate *ip-prefix/ip-prefix-length*

Context config>router
 config>service>vprn

Description Use this command to automatically install an aggregate in the routing table when there are one or more component routes. A component route is any route used for forwarding that is a more-specific match of the aggregate.

The use of aggregate routes can reduce the number of routes that need to be advertised to neighbor routers, leading to smaller routing table sizes.

Overlapping aggregate routes may be configured; in this case a route becomes a component of only the one aggregate route with the longest prefix match. For example if one aggregate is configured as 10.0.0.0/16 and another as 10.0.0.0/24, then route 10.0.128/17 would be aggregated into 10.0.0.0/16, and route 10.0.0.128/25 would be aggregated into 10.0.0.0/24. If multiple entries are made with the same prefix and the same mask the previous entry is overwritten.

A standard 4-byte BGP community may be associated with an aggregate route in order to facilitate route policy matching.

By default aggregate routes are not installed in the forwarding table, however there are configuration options that allow an aggregate route to be installed with a black-hole next hop or with an indirect IP address as next hop.

The **no** form of the command removes the aggregate.

Default No aggregate routes are defined.

Parameters *ip-prefix/ip-prefix-length* — The destination address of the aggregate route.

Values:

ipv4-prefix	a.b.c.d (host bits must be 0)
ipv4-prefix-length	0 — 32
ipv6-prefix	x:x:x:x:x:x:x
	x:x:x:x:x:d.d.d.d
	x: [0 — FFFF]H
	d: [0 — 255]D
ipv6-prefix-length	0 — 128

summary-only — This optional parameter suppresses the advertisement of more specific component routes for the aggregate. To remove the summary-only option, enter the same aggregate command without the summary-only parameter.

as-set — This optional parameter is only applicable to BGP and creates an aggregate where the path advertised for this route will be an AS_SET consisting of all elements contained in all paths that are being summarized. Use this feature carefully as it can increase the amount of route churn due to best path changes.

aggregator as-number:ip-address — This optional parameter adds the BGP aggregator path attribute to the aggregate route. When configuring the aggregator, a two-octet AS number used to form the aggregate route must be entered, followed by the IP address of the BGP system that created the aggregate route.

community comm-id — This configuration option associates a BGP community with the aggregate route. The community can be matched in route policies and is automatically added to BGP routes exported from the aggregate route.

Values: *comm-id* *asn:comm-val | well-known-comm*
asn 0 — 65535
comm-val 0 — 65535
well-known-comm no-advertise, no-export, no-export-subconfed

black-hole — This optional parameter installs the aggregate route, when activated, in the FIB with a black-hole next-hop. Packets matching an aggregate route with a black-hole next hop are discarded.

indirect *ip-address* — This configuration option specifies that the aggregate route should be installed in the FIB with a next-hop taken from the route used to forward packets to *ip-address*.

Values *ipv4-prefix* a.b.c.d
ipv6-prefix x:x:x:x:x:x:x
x:x:x:x:x:d.d.d.d
x: [0 — FFFF]H
d: [0 — 255]D

```
A:PE-1>config#router aggregate 10.10.10.0/24 community 64496:64497 indirect 192.168.11.11
```

```
A:PE-1# show router aggregate
```

```
=====
Aggregates (Router: Base)
=====
```

Prefix	Aggr IP-Address	Aggr AS
Summary	AS Set	State
NextHop	Community	NextHopType
10.10.10.0/24	0.0.0.0	0
False	False	Inactive
192.168.11.11	64496:64497	Indirect

```
-----
No. of Aggregates: 1
=====
```

Step 2. Configure the contributing routes to activate the aggregate route.

```
*A:PE-1# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type    Proto    Age          Pref
  Next Hop[Interface Name]                        Metric
-----
10.10.10.0/24                                     Remote  Aggr      00h00m20s  130
    Black Hole                                     0
10.10.10.3/32                                     Remote  Static    00h00m24s   5
    10.18.0.89                                     1
10.10.10.4/32                                     Remote  Static    00h00m24s   5
    10.18.0.89                                     1
10.10.10.5/32                                     Remote  Static    00h00m24s   5
    10.18.0.89                                     1
10.10.10.8/32                                     Remote  Static    00h00m24s   5
    10.18.0.89                                     1
10.12.0.0/24                                     Local   Local     00h00m24s   0
    to_PE-2                                         0
10.18.0.0/24                                     Local   Local     00h00m24s   0
    to_Routers                                     0
10.19.0.0/24                                     Local   Local     00h00m24s   0
    to_Tester                                      0
10.20.1.1/32                                     Local   Local     00h00m24s   0
    system                                          0
-----
No. of Routes: 9
Flags: L = LFA nexthop available    B = BGP backup route available
      n = Number of times nexthop is repeated
=====
```

The aggregate route is now active:

```
A:PE-1# show router aggregate
=====
Aggregates (Router: Base)
=====
Prefix                                Aggr IP-Address  Aggr AS
  Summary                            AS Set          State
  NextHop                           Community       NextHopType
-----
10.10.10.0/24                        0.0.0.0         0
  False                             False           Active
  192.168.11.11                     64496:64497     Indirect
-----
No. of Aggregates: 1
=====
```


Step 3. Configure the resolving route.

```
A:PE-2# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type   Proto   Age      Pref
  Next Hop[Interface Name]                        Metric
-----
10.10.10.0/24                                     Remote Static 00h00m14s 5
      10.29.0.99                                     1
10.12.0.0/24                                     Local  Local  00h00m14s 0
      to_PE-1                                         0
10.20.1.2/32                                     Local  Local  00h00m14s 0
      system                                         0
10.29.0.0/24                                     Local  Local  00h00m14s 0
      to_Tester                                       0
192.168.11.0/24                                   Remote Static 00h00m14s 5
      10.12.0.2                                     1
-----
No. of Routes: 5
Flags: L = LFA nexthop available    B = BGP backup route available
      n = Number of times nexthop is repeated
=====
```

The aggregate route now has the resolved next-hop.

```
*A:PE-2# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type   Proto   Age      Pref
  Next Hop[Interface Name]                        Metric
-----
10.10.10.0/24                                     Remote Aggr 00h00m20s 130
      10.12.0.2                                     0
10.10.10.3/32                                     Remote Static 00h00m24s 5
      10.18.0.89                                     1
10.10.10.4/32                                     Remote Static 00h00m24s 5
      10.18.0.89                                     1
10.10.10.5/32                                     Remote Static 00h00m24s 5
      10.18.0.89                                     1
10.10.10.8/32                                     Remote Static 00h00m24s 5
      10.18.0.89                                     1
10.12.0.0/24                                     Local  Local  00h00m24s 0
      to_PE-2                                         0
10.18.0.0/24                                     Local  Local  00h00m24s 0
      to_Routers                                       0
10.19.0.0/24                                     Local  Local  00h00m24s 0
      to_Tester                                       0
10.20.1.1/32                                     Local  Local  00h00m24s 0
      system                                         0
192.168.11.0/24                                   Remote Static 00h00m24s 5
      10.12.0.2                                     1
-----
No. of Routes: 10
```

Configuration

Flags: L = LFA nexthop available B = BGP backup route available
 n = Number of times nexthop is repeated

=====

Conclusion

Aggregate routes offer several advantages, the key being reduction in the routing table size and overcoming routing loops, among other things. Aggregate routes with indirect next hop option helps in faster network convergence by decreasing the number of route table changes. This example shows how to configure aggregate routes with indirect next hop option.

Bi-Directional Forwarding Detection

In This Chapter

This section provides information about bi-directional forwarding (BFD) detection.

Topics in this section include:

- [Applicability on page 202](#)
- [Overview on page 203](#)
- [Configuration on page 205](#)
- [Conclusion on page 248](#)

Applicability

This section is applicable to all of the 7x50 and 7710 series but the timing differs among platforms and these will be indicated. Note that the centralized cpm-np type is only supported by 7750/7450s equipped with SF/CPM 2 or higher. The information contained in this section has been tested with Release 8.0.R4.

Overview

Bi-Directional Forwarding Detection (BFD) is a light-weight protocol which provides rapid path failure detection between two systems. It has been recently published as a series of RFCs (RFC 5880, *Bidirectional Forwarding Detection (BFD)*, to RFC 5884, *Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)*).

If a system running BFD stops receiving BFD messages on an interface, it will determine that there has been a failure in the path and notifies other protocols associated with the interface. BFD is useful in situations where two nodes are interconnected through either an optical (DWDM) or Ethernet network. In both cases, the physical network has numerous extra hops which are not part of the Layer 3 network and therefore, the Layer 3 nodes are incapable of detecting failures which occur in the physical network on spans to which the Layer 3 devices are not directly connected.

BFD protocol provides rapid link continuity checking between network devices, and the state of BFD can be propagated to IP routing protocols to drastically reduce convergence time in cases where a physical network error occurs in a transport network.

RFC 5880 define two modes of operation for BFD:

- Asynchronous mode (supported by ALU routers covered in this section) — Uses periodic BFD control messages to test the path between systems
- Demand mode (not supported by ALU router covered in this section)

In addition to the two operational modes, an echo function is defined (ALU routers covered by this section only support response, looping back received BFD messages to the original sender).

The goal of this section is to describe the configuration and troubleshooting for BFD on a link between two peers in the following scenarios:

- BFD for ISIS
- BFD for OSPF
- BFD for PIM
- BFD for Static route
- BFD IES
- BFD for RSVP
- BFD for T-LDP
- BFD support of OSPF CE-PE adjacencies
- BFD over IPSec tunnel
- BFD over VRRP

Figure 39 provides an overview of the possible BFD implementations and shows all protocols that can be bound to a BFD session.

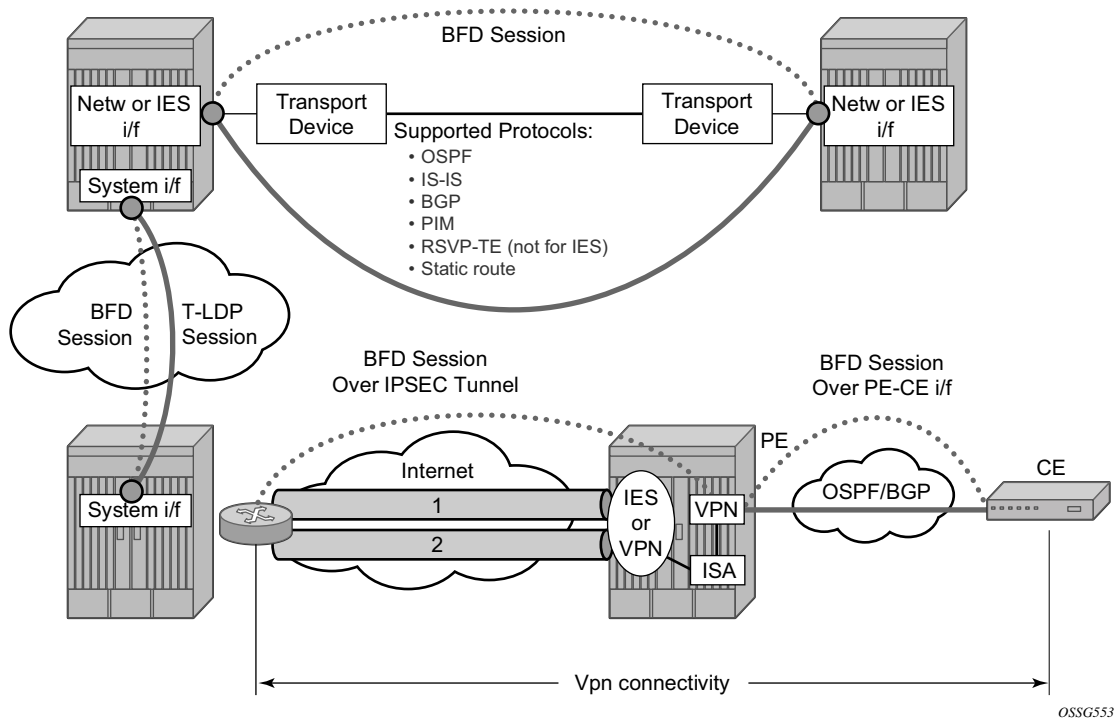


Figure 39: BFD Multi-Scenarios

Configuration

BFD packets are processed both locally (processed on IOM CPU) and centrally (processed on the CPM).

Starting with Release 8, the CPM is able to centrally generate the BFD packets at a sub second interval as low as 10 msec. However it should be noted that the BFD state machine is still implemented in software. It is the BFD packet generation that can be now selectively delegated to CPM hardware as needed. This is applicable where sub second operational requirements for BFD or scaling the number of BFD sessions beyond 250 are required.

Centralized sessions are processed:

- in software by 7x50 SR-1 and ESS-1, 7710 c4 and c12 and 7x50 equipped with SF/CPM 1.
- in hardware by 7x50 equipped with SF/CPM 2 or higher.

Minimum transmitting and receiving Intervals are as follows:

- Centralized sessions:
 - Minimum 300 ms in 7x50 SR-1 and ESS-1, 7710 c4 and c12
 - Minimum 100 ms in 7x50 equipped with SF/CPM 1 and in every 7x50 up to Release 7.0
 - Minimum 10 ms in 7x50 equipped with SF/CPM 2 or higher
- Local sessions:
 - Minimum 100 ms

The following applications require BFD to run centrally on the SF/CPM and a centralized session will be created independently of the type explicitly declared by the user:

- BFD for IES/VP RN over Spoke SDP
- BFD over LAG and VSM Interfaces
- Protocol associations using loopback and system interfaces (e.g. BFD for T-LDP)
- BFD over IPSec sessions
- BFD sessions associated with multi-hop peering

Figure 40 shows the most relevant scenarios where BFD centralized sessions are used.

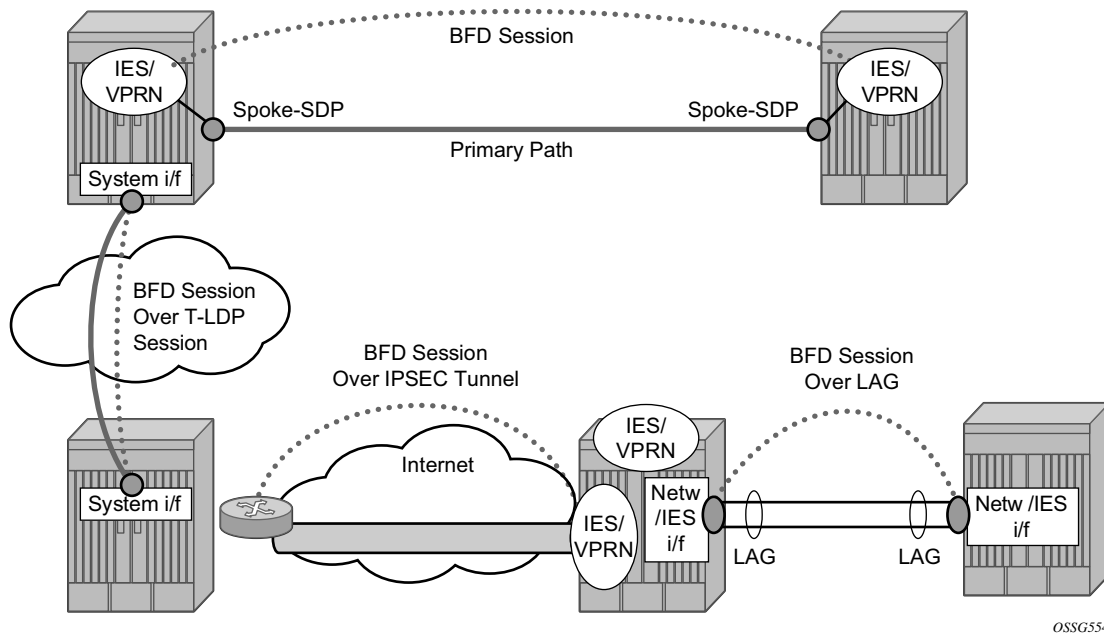


Figure 40: BFD Centralized Sessions

On the other end, when the two peers are directly connected, the BFD session is local by default, but in a 7x50 equipped with SF/CPM 2 or higher, the user can choose which session (local or centralized) to implement.

As general rule, the following steps are required to configure and enable a BFD session when peers are directly connected:

1. Configure BFD parameters on the peering interfaces.
2. Check that the Layer 3 protocol, that is to be bound to BFD, is up and running.
3. Enable BFD under the Layer 3 protocol interface.

Since most of the following procedures share the same first step, it is described only once in the next paragraph and then referred to in the following paragraphs.

BFD Base Parameter Configuration and Troubleshooting

The reference topology for the generic configuration of BFD over two local peers is shown in [Figure 41](#).

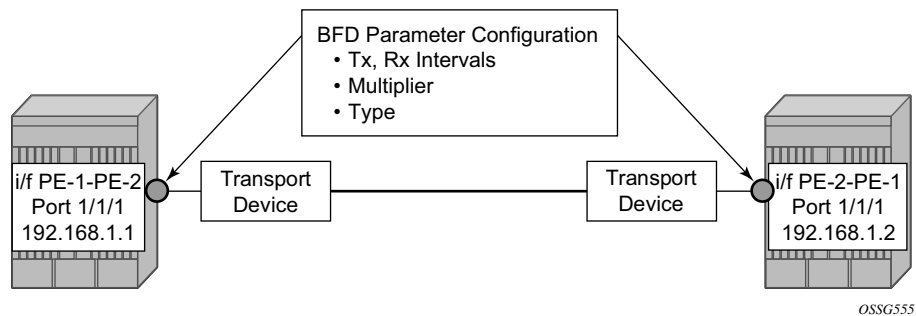


Figure 41: BFD Interface Configuration

To configure BFD between two peers, the user should firstly enable base level BFD on interfaces between PE-1 and PE-2.

On PE1:

```
configure
  router
    interface PE-1-PE-2
      address 192.168.1.1/30
      port 1/1/1
      bfd 100 receive 100 multiplier 3
    exit
  exit
exit
```

On PE2:

```
configure
  router
    interface PE-2-PE-1
      address 192.168.1.2/30
      port 1/1/1
      bfd 100 receive 100 multiplier 3
    exit
  exit
exit
```

BFD Base Parameter Configuration and Troubleshooting

The following **show** commands are used to verify the BFD configuration on the router interfaces on PE1 and PE2.

On PE1:

```
A:PE1# show router bfd interface
=====
BFD Interface
=====
Interface name          Tx Interval    Rx Interval    Multiplier
-----
PE-1-PE-2              100           100           3
-----
No. of BFD Interfaces: 1
=====
A:PE1#
```

On PE2:

```
A:PE2# show router bfd interface
=====
BFD Interface
=====
Interface name          Tx Interval    Rx Interval    Multiplier
-----
PE-2-PE-1              100           100           3
-----
No. of BFD Interfaces: 1
=====
A:PE2#
```

Note that, BFD being an asynchronous protocol, it is possible to configure different tx and rx intervals on the two peers. This is because BFD rx/tx interval values are signaled in the BFD packets while establishing the BFD session.

In 7x50s equipped with SF/CPM 2 or higher, configurable BFD parameters are as follows:

```
bfd <transmit-interval> [receive <receive-interval>] [multiplier <multiplier>] [echo-
receive <echo-interval>] [type <cpm-np>]
no bfd

<transmit-interval>    : [10..100000] in milliseconds
<receive-interval>     : [10..100000] in milliseconds
<multiplier>           : [3..20]
<echo-interval>        : [100..100000] in milliseconds
<cpm-np>               : keyword - use CPM network processor
```

Note that it is possible to force the BFD session to be centrally managed by the CPM hardware.

As regards the echo function, it is possible to set the minimum echo receive interval, in milliseconds, for the BFD session. The default value is 100 ms.

If a BFD session is running, it is possible to modify its parameters but to change its type the session must be previously shut down manually. Note that this causes the upper layer protocols bound to it to be brought down as well.

```
configure
router
  interface PE-2-PE-1
    bfd 10 receive 10 multiplier 3 type cpm-np
  exit
exit
exit
```

Forcing a centralized session in the case of directly connected peers can be useful when:

- Lower Tx and Rx intervals are requested (up to 10 ms instead of 100 ms supported by local sessions)
- There are no more available local sessions
- Max limit of 500 packet per second per IOM has been reached

The instructions illustrated in following paragraphs are required to complete the configuration and enable BFD.

The BFD session should come up. To verify it, execute a **show router bfd session** command (bound to OSPF in the following example).

```
A:PE1# show router bfd session
=====
BFD Session
=====
Interface                               State           Tx Intvl  Rx Intvl  Multipl
  Remote Address                       Protocol        Tx Pkts   Rx Pkts   Type
-----
PE-1-PE-2                               Up (3)          100       100       3
  192.168.1.2                           ospf            165       174       iom
-----
No. of BFD sessions: 1
=====
A:PE1#
```

If the command gives a negative output, troubleshoot it by firstly checking that the protocol that is bound to it is up: for instance, check the OSPF neighbor adjacency as shown in following example.

```
A:PE-1# show router ospf neighbor
=====
OSPF Neighbors
=====
Interface-Name                       Rtr Id          State        Pri  RetxQ  TTL
-----
PE-1-PE-2                           192.0.2.1       Full         1    0      34
...
=====
A:PE-1#
```

Then check whether a BFD resource limit has been reached (maximum number of local/centralized sessions or maximum number of packet per second per IOM).

If the overloaded limit is the maximum supported number of sessions, the cause is shown by log-id 99. In the reported example, the maximum number of sessions per slot has been reached.

```
A:PE-2# show log log-id 99
=====
Event Log 99
=====
Description : Default System Log
Memory Log contents [size=500 next event=7845 (wrapped)]

7844 2010/10/02 16:43:30.21 UTC MINOR: VRTR #2020 Base 192.168.1.1
BFD Session on node 192.168.1.1 has been deleted.

7843 2010/10/02 16:43:30.21 UTC MAJOR: VRTR #2013 Base Max supported sessions reached
The number of BFD sessions on slot 1 has exceeded 250, constrained by maxSessionsPerSlot"
```

In this case, when one of the running sessions is manually removed or goes down, then the additional configured session will come up. If the limit reached is local (on IOM) it is possible to bring up the session by re-configuring it as centralized, by changing the type.

To check if IOM CPU is able to start more local BFD sessions, execute a **show router BFD session summary** command.

```
A:PE2# show router bfd session summary
=====
BFD Session Summary
=====
Termination      Session Count
-----
central          0
cpm-np           1
iom, slot 1      250
iom, slot 2      0
iom, slot 3      0
iom, slot 4      0
iom, slot 5      0
Total            251
=====
```

If the **show router bfd session** command reports that the BFD session is down, then check the BFD peer's configuration and state.

The following **log 99** output reports PE-1 logs after a misconfiguration of PE-2 (disabling BFD on the OSPF interface).

As soon as BFD is shutdown on the OSPF interface PE-2-PE-1 of PE-2, the BFD session in PE-1 goes to the down state, then the OSPF adjacency is brought down for approximately 2.8 secs and finally the OSPF state goes back to full, while the BFD session stays in down state.

This state will last until BFD is re-enabled on PE-2 interface.

```
A:PE-1# show log log-id 99
=====
Event Log 99
=====
Description : Default System Log
Memory Log contents [size=500 next event=7 (not wrapped)]

6 2010/10/02 08:47:35.91 UTC WARNING: OSPF #2002 Base VR: 1 OSPFv2 (0)
LCL_RTR_ID 192.0.2.1: Neighbor 192.0.2.2 on PE-1-PE-2 router state changed to full (event
EXC_DONE)

5 2010/10/02 08:47:35.91 UTC MINOR: VRTR #2021 Base 192.168.1.2
BFD: The protocols using BFD session on node 192.168.1.2 have changed.

4 2010/10/02 08:47:33.10 UTC WARNING: OSPF #2002 Base VR: 1 OSPFv2 (0)
LCL_RTR_ID 192.0.2.1: Neighbor 192.0.2.2 on PE-1-PE-2 router state changed to down (event
BFD_DOWN)

3 2010/10/02 08:47:33.10 UTC MINOR: VRTR #2021 Base 192.168.1.2
BFD: The protocols using BFD session on node 192.168.1.2 have changed.
```

BFD Base Parameter Configuration and Troubleshooting

```
2 2010/10/02 08:47:33.10 UTC MAJOR: VRTR #2012 Base 192.168.1.2
BFD: Local Discriminator 4009 BFD session to node 192.168.1.2 is down due to noHeartBeat
```

```
A:PE-1# show router bfd session
```

```
=====
BFD Session
=====
Interface          State          Tx Intvl  Rx Intvl  Multipl
  Remote Address    Protocols      Tx Pkts   Rx Pkts   Type
-----
PE-1-PE-2          Down (1)       100       100       3
  192.168.1.2      ospf2          10        0        iom
```

The 2nd column reports the current BFD session state. Possible values are:

- 0 — AdminDown
- 1 — Down
- 2 — Init
- 3 — Up

The **show router bfd session src <ip-address> detail** command can help in debugging the BFD session.

```
A:PE-1# show router bfd session src 192.168.1.1 detail
```

```
=====
BFD Session
=====
Remote Address : 192.168.1.2
Admin State    : Up                               Oper State     : Up (3)
Protocols      : ospf2 pim isis static
Rx Interval    : 100                               Tx Interval    : 100
Multiplier    : 3                                 Echo Interval  : 0
Recd Msgs      : 24046                             Sent Msgs      : 25723
Up Time        : 0d 00:40:05                       Up Transitions : 1
Down Time      : None                               Down Transitions : 0
Version Mismatch : 0

Forwarding Information

Local Discr    : 4002                               Local State    : Up (3)
Local Diag     : 0 (None)                           Local Mode     : Async
Local Min Tx   : 100                                 Local Mult     : 3
Last Sent      : 10/08/2010 20:30:27                 Local Min Rx   : 100
Type           : iom
Remote Discr    : 4001                               Remote State    : Up (3)
Remote Diag     : 0 (None)                           Remote Mode     : Async
Remote Min Tx   : 100                                 Remote Mult     : 3
Last Recv       : 10/08/2010 20:30:27                 Remote Min Rx   : 100
=====
```


BFD for IS-IS

The goal of this section is to configure BFD on a network interlink between two 7750 nodes that are IS-IS peers. The topology referred to in this paragraph is shown in [Figure 42](#).

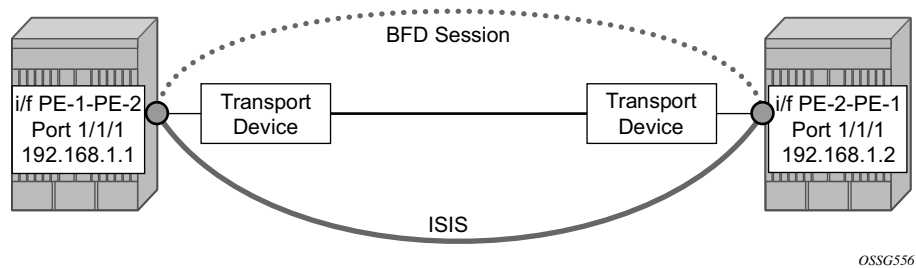


Figure 42: BFD for IS-IS

For the base BFD configuration, please refer to [BFD Base Parameter Configuration and Troubleshooting on page 207](#).

Apply BFD on the IS-IS Interfaces.

On PE1:

```
configure
router
isis
interface PE-1-PE-2
bfd-enable ipv4
exit
exit
exit
exit
```

On PE2:

```
configure
router
isis
interface PE-2-PE-1
bfd-enable ipv4
exit
exit
exit
exit
```

Finally, verify that the BFD session is operational between PE1 and PE2.

On PE1:

```
A:PE1# show router bfd session
=====
BFD Session
=====
Interface          State          Tx Intvl  Rx Intvl  Multipl
Remote Address      Protocol      Tx Pkts   Rx Pkts   Type
-----
PE-1-PE-2          Up (3)        100       100       3
192.168.1.2        isis          165       174       iom
-----
No. of BFD sessions: 1
=====
A:PE1#
```

On PE2:

```
A:PE2# show router bfd session
=====
BFD Session
=====
Interface          State          Tx Intvl  Rx Intvl  Multipl
Remote Address      Protocol      Tx Pkts   Rx Pkts   Type
-----
PE-2-PE-1          Up (3)        100       100       3
192.168.1.1        isis          496       487       iom
-----
No. of BFD sessions: 1
=====
A:PE2#
```

BFD for OSPF

The goal of this section is to configure BFD on a network interlink between two 7750 nodes that are OSPF peers.

For this scenario, the topology is shown in [Figure 43](#).

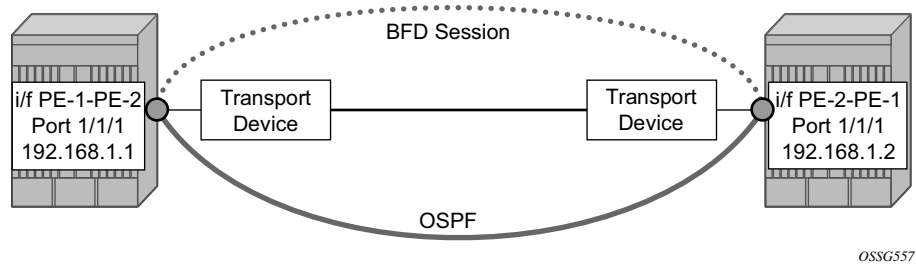


Figure 43: BFD for OSPF

For the base BFD configuration, refer to [BFD Base Parameter Configuration and Troubleshooting on page 207](#).

Apply BFD on the OSPF Interfaces.

On PE1:

```
configure
router
  ospf
    interface PE-1-PE-2
      bfd-enable
    exit
  exit
exit
```

On PE2:

```
configure
router
  ospf
    interface PE-2-PE-1
      bfd-enable
    exit
  exit
exit
```

Verify that the BFD session is operational between PE1 and PE2.

On PE1:

```
A:PE1# show router bfd session
=====
BFD Session
=====
```

Interface	State	Tx Intvl	Rx Intvl	Multipl
Remote Address	Protocol	Tx Pkts	Rx Pkts	Type
PE-1-PE-2	Up (3)	100	100	3
192.168.1.2	ospf	170	179	iom

```
-----
No. of BFD sessions: 1
=====
A:PE1#
```

On PE2:

```
A:PE2# show router bfd session
=====
BFD Session
=====
```

Interface	State	Tx Intvl	Rx Intvl	Multipl
Remote Address	Protocol	Tx Pkts	Rx Pkts	Type
PE-2-PE-1	Up (3)	100	100	3
192.168.1.1	ospf	501	492	iom

```
-----
No. of BFD sessions: 1
=====
A:PE2#
```

BFD for PIM

Since the implementation of PIM uses an Interior Gateway Protocol (IGP) in order to determine its Reverse Path Forwarding (RPF) tree, BFD configuration to support PIM will require BFD configuration of both the IGP protocol and the PIM protocol. Let's assume that IGP protocol is OSPF and that the starting configuration is as described in the previous section.

In this paragraph, configure and enable BFD for PIM on the same interfaces that were previously configured with BFD for OSPF, in reference to the topology shown in [Figure 44](#).

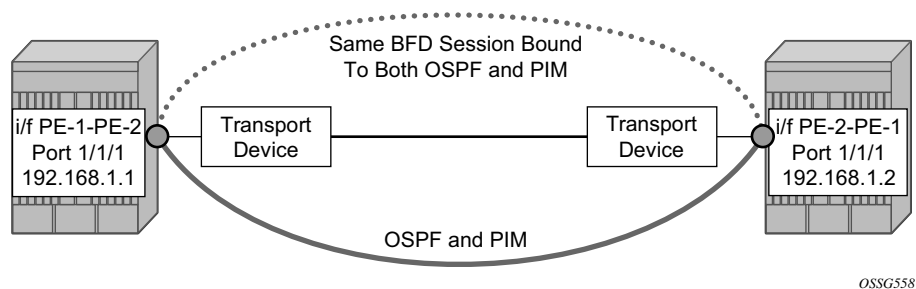


Figure 44: BFD for OSPF and PIM

Since BFD has been already configured on the router interfaces, let's start by applying BFD on the PIM Interface.

On PE1:

```
configure
router
  pim
    interface PE-1-PE-2
      bfd-enable
    exit
  exit
exit
exit
```

On PE2:

```
configure
router
  pim
```

BFD for PIM

```
        interface PE-2-PE-1
        bfd-enable
    exit
    exit
exit
exit
```

The final step is to verify whether the BFD Session is operational between PE1 and PE2 for PIM.

On PE1:

```
A:PE1# show router bfd session
=====
BFD Session
=====
Interface                State          Tx Intvl  Rx Intvl  Multipl
  Remote Address          Protocol      Tx Pkts   Rx Pkts   Type
-----
PE-1-PE-2                Up (3)        100       100       3
  192.168.1.2             ospf2 pim     3874      3845      iom
-----
No. of BFD sessions: 1
=====
A:PE1#
```

On PE2:

```
A:PE2# show router bfd session
=====
BFD Session
=====
Interface                State          Tx Intvl  Rx Intvl  Multipl
  Remote Address          Protocol      Tx Pkts   Rx Pkts   Type
-----
PE-1-PE-2                Up (3)        100       100       3
  192.168.1.1             ospf2 pim     3137      3145      iom
-----
No. of BFD sessions: 1
=====
A:PE2#
```

BFD for Static Routes

The following procedures will go through the necessary steps to configure the base level BFD configuration and then apply BFD to the static routes between PE1 and PE2, referring to topology shown in [Figure 45](#).

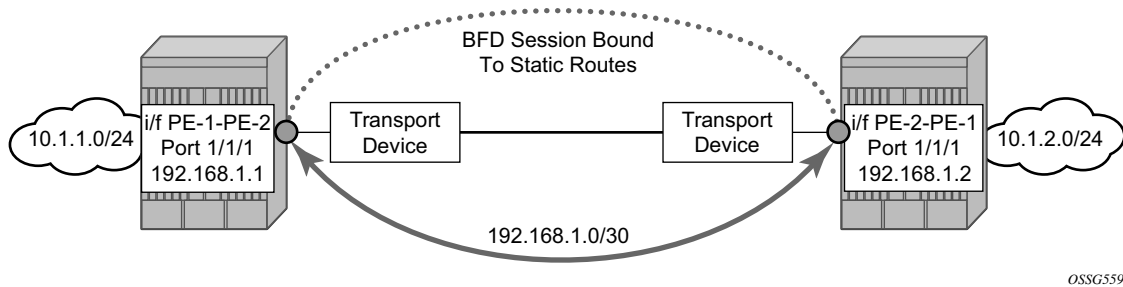


Figure 45: BFD for Static Routes

First, create the static routes for the remote networks both in PE-1 and PE-2.

On PE1:

```
configure
router
  static-route 10.1.2.0/24 next-hop 192.168.1.2
exit
exit
```

On PE2:

```
configure
router
  static-route 10.1.1.0/24 next-hop 192.168.1.1
exit
exit
```

Next, verify that static routes are populated in the routing table.

On PE1:

```
A:PE1# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix                                Type   Proto   Age           Pref
  Next Hop[Interface Name]                Metric
-----
10.1.2.0/24                                Remote Static  00h20m55s     5
  192.168.1.2                               1
```

On PE2:

```
A:PE2# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix                                Type   Proto   Age           Pref
  Next Hop[Interface Name]                Metric
-----
10.1.1.0/24                                Remote Static  00h21m15s     5
  192.168.1.1                               1
```

The next step is to configure the base level BFD on PE1 and PE2.

Refer to paragraph [BFD Base Parameter Configuration and Troubleshooting on page 207](#).

Then apply BFD to the static routing entries using the BFD interfaces as next-hop.

On PE1:

```
configure
router
    static-route 10.1.2.0/24 next-hop 192.168.1.2 bfd-enable
exit
exit
```

On PE2:

```
configure
router
    static-route 10.1.1.0/24 next-hop 192.168.1.1 bfd-enable
exit
exit
```

Note that BFD cannot be enabled if the next hop is indirect or the **blackhole** keyword is specified.

Finally, show the BFD session status.

On PE1:

```
A:PE1# show router bfd session
=====
BFD Session
=====
Interface          State          Tx Intvl  Rx Intvl  Multipl
Remote Address     Protocol      Tx Pkts   Rx Pkts   Type
-----
PE-1-PE-2          Up (3)         100       100       3
192.168.1.2        static         699       661       iom
-----
No. of BFD sessions: 1
=====
```

On PE2:

```
A:PE2# show router bfd session
=====
BFD Session
=====
Interface          State          Tx Intvl  Rx Intvl  Multipl
Remote Address     Protocol      Tx Pkts   Rx Pkts   Type
-----
PE-2-PE-1          Up (3)         100       100       3
192.168.1.1        static         691       729       iom
-----
No. of BFD sessions: 1
=====
```

BFD for IES

The goal of this section is to configure BFD for one IES service over a spoke SDP.

The IES service is configured in both 7750 nodes, PE1 and PE2, and their interfaces are connected by spoke SDP's. The topology is shown in [Figure 46](#).

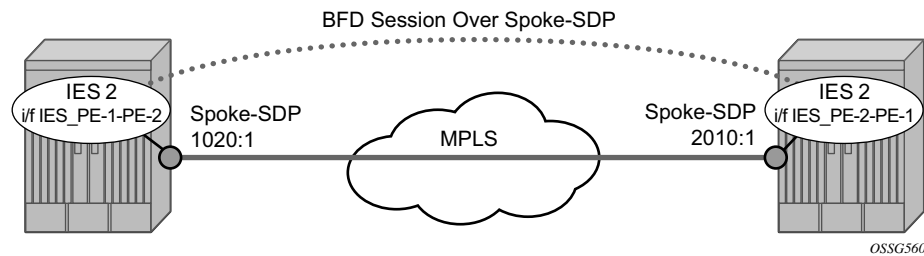


Figure 46: BFD for IES over Spoke SDP

Note that in this scenario BFD is run between the IES interfaces independent of the SDP/LSP paths.

The first step is to configure the IES service on both nodes.

On PE-1:

```
configure
  service
    ies 2 customer 1 create
    interface IES_PE-1-PE-2 create
    address 192.168.3.1/30
    spoke-sdp 1020:1 create
    exit
  exit
  no shutdown
  exit
exit
exit
```

On PE-2:

```
configure
  service
    ies 2 customer 1 create
    interface IES_PE-2-PE-1 create
```

```

        address 192.168.3.2/30
        spoke-sdp 2010:1 create
        exit
    exit
    no shutdown
    exit
exit
exit

```

The next step is to add the IES interfaces to the OSPF area domain.

```

On PE-1:
configure
router
ospf
traffic-engineering
area 0.0.0.0
interface IES-PE-1-PE-2
exit
exit
exit
exit
exit

```

On PE-2:

```

configure
router
ospf
traffic-engineering
area 0.0.0.0
interface IES-PE-2-PE-1
exit
exit
exit
exit
exit

```

Then verify that OSPF and the services are up using show commands on both routers.

On PE-1:

```
A:PE-1# show service id 1 base
=====
Service Basic Information
=====
Service Id       : 2                Vpn Id           : 0
Service Type     : IES
Customer Id      : 1
Last Status Change: 09/30/2010 08:09:22
Last Mgmt Change  : 09/30/2010 08:08:31
Admin State      : Up               Oper State        : Up
SAP Count        : 0
...
=====
A:PE-1#
```

```
A:PE-1# show router ospf neighbor
=====
OSPF Neighbors
=====
Interface-Name      Rtr Id           State      Pri  RetxQ  TTL
-----
IES-PE-1-PE-2       192.0.2.2       Full       1    0      34
-----
=====
A:PE-1#
```

On PE-2:

```
A:PE-2# show service id 2 base
=====
Service Basic Information
=====
Service Id       : 2                Vpn Id           : 0
Service Type     : IES
Customer Id      : 1
Last Status Change: 09/30/2010 08:16:50
Last Mgmt Change  : 09/30/2010 08:16:50
Admin State      : Up               Oper State        : Up
SAP Count        : 0
...
=====
A:PE-2#
```

```
A:PE-2# show router ospf neighbor
=====
OSPF Neighbors
=====
Interface-Name      Rtr Id           State      Pri  RetxQ  TTL
-----
IES-PE-2-PE-1       192.0.2.1       Full       1    0      33
-----
...
=====
A:PE-2#
```

Then configure BFD on the IES interfaces.

On PE-1:

```
configure service ies 2
    interface IES-PE-1-PE-2
        bfd 100 receive 100 multiplier 3
    exit
no shutdown
exit
```

On PE-2:

```
configure service ies 2
    interface IES-PE-2-PE-1
        bfd 100 receive 100 multiplier 3
    exit
no shutdown
exit
```

Finally, enable BFD on the interfaces under OSPF area 0.

On PE-1:

```
A:PE-1# configure router ospf area 0.0.0.0 interface IES-PE-1-PE-2 bfd-enable
```

On PE-2:

```
A:PE-2# configure router ospf area 0.0.0.0 interface IES-PE-2-PE-1 bfd-enable
```

Note that in case of BFD over spoke SDP, a centralized BFD session is created even if a physical link exists between the two nodes. In fact, the next output shows that BFD session type is cpm-np. This is because the spoke SDP is terminated at the CPM. This is also true for BFD running over LAG bundles.

The cpm-np type only exists in 7x50 SR/ESS systems equipped with SF/CPM 2 or higher. When other network elements run centralized BFD sessions like this one, the BFD type is shown as **central**.

```
A:PE-1# show router bfd session
=====
BFD Session
=====
```

Interface	State	Tx Intvl	Rx Intvl	Multipl
Remote Address	Protocols	Tx Pkts	Rx Pkts	Type
IES-PE-1-PE-2	Up (3)	100	100	3
192.168.3.2	ospf2	N/A	N/A	cpm-np

```
-----
No. of BFD sessions: 1
```

BFD for IES

```
A:PE-2# show router bfd session
```

```
=====
```

```
BFD Session
```

```
=====
```

Interface	State	Tx Intvl	Rx Intvl	Multipl
Remote Address	Protocols	Tx Pkts	Rx Pkts	Type
IES-PE-2-PE-1	Up (3)	100	100	3
192.168.3.1	ospf2	N/A	N/A	cpm-np

```
-----
```

```
No. of BFD sessions: 1
```

Note that in the case of centralized BFD sessions, transmitted and received packet counters are not shown.

BFD for RSVP

The goal of this section is to configure BFD between two RSVP interfaces configured in two 7750 nodes.

For this scenario, the topology is shown in [Figure 47](#).

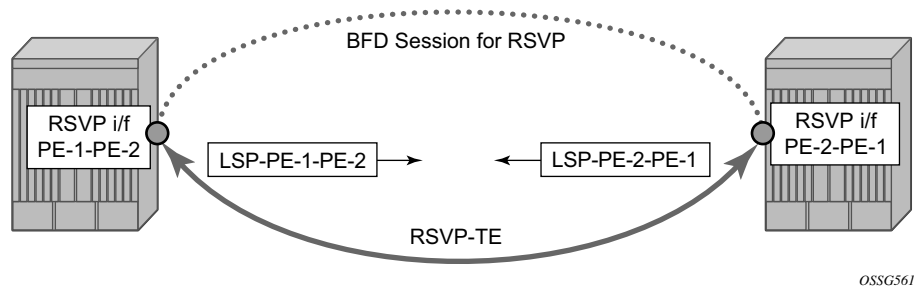


Figure 47: BFD for RSVP

To enable the BFD session between the two RSVP peers, the user should follow these steps:

First, configure BFD on interfaces between PE-1 and PE-2 as described in [BFD Base Parameter Configuration and Troubleshooting on page 207](#).

Next, configure MPLS, creating the path, the LSP and the interfaces within MPLS (and RSVP).

On PE-1:

```
configure router
  mpls
    interface system
    exit
    interface PE-1-PE-2
    exit
  exit
  rsvp
    interface system
    exit
    interface PE-1-PE-2
    exit
    no shutdown
  exit
  mpls
    path dyn
    no shutdown
  exit
  lsp LSP-PE-1-PE-2
    to 192.0.1.2
    cspf
```

BFD for RSVP

```
        primary dyn
        exit
        no shutdown
    exit
    no shutdown
exit
```

On PE-2:

```
configure router
    mpls
        interface system
        exit
        interface PE-2-PE-1
        exit
    exit
    rsvp
        interface system
        exit
        interface PE-2-PE-1
        exit
        no shutdown
    exit
    mpls
        path dyn
        no shutdown
        exit
        lsp LSP-PE-2-PE-1
        to 192.0.1.1
        cspf
        primary dyn
        exit
        no shutdown
    exit
    no shutdown
exit
```


Next, verify that the RSVP session is up.

```
A:PE-1# show router rsvp session
=====
RSVP Sessions
=====
```

From	To	Tunnel ID	LSP ID	Name	State
192.0.2.2	192.0.2.1	2	516	LSP-PE-2-PE-1::dyn	Up
192.0.2.1	192.0.2.2	1	61446	LSP-PE-1-PE-2::dyn	Up

```
-----
Sessions : 2
=====
A:PE-1#
```

Then, apply BFD on the RSVP Interfaces.

On PE1:

```
configure
router
    rsvp
        interface PE-1-PE-2
            bfd-enable
        exit
        no shutdown
    exit
exit
```

On PE2:

```
configure
router
    rsvp
        interface PE-2-PE-1
            bfd-enable
        exit
        no shutdown
    exit
exit
```

Finally, verify that the BFD session is operational between PE1 and PE2.

On PE1:

```
=====
BFD Session
=====
Interface          State          Tx Intvl  Rx Intvl  Multipl
Remote Address     Protocols     Tx Pkts   Rx Pkts   Type
-----
PE-1-PE-2          Up (3)        100       100       3
192.168.1.2        rsvp          31515     31506     iom
-----
No. of BFD sessions: 1
=====
```

On PE2:

```
=====
BFD Session
=====
Interface          State          Tx Intvl  Rx Intvl  Multipl
Remote Address     Protocols     Tx Pkts   Rx Pkts   Type
-----
PE-2-PE-1          Up (3)        100       100       3
192.168.1.1        rsvp          31563     31572     iom
-----
No. of BFD sessions: 1
=====
```

BFD for T-LDP

BFD tracking of an LDP session associated with a T-LDP adjacency allows for faster detection of the liveliness of the session by registering the transport address of an LDP session with a BFD session.

The goal of this paragraph is to configure BFD for T-LDP, referring to the scheme shown in [Figure 48](#).

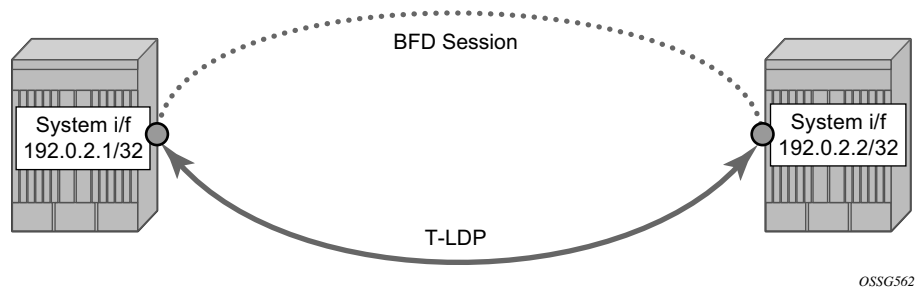


Figure 48: BFD for T-LDP

The parameters used for the BFD session are configured under the loopback interface corresponding to the LSR-ID (by default, the LSR-ID matches the system interface address).

```
configure
router
  interface system
    address 192.0.2.1/32
    bfd 3000 receive 3000 multiplier 3
  exit
exit
exit
```

By enabling BFD for a selected targeted session, the state of that session is tied to the state of the underlying BFD session between the two nodes.

When using BFD over other links with the ability to reroute, such as spoke-SDPs, the interval and multiplier values configuring BFD should be set to allow sufficient time for the underlying network to re-converge before the associated BFD session expires. A general rule of thumb should be that the expiration time (interval * multiplier) is three times the convergence time for the IGP network between the two endpoints of the BFD session.

Before enabling BFD, ensure that the T-LDP session is up.

On PE-1:

```
B:PE-1# show router ldp session
```

```
=====
LDP Sessions
=====
Peer LDP Id      Adj Type  State      Msg Sent  Msg Recv  Up Time
-----
192.0.2.2      Targeted  Established  35        41        0d 00:02:50
-----
=====
```

On PE-2:

```
B:PE-2# show router ldp session
```

```
=====
LDP Sessions
=====
Peer LDP Id      Adj Type  State      Msg Sent  Msg Recv  Up Time
-----
192.0.2.1      Targeted  Established  27        23        0d 00:01:32
-----
=====
```

Then, enable the BFD session.

```
configure
router
  ldp
    targeted-session
      peer 192.0.2.2
      bfd-enable
    exit
  exit
exit
exit
exit
```

Note that the loopback interface can be used to source BFD sessions to many peers in the network.

Finally, check that the BFD session is up.

On PE-1:

```
A:PE-1# show router bfd session
```

```
=====
BFD Session
=====
Interface      State      Tx Intvl  Rx Intvl  Multipl
Remote Address  Protocols  Tx Pkts   Rx Pkts   Type
-----
system          Up (3)     100       100       3
```

Bi-Directional Forwarding Detection

192.0.2.2	ldp	N/A	N/A	cpm-np
-----------	-----	-----	-----	--------

On PE-2:

```
A:PE-1# show router bfd session
```

```
=====
```

BFD Session

```
=====
```

Interface	State	Tx Intvl	Rx Intvl	Multipl
Remote Address	Protocols	Tx Pkts	Rx Pkts	Type
system	Up (3)	100	100	3
192.0.2.1	ldp	N/A	N/A	cpm-np

When the T-LDP session comes up, a centralized BFD session is always created even if the local interface has a direct link to the peer.

BFD Support of OSPF PE-CE Adjacencies

This feature, introduced with Release 8.0, extends BFD support to OSPF within a VPRN context when OSPF is used as the PE-CE protocol. In this section, the topology shown in [Figure 49](#).

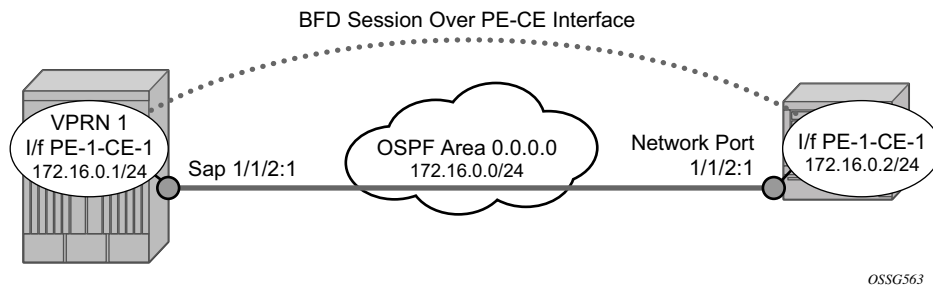


Figure 49: BFD for OSPF PE-CE I/F

First, configure the VPRN service interface PE-1-CE-1 on PE-1 with BFD parameters.

```
config
  service
    vprn 1 customer 1 create
    route-distinguisher 1:1
    vrf-target target:1:1
    interface PE-1-CE-1 create
      address 172.16.0.1/24
      bfd 100 receive 100 multiplier 3
      sap 1/1/1:1 create
    exit
  exit
  ospf
    area 0.0.0.0
    interface PE-1-CE-1
    exit
  exit
exit
no shutdown
exit
exit
exit
```

Next, configure the router interface on CE-1 and add it to the OSPF area 0 domain.

```
configure
  router
    interface CE-1-PE-1
      address 172.16.0.2/24
      port 1/1/1:1
      bfd 100 receive 100 multiplier 3
    exit
  ospf
    area 0.0.0.0
      interface CE-1-PE-1
        exit
      exit
    exit
  exit
exit
```

Then, ensure that OSPF adjacency is up.

On PE-1:

```
A:PE-1>config>service>vprn# show router 1 ospf neighbor
=====
OSPF Neighbors
=====
Interface-Name          Rtr Id          State          Pri  RetxQ  TTL
-----
PE-1-CE-1              192.0.2.5       Full           1    2      33
-----
No. of Neighbors: 1
=====
```

On CE-1:

```
A:CE-1# show router ospf neighbor
=====
OSPF Neighbors
=====
Interface-Name          Rtr Id          State          Pri  RetxQ  TTL
-----
CE-1-PE-1              192.0.2.1       Full           1    0      31
-----
No. of Neighbors: 1
=====
```

BFD Support of OSPF PE-CE Adjacencies

Then, enable BFD on the PE-1-CE-1 interface on PE-1.

```
configure service vprn 1 ospf area 0.0.0.0 interface PE-1-CE-1 bfd-enable
```

Enable BFD on the CE-1-PE-1 interface on CE-1.

```
configure router ospf area 0.0.0.0 interface CE-1-PE-1 bfd-enable
```

Finally, check that the BFD sessions are up in both PE-1 and CE-1.

```
A:PE-1# show router 1 bfd session
```

```
=====
BFD Session
=====
```

Interface	State	Tx Intvl	Rx Intvl	Multipl
Remote Address	Protocols	Tx Pkts	Rx Pkts	Type
PE-1-CE-1	Up (3)	100	100	3
172.16.0.2	ospf2	6331	6340	iom

```
-----
No. of BFD sessions: 1
```

```
A:CE-1# show router bfd session
```

```
=====
BFD Session
=====
```

Interface	State	Tx Intvl	Rx Intvl	Multipl
Remote Address	Protocols	Tx Pkts	Rx Pkts	Type
CE-1-PE-1	Up (3)	100	100	3
172.16.0.1	ospf2	6691	6682	iom

```
-----
No. of BFD sessions: 1
```


BFD within IPsec Tunnels

The ability to assign a BFD session to a given static LAN-to-LAN IPsec tunnel that provides heart-beat mechanism for fast failure detection has been introduced in Release.8.0.

IPsec needs a Multi-service Integrated Service Adapter (MS-ISA) installed, so this scenario is only applicable to 7750 SR-7/12 equipped with IOM-2 or 3.

In this section, the topology is shown in [Figure 50.s](#)

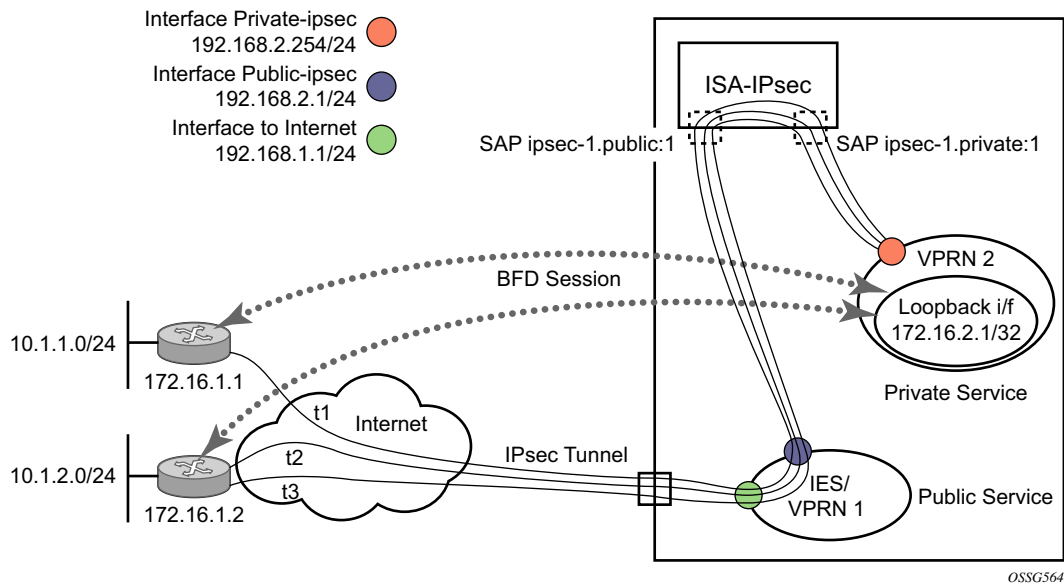


Figure 50: BFD Sessions within IPsec Tunnels

The first step is to configure MS-ISA card as **type isa-tunnel**.

```
configure
  card 1
    card-type iom3-xp
    mda 1
      mda-type isa-tunnel
    exit
    mda 2
      mda-type m10-1gb-sfp-b
    exit
  exit
exit
```

Next, instantiate the tunnels t1, t2 and t3 from the private service (in this example, VPRN 2) to the peers passing through the public service (in this example VPRN 1, but it could be instead an IES).

Since the configuration of IPSec tunnels is out of the scope of this section, only relevant command lines are reported to configure the interfaces shown in [Figure 50](#).

```
configure service
  vprn 1 customer 1 create
    route-distinguisher 1:1
    interface toInternet create
      address 192.168.1.1/24
      sap 1/2/1 create
    exit
  exit
  interface public-ipsec create
    address 192.168.2.1/24
    sap tunnel-1.public:1 create
  exit
  exit
  no shutdown
exit
vprn 2 customer 1 create
  ipsec
    security-policy 1 create
      entry 10 create
        local-ip 192.168.3.1/32
        remote-ip any
      exit
    exit
  exit
  route-distinguisher 1:2
  interface private-ipsec tunnel create
    sap tunnel-1.private:1 create
    ipsec-tunnel t1 create
      local-gateway-address 192.168.2.254 peer 172.16.1.1 delivery-service 1
    exit
  exit
  ipsec-tunnel t2 create
    local-gateway-address 192.168.2.254 peer 172.16.1.2 delivery-service 1
  exit
  exit
  ipsec-tunnel t3 create
    local-gateway-address 192.168.2.254 peer 172.16.1.2 delivery-service 1
  exit
  exit
  exit
  interface loop create
    address 172.16.2.1/32
    loopback
  exit
  static-route 10.1.1.0/24 ipsec-tunnel t1
  static-route 10.1.2.0/24 ipsec-tunnel t2 metric 1
  static-route 10.1.2.0/24 ipsec-tunnel t3 metric 5
  no shutdown
```

Then configure the BFD parameters within loopback interface loop (refer to [BFD Base Parameter Configuration and Troubleshooting on page 207](#)).

```
configure service vprn 2
    interface loop
        bfd 100 receive 100 multiplier 3
    exit
exit
```

And finally enable BFD within the tunnels.

```
configure service
    vprn 2
        interface private-ipsec tunnel
            sap tunnel-1.private:1
            ipsec-tunnel t1
                bfd-enable service 2 interface loop dst-ip 172.16.1.1
            exit
            ipsec-tunnel t2
                bfd-enable service 2 interface loop dst-ip 172.16.1.2
                bfd-designate
            exit
            ipsec-tunnel t3
                bfd-enable service 2 interface loop dst-ip 172.16.1.2
exit all
```

The BFD-enable parameters are as follows:

- **service** *<service-id>* — Specifies the service-id where the BFD session resides.
- **interface** *<interface-name>* — Specifies the name of the interface used by the BFD session.
- **dst-ip** *<ip-address>* — Specifies the destination address to be used for the BFD session.

The following statements are to be taken into consideration to correctly configure BFD in this environment:

- BFD over IPSec sessions are centralized, managed by the hardware on the CPM.
- Only BFD over static lan-to-lan tunnel is supported in Release 8.0 (not dynamic).
- Only one BFD session is allowed between a given source/destination address pair.
- Each tunnel can be associated to only one BFD session but multiple tunnels can be associated to the same BFD session.
- In case of multiple tunnels sharing the same BFD session, one IPSec tunnel carries BFD traffic: the BFD-DESIGNATED tunnel.

Referring to [Figure 50](#) and to the above configuration, the tunnels t2 and t3 share the same BFD-session. Tunnel t2 is the bfd-designated tunnel, the BFD session runs within it and the other tunnel t3 shares its BFD session. If the BFD session goes down, the system will bring down both the designated tunnel t2 and the associated tunnel t3.

The state machine in [Figure 51](#) shows the decision process in case of shared BFD sessions.

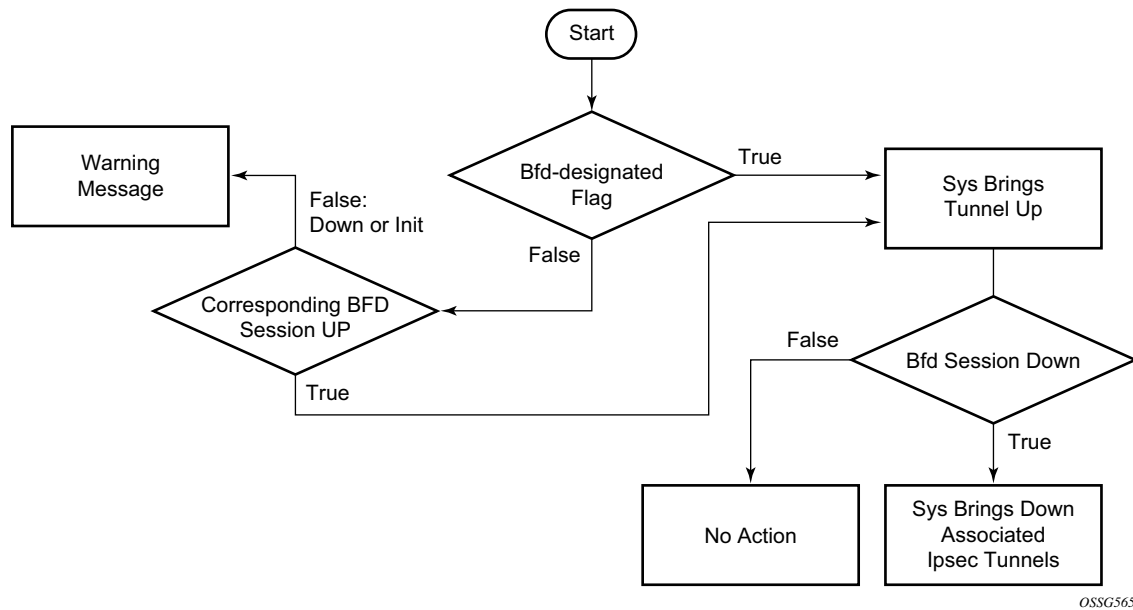


Figure 51: Logic for Shared BFD Sessions

BFD for VRRP

This feature assigns a BFD session to provide a heart-beat mechanism for the given VRRP/SRRP instance. It should be noted that there can be only one BFD session assigned to any given VRRP/SRRP instance, but there can be multiple SRRP/VRRP sessions using the same BFD session.

In this section, the topology is shown in [Figure 52](#).

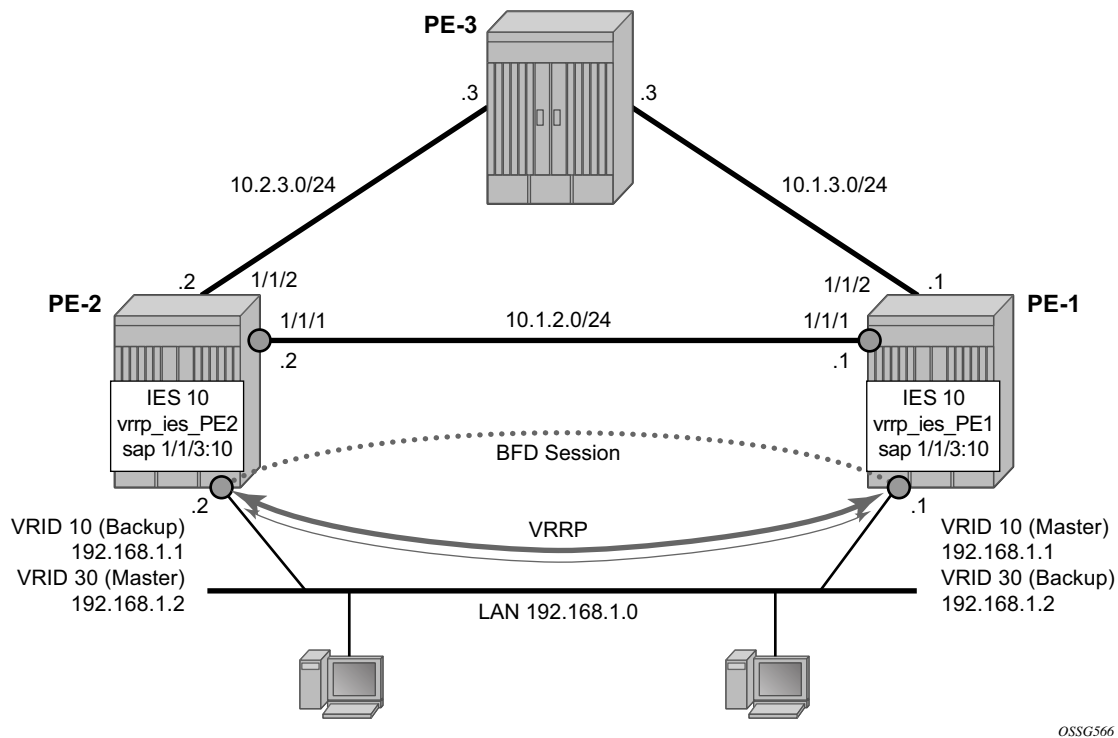


Figure 52: BFD for VRRP

First, create the LAN subnet. Two PE routers are connected by IES or VPRN services (in following examples IES 10 is created in both routers).

On PE-1:

```
configure service ies 10 customer 1 create
    interface vrrp_ies_PE1 create
        address 192.168.1.1/24
        sap 1/1/3:10 create
        exit
    exit
no shutdown
```

```
exit
```

On PE-2:

```
configure service ies 10 customer 1 create
    interface vrrp_ies_PE2 create
        address 192.168.1.2/24
        sap 1/1/3:10 create
    exit
exit
no shutdown
exit
```

Verify that the IES services are operational (**show service service-using**) and verify that you can ping the remote interface IP address.

Next, configure the VRRP parameters for both PE-1 and PE-2, enable VRRP on the IES interface that connects to the 192.168.1.0/24 subnet.

In this section, the configurations are shown for the VRRP owner mode for master but any other scenario for VRRP can be configured (non owner mode for master).

In the following examples two VRRP instances are created on the 192.168.1.0/24 subnet:

VRID = 10	Master (owner) = PE-1
	Backup = PE-2
	VRRP IP = 192.168.1.1
VRID = 30	Master (owner) = PE-2
	Backup = PE-1
	VRRP IP = 192.168.1.2

Host 1 is configured with default gateway = 192.168.1.1

Host 2 is configured with default gateway = 192.168.1.2

On PE-1:

```
configure service ies 10 interface vrrp_ies_PE1
    vrrp 10 owner
        backup 192.168.1.1
    exit
    vrrp 30
        backup 192.168.1.2
        ping-reply
        telnet-reply
        ssh-reply
    exit
```

On PE-2:

```
configure service ies 10 interface vrrp_ies_PE2
    vrrp 10
        backup 192.168.1.1
        ping-reply
        telnet-reply
        ssh-reply
    exit
    vrrp 30 owner
        backup 192.168.1.2
    exit
```

To bind the VRRP instances with a BFD session, add the following command under any VRRP instance: **bfd-enable service-id interface interface-name dst-ip ip-address**.

Note that the IES service-id must be declared where the interface is configured.

On PE-1:

```
configure service ies 10 interface vrrp_ies_PE1
    vrrp 10 owner
    bfd-enable 10 interface vrrp_ies_PE1 dst-ip 192.168.1.2
    exit
    vrrp 30
    bfd-enable 10 interface vrrp_ies_PE1 dst-ip 192.168.1.2
    exit
```

On PE-2:

```
configure service ies 10 interface vrrp_ies_PE2
    vrrp 10 owner
    bfd-enable 10 interface vrrp_ies_PE2 dst-ip 192.168.1.1
    exit
    vrrp 30
    bfd-enable 10 interface vrrp_ies_PE2 dst-ip 192.168.1.1
    exit
```

The parameters used for the BFD are set by the BFD command under the IP interface.

Note that unlike the previous scenarios, the user can enter the commands above, enabling the BFD session, even if the specified interface (vrrp_ies_PE1) has not been configured with BFD parameters.

If it has not been configured yet, the BFD session will be initiated only after the following configuration.

BFD for VRRP

On PE-1:

```
configure service ies 10 interface vrrp_ies_PE1
    bfd 1000 receive 1000 multiplier 3
```

On PE-2:

```
configure service ies 10 interface vrrp_ies_PE2
    bfd 1000 receive 1000 multiplier 3
```

Finally, verify that the BFD session is up (for instance on PE-1):

```
A:PE1>show router bfd session src 192.168.1.1 detail
=====
BFD Session
=====
Remote Address : 192.168.1.2
Admin State   : Up                               Oper State    : Up (3)
Protocols     : vrrp
Rx Interval   : 100                             Tx Interval   : 100
Multiplier    : 3                               Echo Interval : 0
Recd Msgs     : 7404                             Sent Msgs     : 7412
Up Time       : 0d 00:04:26                       Up Transitions : 2
Down Time     : None                             Down Transitions : 1
Version Mismatch : 0

Forwarding Information

Local Discr   : 4006                             Local State   : Up (3)
Local Diag    : 1 (Detect time expired)          Local Mode    : Async
Local Min Tx  : 100                             Local Mult    : 3
Last Sent     : 12/14/2010 17:44:34              Local Min Rx  : 100
Type          : iom
Remote Discr   : 4003                             Remote State  : Up (3)
Remote Diag    : 1 (Detect time expired)          Remote Mode   : Async
Remote Min Tx  : 100                             Remote Mult   : 3
Last Recv     : 12/14/2010 17:44:34              Remote Min Rx : 100
=====
```

This session is shared by all the VRRP instances configured between the specified interfaces.

When BFD is configured in a VRRP instance, the following command gives details of BFD related to every instance:

```
show router vrrp instance interface vrrp_ies_PE1
=====
VRRP Instances for interface vrrp_ies_PE1
=====
-----
VRID 10
```



```

-----
Owner                : Yes                VRRP State         : Master
Primary IP of Master: 192.168.1.1 (Self)
Primary IP           : 192.168.1.1        Standby-Forwarding: Disabled
VRRP Backup Addr     : 192.168.1.1
Admin State          : Up                 Oper State           : Up
Up Time              : 12/14/2010 16:47:47 Virt MAC Addr       : 00:00:5e:00:01:0a
Auth Type            : None
Config Mesg Intvl    : 1                  In-Use Mesg Intvl    : 1
Base Priority         : 255                In-Use Priority       : 255
Init Delay           : 0                   Init Timer Expires: 0.000 sec
Creation State        : Active
-----

```

BFD Interface

```

-----
Service ID           : 10
Interface Name        : vrrp_ies_PE1
Src IP                : 192.168.1.1
Dst IP                : 192.168.1.2
Session Oper State    : connected
-----

```

Master Information

```

-----
Primary IP of Master: 192.168.1.1 (Self)
Addr List Mismatch   : No                Master Priority      : 255
Master Since         : 12/14/2010 16:47:47
-----

```

Masters Seen (Last 32)

```

-----
Primary IP of Master  Last Seen          Addr List Mismatch  Msg Count
-----
192.168.1.1           12/14/2010 16:47:47  No                  0
192.168.1.2           12/14/2010 17:39:57  No                  5
-----

```

Statistics

```

-----
Become Master        : 7                Master Changes      : 7
Adv Sent              : 347577            Adv Received         : 5
Pri Zero Pkts Sent   : 6                Pri Zero Pkts Rcvd  : 0
Preempt Events        : 0                Preempted Events     : 0
Mesg Intvl Discards   : 0                Mesg Intvl Errors    : 0
Addr List Discards    : 0                Addr List Errors     : 0
Auth Type Mismatch    : 0                Auth Failures        : 0
Invalid Auth Type     : 0                Invalid Pkt Type     : 0
IP TTL Errors         : 0                Pkt Length Errors    : 0
Total Discards        : 0
-----

```

VRID 30

```

-----
Owner                : No                VRRP State         : Backup
Primary IP of Master: 192.168.1.2 (Other)
Primary IP           : 192.168.1.1        Standby-Forwarding: Disabled
VRRP Backup Addr     : 192.168.1.2
Admin State          : Up                 Oper State           : Up
Up Time              : 12/14/2010 17:39:49 Virt MAC Addr       : 00:00:5e:00:01:1e
Auth Type            : None
Config Mesg Intvl    : 1                  In-Use Mesg Intvl    : 1
Master Inherit Intvl: No
-----

```

BFD for VRRP

```
Base Priority      : 100                      In-Use Priority   : 100
Policy ID         : n/a                      Preempt Mode     : Yes
Ping Reply        : Yes                      Telnet Reply     : Yes
SSH Reply         : Yes                      Traceroute Reply : No
Init Delay        : 0                        Init Timer Expires: 0.000 sec
Creation State    : Active

-----
BFD Interface
-----
Service ID        : 10
Interface Name    : vrrp_ies_PE1
Src IP            : 192.168.1.1
Dst IP            : 192.168.1.2
Session Oper State : connected

-----
Master Information
-----
Primary IP of Master: 192.168.1.2 (Other)
Addr List Mismatch  : No                      Master Priority   : 255
Master Since        : 12/14/2010 17:39:57
Master Down Interval: 3.609 sec (Expires in 3.000 sec)

-----
Masters Seen (Last 32)
-----
Primary IP of Master  Last Seen          Addr List Mismatch  Msg Count
-----
192.168.1.1           12/14/2010 17:39:57  No                  0
192.168.1.2           12/14/2010 17:54:03  No                  342583

-----
Statistics
-----
Become Master        : 6                      Master Changes     : 11
Adv Sent              : 4441                    Adv Received       : 342583
Pri Zero Pkts Sent   : 1                      Pri Zero Pkts Rcvd : 0
Preempt Events       : 0                      Preempted Events   : 5
Mesg Intvl Discards  : 0                      Mesg Intvl Errors  : 0
Addr List Discards   : 0                      Addr List Errors    : 338989
Auth Type Mismatch   : 0                      Auth Failures       : 0
Invalid Auth Type    : 0                      Invalid Pkt Type    : 0
IP TTL Errors        : 0                      Pkt Length Errors  : 0
Total Discards       : 0

=====
```

Finally, for troubleshooting: it could be that the BFD session between the two IP interfaces is up but (in one or both peers) the command **show router vrrp instance interface *interface-name*** gives the following output regarding BFD for one or more VRID's.

```
-----
BFD Interface
-----
Service ID        : None
Interface Name    : vrrp_ies_PE2
Src IP            : 0.0.0.0
Dst IP            : 192.168.1.1
Session Oper State : notConfigured

-----
```

To fix this, check that BFD has been correctly configured for the VRRP instances.

For instance, in the following example, the cause of the misconfiguration is that the IES service-id is not declared in the bfd-enable command:

```
configure service ies 10 interface vrrp_ies_PE2
    vrrp 10 owner
    bfd-enable interface vrrp_ies_PE2 dst-ip 192.168.1.1
exit
```

Conclusion

BFD is a light-weight protocol which provides rapid path failure detection between two systems and it is useful in situations where the physical network has numerous intervening hops which are not part of the Layer 3 network.

BFD is linked to a protocol state. For BFD session to be established, the prerequisite condition is that the protocol to which the BFD is linked must be operationally active. Once the BFD session is established, the state of the protocol to which BFD is tied to is then determined based on the BFD session's state. This means that if the BFD session goes down, the corresponding protocol will be brought down.

In this section every scenario where BFD could be implemented has been described, including the configuration, show output and troubleshooting hints.

LFA Policies Using OSPF as IGP

In This Chapter

This section provides information about LFA policies using OSPF as IGP.

Topics in this section include:

- [Applicability on page 250](#)
- [Overview on page 251](#)
- [Configuration on page 253](#)
- [Conclusion on page 266](#)

Applicability

Loop Free Alternate (LFA) policies is a local control plane feature. The functionality is limited to the hardware supported by LDP Fast ReRoute (FRR) and IP FRR:

- LDP FRR is supported on the 7750 SR-7/12, 7450 ESS-6/6v/7/12 in all chassis modes, on the 7450 ESS-6/6v/7/12 in mixed-mode and on the 7950 XRS-16/20/40. It is also supported on the 7750 SR-c4/12 and 7710 SR-c4/c12 platforms.
- IP FRR is supported on the 7750 SR-7/12 in chassis mode D, on the 7450 ESS-6/6v/7/12 in chassis mode D with or without mixed-mode. It is also supported on the 7750 SR-c4/12 platforms.

This configuration was tested on release 12.0.R4.

Overview

When multiple LFAs exist, RFC 5286, *Basic Specification for IP Fast Reroute: Loop-Free Alternates*, chooses the selection of the LFA providing the best coverage of the failure cases. In general, this means that node LFA has preference above link LFA. In some deployments, however, this can lead to suboptimal LFA. For example an aggregation router (typically using lower bandwidth links) protecting a core node/link (typically using high bandwidth links) is potentially undesirable.

For this reason, the operator wants to have more control in the LFA next-hop selection algorithm. This is achieved by the introduction of LFA Shortest Path First (SPF) policies.

LFA policies can work in combination with IP FRR and/or LDP FRR.

Implementation

The 7x50 LFA policy implementation is built around the concept of route next-hop (NH) templates which are applied to IP interfaces. A route-next-hop template specifies criteria which influence the selection of an LFA backup NH for either:

- a set of prefixes in a prefix-list or
- a set of prefixes which resolve to a specific primary NH

Refer to <http://tools.ietf.org/html/draft-litkowski-rtgwg-lfa-manageability> for further information. Two powerful methods which can be used as criteria inside a route-next-hop template are IP admin-groups and IP Shared Risk Link Group (SRLG). IP SRLG and IP admin-group criteria are applied before running the LFA NH algorithm. IP Admin-groups and IP SRLG work in a similar way as the well-known MPLS admin-groups and MPLS SRLG.

For example, when one or more IP admin-groups/SRLG are applied to an IP interface, the same MPLS admin-group/SRLG rules apply:

- IP interfaces which do not include one or more of the admin-groups in the **include** statements are pruned before computing the LFA next-hop.
- IP interfaces which belong to admin-groups which have been explicitly excluded using the **exclude** statement are pruned before computing the LFA next-hop.
- IP interfaces which belong to the SRLGs used by the primary NH of a prefix are pruned before computing the LFA next-hop.

For compatibility reasons with the existing MPLS, admin-groups and SRLG, a single set of admin-groups and SRLGs are defined within the **configure router if-attribute** context from 12.0.R1 onward. Configuration of admin-groups and SRLGs in the **configure router mpls** context is deprecated from this release onwards.

Implementation

Once one or more admin-groups/SRLGs have been defined, it is possible to apply them on an MPLS interface and/or an IP interface.

In the current implementation IP admin-groups/SRLGs are locally significant, meaning they are not advertised by the IGP.

Keep in mind that the well-known MPLS admin-groups/SRLGs are advertised in TE link TLVs and sub-TLVs when the traffic-engineering option is enabled in the IGP protocol.

Other selection criteria which can be configured inside a route-next-hop template are protection type preference and NH type preference. More details on these parameters are provided later in this example.

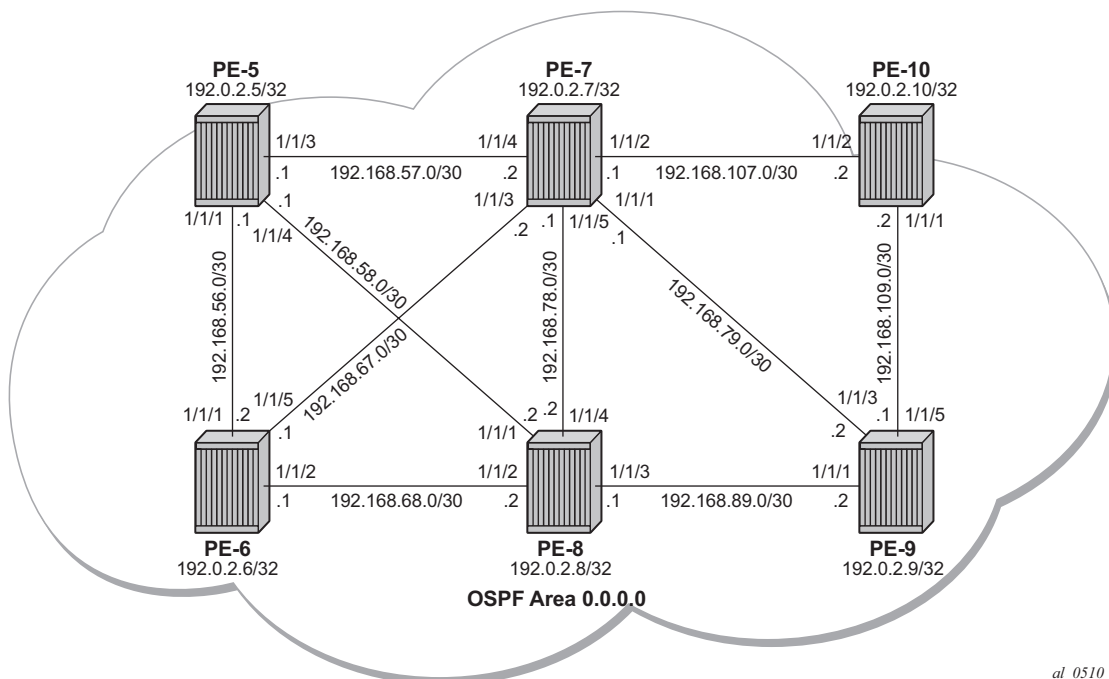


Figure 53: Network Topology

Configuration

Step 1. Configuring an IP/MPLS network with LDP FRR enabled on PE-7.

Since the focus is not on how to setup an IP/MPLS network, only summary bullets are provided.

- The system and IP interface addresses are configured according to [Figure 53](#).
- OSPF area 0 is selected as the interior gateway protocol (IGP) to distribute routing information between all PEs. All OSPF interfaces are setup as **type point-to-point** to avoid running the DR/BDR election process.
- Enable link LDP on all interfaces. This establishes a full mesh of LDP LSPs between all PEs' system interfaces. As an example, the tunnel-table on PE-7 looks like this:

```
*A:PE-7# show router tunnel-table
=====
Tunnel Table (Router: Base)
=====
Destination          Owner Encap TunnelId Pref    Nexthop      Metric
-----
192.0.2.5/32         ldp   MPLS   -       9       192.168.57.1 1000
192.0.2.6/32         ldp   MPLS   -       9       192.168.67.1 1000
192.0.2.8/32         ldp   MPLS   -       9       192.168.57.1 2000
192.0.2.9/32         ldp   MPLS   -       9       192.168.79.2 1000
192.0.2.10/32        ldp   MPLS   -       9       192.168.107.2 1000
-----
Flags: B = BGP backup route available
       E = inactive best-external BGP route
=====
*A:PE-7#
```

Note that the LDP LSP metric follows the IGP cost.

- Enable LDP FRR on PE-7. This is a two-fold configuration command: first the IGP needs to be triggered to do LFA NH computation, and secondly, FRR needs to be enabled within the LDP context. Translated into configuration commands, this becomes:

```
*A:PE-7# configure router ospf loopfree-alternate
*A:PE-7# show router ospf status | match LFA
LFA                               : Enabled

*A:PE-7# configure router ldp fast-reroute

*A:PE-7# show router ldp status | match FRR
FRR                               : Enabled          Mcast Upstream FRR    : Disabled
```

After issuing these two CLI commands, the software pre-computes both a primary and a backup Next-hop Label Forwarding Entry (NHLFE) for each LDP FEC in the network and downloads it to the IOM/IMM. The primary NHLFE corresponds to the label of the FEC received from the primary NH as per standard LDP resolution of the FEC prefix in the Routing Table Manager (RTM). The backup NHLFE corresponds to the label received for the same FEC from an LFA NH.

The **show router route-table alternative** command adds an LFA flag to the associated alternative NH for a specific destination prefix. Other useful IGP related show commands are **show router ospf lfa-coverage** and **show router ospf routes alternative detail**.

```
*A:PE-7# show router route-table alternative
```

```
=====
Route Table (Router: Base)
=====
```

Dest Prefix[Flags]	Type	Proto	Age	Pref
Next Hop[Interface Name]			Metric	
Alt-NextHop			Alt-Metric	
192.0.2.5/32	Remote	OSPF	16h27m07s	10
192.168.57.1			1000	
192.168.67.1 (LFA)			2000	
192.0.2.6/32	Remote	OSPF	16h27m07s	10
192.168.67.1			1000	
192.168.57.1 (LFA)			2000	
192.0.2.7/32	Local	Local	16h37m53s	0
system			0	
192.0.2.8/32	Remote	OSPF	16h24m34s	10
192.168.57.1			2000	
192.168.67.1 (LFA)			2000	
192.0.2.9/32	Remote	OSPF	16h22m15s	10
192.168.79.2			1000	
192.168.107.2 (LFA)			2000	
192.0.2.10/32	Remote	OSPF	16h20m19s	10
192.168.107.2			1000	
192.168.79.2 (LFA)			2000	
192.168.56.0/30	Remote	OSPF	16h27m07s	10
192.168.57.1			2000	
192.168.67.1 (LFA)			3000	
192.168.57.0/30	Local	Local	16h29m11s	0
int-PE-7-PE-5			0	
192.168.58.0/30	Remote	OSPF	16h27m07s	10
192.168.57.1			2000	
192.168.67.1 (LFA)			3000	
192.168.67.0/30	Local	Local	16h28m55s	0
int-PE-7-PE-6			0	
192.168.68.0/30	Remote	OSPF	16h27m07s	10
192.168.67.1			2000	
192.168.57.1 (LFA)			3000	
192.168.78.0/30	Remote	OSPF	16h24m27s	10
192.168.57.1			3000	
192.168.67.1 (LFA)			3000	
192.168.79.0/30	Local	Local	16h28m16s	0
int-PE-7-PE-9			0	
192.168.89.0/30	Remote	OSPF	16h22m10s	10
192.168.79.2			2000	
192.168.107.2 (LFA)			3000	
192.168.107.0/30	Local	Local	16h27m52s	0
int-PE-7-PE-10			0	
192.168.109.0/30	Remote	OSPF	16h22m10s	10
192.168.79.2			2000	
192.168.107.2 (LFA)			3000	

```
-----
*A:PE-7#
```

Displaying the Label Forwarding Information Base (LFIB) on PE-7 shows the available alternate NHs; displayed with BU flag.

```
*A:PE-7# show router ldp bindings active
=====
Legend:  (S) - Static          (M) - Multi-homed Secondary Support
          (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP Prefix Bindings (Active)
=====
Prefix                Op    IngLbl  EgrLbl  EgrIntf/LspId  EgrNextHop
-----
192.0.2.5/32          Push  --      262143  1/1/4          192.168.57.1
192.0.2.5/32          Push  --      262142BU 1/1/3          192.168.67.1
192.0.2.5/32          Swap  262142  262143  1/1/4          192.168.57.1
192.0.2.5/32          Swap  262142  262142BU 1/1/3          192.168.67.1
192.0.2.6/32          Push  --      262143  1/1/3          192.168.67.1
192.0.2.6/32          Push  --      262142BU 1/1/4          192.168.57.1
192.0.2.6/32          Swap  262141  262143  1/1/3          192.168.67.1
192.0.2.6/32          Swap  262141  262142BU 1/1/4          192.168.57.1
192.0.2.7/32          Pop   262143  --      --             --
192.0.2.8/32          Push  --      262140  1/1/4          192.168.57.1
192.0.2.8/32          Push  --      262140BU 1/1/3          192.168.67.1
192.0.2.8/32          Swap  262140  262140  1/1/4          192.168.57.1
192.0.2.8/32          Swap  262140  262140BU 1/1/3          192.168.67.1
192.0.2.9/32          Push  --      262143  1/1/1          192.168.79.2
192.0.2.9/32          Push  --      262138BU 1/1/2          192.168.107.2
192.0.2.9/32          Swap  262139  262143  1/1/1          192.168.79.2
192.0.2.9/32          Swap  262139  262138BU 1/1/2          192.168.107.2
192.0.2.10/32         Push  --      262143  1/1/2          192.168.107.2
192.0.2.10/32         Push  --      262138BU 1/1/1          192.168.79.2
192.0.2.10/32         Swap  262138  262143  1/1/2          192.168.107.2
192.0.2.10/32         Swap  262138  262138BU 1/1/1          192.168.79.2
=====
```

```
*A:PE-7#
```

Finally, a synchronization timer is enabled between the IGP and LDP protocol when LDP FRR is enabled. From the moment that the interface for the previous primary NH is restored, the IGP may re-converge back to that interface before LDP has completed the FEC exchange with its neighbor over that interface. This may cause LDP to de-program the LFA NH from the FEC and blackhole the traffic. In this example a timer of 10 seconds is used. Translated into configuration commands, this becomes:

```
*A:PE-x# configure router interface <itf-name> ldp-sync-timer 10
```

When this timer is set, when a failed interface is subsequently restored, the IGP advertises this link into the network with an infinite metric for the period of this timer. When the failed link is restored, the **ldp-sync-timer** is started, and LDP adjacencies are brought up over the restored link and a label exchange is completed between the peers. After the **ldp-sync-timer expires**, the normal metric is advertised into the network again.

At this point, everything is in place to start creating LFA policies to influence the calculated LFA NHs.

Step 2. Create a route-next-hop policy template.

This is a mandatory step in the context of LFA policies. The route-next-hop template name is maximum of 32 characters long. Creating a route-next-hop policy is done in the following way:

```
*A:PE-x# configure router route-next-hop-policy template <template name>
```

Commands within a **route-next-hop** policy template follow the **begin-abort-commit** model. After a **commit**, the IGP re-evaluates the template and schedules a new LFA SPF to re-compute the LFA NH for the prefixes associated with this template.

Step 3. Configure admin-group constraints in route-next-hop policy.

This is an optional step in the context of LFA policies. Firstly, configure a group-name and a group-value, of each admin-group locally on the router. Translated into configuration commands:

```
*A:PE-x# configure router if-attribute admin-group <group-name> value <group-value>
```

Second, configure the admin-group membership of the IP interface(s) (network, IES or VPRN). Up to five admin-groups can be applied to an IP interface in one command but the command can be applied multiple times. The configured IP admin-group membership is applied in all levels/ areas the interface is participating in. Translated into configuration commands:

```
*A:PE-x# configure router interface <itf-name> if-attribute admin-group <group-name> [
<group-name> ... (upto 5 max)]
*A:PE-x# configure service vprn <svc-id> interface <itf-name> if-attribute admin-group
<group-name> [ <group-name> ... (upto 5 max)]
*A:PE-x# configure service ies <svc-id> interface <itf-name> if-attribute admin-group
<group-name> [ <group-name> ... (upto 5 max)]
```

Third, add the IP admin-group constraints into the route-next-hop policy template one by one. The **include-group** statement instructs the LFA SPF selection algorithm to select a subset of LFA NHs among the links which belong to one or more of the specified admin groups. A link which does not belong to at least one of the admin-groups is excluded. The **pref** option is used to provide a relative preference for the admin group selection. A lower preference value means that LFA SPF will first attempt to select an LFA backup NH which is a member of the corresponding admin group. If none is found, then the admin group with the next higher preference value is evaluated. If no preference is configured, then it is the least preferred (default preference value is 255).

When evaluating multiple **include-group** statements within the same preference, any link which belongs to one or more of the included admin groups can be selected as an LFA next-hop. There is no relative preference based on how many of those included admin groups the link is a member.

The **exclude-group** command simply prunes all links belonging to the specified admin group before making the LFA backup NH selection for a prefix. If the same group name is part of both **include** and **exclude** statements, the exclude statement will take precedence. In other words, the **exclude** statement can be viewed as having an implicit preference value of 0.

Translated into configuration commands, this becomes:

```
*A:PE-x# configure router route-next-hop-policy template <template-name> exclude-group
<group-name>
*A:PE-x# configure router route-next-hop-policy template <template-name> include-group
<group-name> [pref <preference>]
```

Step 4. Configure SRLG constraints in route-next-hop policy.

This is an optional step in the context of LFA policies. Firstly, configure a group-name and group-value, of each SRLG group locally on the router. Translated into configuration commands this becomes:

```
*A:PE-x# configure router if-attribute srlg-group <group-name> value <group-value>
```

Second, configure the SRLG group membership of the IP interfaces (network, IES or VPRN). Up to five SRLG groups can be applied to an IP interface in one command but the command can be applied multiple times. The configured IP SRLG group membership is applied in all levels/areas the interface is participating in. Translated into configuration commands this becomes:

```
*A:PE-x# configure router interface <itf-name> if-attribute srlg-group <group-name> [
<group-name> ... (upto 5 max)]
*A:PE-x# configure service vprn <svc-id> interface <itf-name> if-attribute srlg-group
<group-name> [ <group-name> ... (upto 5 max)]
*A:PE-x# configure service ies <svc-id> interface <itf-name> if-attribute srlg-group
<group-name> [ <group-name> ... (upto 5 max)]
```

Third, add the IP SRLG group constraints into the route-next-hop policy template. When this command is applied to a prefix, the LFA SPF attempts to select an LFA NH which uses an outgoing interface that does not participate in any of the SRLGs of the outgoing interface used by the primary NH. Translated into configuration commands, this becomes:

```
*A:PE-x# configure router route-next-hop-policy template <template-name> srlg-enable
```

Step 5. Configure the protection type in route-next-hop policy.

This is an optional step in the context of LFA policies. With the use of LFA policies, the user can also select if link protection or node protection is preferred for IP prefixes and LDP FEC prefixes protected by a backup LFA NH. By default, node protection is chosen. The implementation falls back to link protection if no LFA NH is found for node protection. Translated into configuration commands, this becomes:

```
*A:PE-x# configure router route-next-hop-policy template <template-name> protection-type {link|node}
```

Step 6. Configure the NH preference type in route-next-hop policy.

This is an optional step in the context of LFA policies. With the use of LFA policies, the user can also select if tunnel backup NH or IP backup NH is preferred for IP prefixes and LDP FEC prefixes protected by a backup LFA NH. By default, IP backup NH is chosen. The implementation falls back to the other type (tunnel) if no LFA NH of the preferred type is found. Translated into configuration commands, this becomes:

```
*A:PE-x# configure router route-next-hop-policy template <template-name> nh-type {ip|tunnel}
```

Step 7. Apply the route-next-hop policy template to an IP interface.

When the route-next-hop policy is applied to an IP interface, all prefixes using this interface as primary NH take the selection criteria specified in Step 3, Step 4, Step 5 and Step 6 into account. Translated into configuration commands, this becomes:

```
*A:PE-x# configure router ospf area interface lfa-policy-map route-nh-template <template-name>
*A:PE-x# configure router ospf3 area interface lfa-policy-map route-nh-template <template-name>
*A:PE-x# configure service vprn ospf area interface lfa-policy-map route-nh-template <template-name>
*A:PE-x# configure service vprn ospf3 area interface lfa-policy-map route-nh-template <template-name>
```

Examples

All of the examples focus on providing another LFA NH for LDP FEC prefix 192.0.2.6/32 and 192.0.2.5/32 (the system IP addresses of PE-6 and PE-5) , with PE-7 being the Point of Local Repair (PLR).

See [Figure 53 on page 252](#) for the network topology.

As shown earlier, the default LFA NH (without policy) for both LDP FEC prefixes is as follows:

```
*A:PE-7# show router ldp bindings active
=====
Legend:  (S) - Static          (M) - Multi-homed Secondary Support
          (B) - BGP Next Hop  (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP Prefix Bindings (Active)
=====
```

Prefix	Op	IngLbl	EgrLbl	EgrIntf/LspId	EgrNextHop
192.0.2.5/32	Push	--	262143	1/1/4	192.168.57.1
192.0.2.5/32	Push	--	262142BU	1/1/3	192.168.67.1
192.0.2.5/32	Swap	262136	262143	1/1/4	192.168.57.1
192.0.2.5/32	Swap	262136	262142BU	1/1/3	192.168.67.1
192.0.2.6/32	Push	--	262143	1/1/3	192.168.67.1
192.0.2.6/32	Push	--	262142BU	1/1/4	192.168.57.1
192.0.2.6/32	Swap	262135	262143	1/1/3	192.168.67.1
192.0.2.6/32	Swap	262135	262142BU	1/1/4	192.168.57.1

This default LFA NH can be changed by adding specific selection criteria inside a route-next-hop policy template.

Example 1: admin-group

The objective is to force the LFA NH for both LDP FEC prefixes to use the path between PE-7 and PE-8.

Define admin-group 'red' with value '1' and apply it on the IP interfaces PE-7 to PE-5 and PE-7 to PE-6.

```
*A:PE-7# configure router if-attribute admin-group "red" value 1

*A:PE-7# configure router interface "int-PE-7-PE-5" if-attribute admin-group "red"
*A:PE-7# configure router interface "int-PE-7-PE-6" if-attribute admin-group "red"
```

Define a route-next-hop policy template 'example1', which excludes IP admin-group 'red'.

```
*A:PE-7# configure router route-next-hop-policy
*A:PE-7>config>router>route-nh# info
    begin
    template "example1"
        exclude-group "red"
    exit
    commit
```

From the moment that route-next-hop policy template 'example1' is applied to the OSPF interfaces towards PE-5 and PE-6, the LFA NHs for both LDP FEC prefixes change. They now both point to the PE-7 to PE-8 IP interface as LFA backup NH:

```
*A:PE-7# configure router ospf area 0 interface "int-PE-7-PE-5" lfa-policy-map route-nh-template "example1"
*A:PE-7# configure router ospf area 0 interface "int-PE-7-PE-6" lfa-policy-map route-nh-template "example1"
```

```
*A:PE-7# show router ldp bindings active
=====
Legend:  (S) - Static          (M) - Multi-homed Secondary Support
          (B) - BGP Next Hop  (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP Prefix Bindings (Active)
=====
```

Prefix	Op	IngLbl	EgrLbl	EgrIntf/LspId	EgrNextHop
192.0.2.5/32	Push	--	262143	1/1/4	192.168.57.1
192.0.2.5/32	Push	--	262142BU	1/1/5	192.168.78.2
192.0.2.5/32	Swap	262136	262143	1/1/4	192.168.57.1
192.0.2.5/32	Swap	262136	262142BU	1/1/5	192.168.78.2
192.0.2.6/32	Push	--	262143	1/1/3	192.168.67.1
192.0.2.6/32	Push	--	262141BU	1/1/5	192.168.78.2
192.0.2.6/32	Swap	262135	262143	1/1/3	192.168.67.1
192.0.2.6/32	Swap	262135	262141BU	1/1/5	192.168.78.2

Example 2: SRLG

The objective is to force the LFA NH for both LDP FEC prefixes to use the PE-7 to PE-8 path.

Define SRLG group 'blue' with value '2' and apply it to the IP interfaces PE-7 to PE-5 and PE-7 to PE-6.

```
*A:PE-7# configure router if-attribute srlg-group "blue" value 2

*A:PE-7# configure router interface "int-PE-7-PE-5" if-attribute srlg-group "blue"
*A:PE-7# configure router interface "int-PE-7-PE-6" if-attribute srlg-group "blue"
```

Define a route-next-hop policy template 'example2', where SRLG is enabled

```
*A:PE-7# configure router route-next-hop-policy
*A:PE-7>config>router>route-nh# info
    begin
    template "example2"
        srlg-enable
    exit
    commit
```

From the moment that route-next-hop policy template 'example2' is applied to the OSPF interfaces towards PE-5 and PE-6, the LFA NHs for both LDP FEC prefixes change. They will both point now to the PE-7 to PE-8 interface as LFA backup NH:

```
*A:PE-7# configure router ospf area 0 interface "int-PE-7-PE-5" lfa-policy-map route-nh-
template "example2"
*A:PE-7# configure router ospf area 0 interface "int-PE-7-PE-6" lfa-policy-map route-nh-
template "example2"
```

```
*A:PE-7# show router ldp bindings active
=====
Legend:  (S) - Static          (M) - Multi-homed Secondary Support
          (B) - BGP Next Hop  (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP Prefix Bindings (Active)
=====
Prefix                Op    IngLbl    EgrLbl    EgrIntf/LspId  EgrNextHop
-----
192.0.2.5/32          Push  --        262143    1/1/4          192.168.57.1
192.0.2.5/32          Push  --        262142BU   1/1/5          192.168.78.2
192.0.2.5/32          Swap  262136    262143    1/1/4          192.168.57.1
192.0.2.5/32          Swap  262136    262142BU   1/1/5          192.168.78.2
192.0.2.6/32          Push  --        262143    1/1/3          192.168.67.1
192.0.2.6/32          Push  --        262141BU   1/1/5          192.168.78.2
192.0.2.6/32          Swap  262135    262143    1/1/3          192.168.67.1
192.0.2.6/32          Swap  262135    262141BU   1/1/5          192.168.78.2
```

Example 3: NH-type

The objective is to force the LFA NH for IP prefix 192.0.2.6/32 to use an RSVP tunnel.

Enable IP FRR and setup an RSVP LSP tunnel¹ towards 192.0.2.6 with a strict MPLS path going over PE-7 to PE-9 to PE-8 to PE-6.

```
*A:PE-7# configure router ip-fast-reroute

*A:PE-7# configure router mpls
      interface "system"
        no shutdown
      exit
      interface "int-PE-7-PE-9"
        no shutdown
      exit
      path "P-PE-7-PE-9-PE-8-PE6"
        hop 10 192.168.79.2 strict
        hop 20 192.168.89.1 strict
        hop 30 192.168.68.1 strict
        no shutdown
      exit
      lsp "LSP-PE-7-PE-6"
        to 192.0.2.6
        primary "P-PE-7-PE-9-PE-8-PE6"
        exit
        no shutdown
      exit
    no shutdown
```

Enable RSVP shortcut within the IGP on PE-7 and indicate that the newly created RSVP LSP is a possible shortcut candidate for LFA backup NH only.

```
*A:PE-7# configure router ospf rsvp-shortcut

*A:PE-7# configure router mpls lsp "LSP-PE-7-PE-6" igp-shortcut lfa-only
```

Displaying the tunnel-table of PE-7 shows that an LDP LSP and an RSVP LSP is available towards PE-6:

```
*A:PE-7# show router tunnel-table 192.0.2.6
=====
Tunnel Table (Router: Base)
=====
Destination          Owner Encap TunnelId  Pref    Nexthop      Metric
-----
192.0.2.6/32         rsvp  MPLS   1          7       192.168.79.2 16777215
192.0.2.6/32         ldp   MPLS   -          9       192.168.67.1 1000
```

1. Since an RSVP LSP is setup between PE-7 and PE-6, MPLS/RSVP protocols also need to be enabled on all the corresponding IP interfaces along the MPLS path.

```
*A:PE-7# show router mpls lsp
=====
MPLS LSPs (Originating)
=====
LSP Name                               To                Tun    Fastfail  Adm  Opr
                               Id                Config
-----
LSP-PE-7-PE-6                        192.0.2.6         1       No        Up   Up
-----
LSPs : 1
=====
*A:PE-7#
```

```
*A:PE-7# show router route-table alternative 192.0.2.6/32
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                Type    Proto    Age          Pref
      Next Hop[Interface Name]          Metric
      Alt-NextHop                      Alt-
                                          Metric
-----
192.0.2.6/32                      Remote  OSPF      00h02m44s    10
      192.168.67.1                      1000
      192.168.57.1 (LFA)                 2000
-----
*A:PE-7#
```

Define a route-next-hop policy template **example3**, where nh-type is set to **tunnel**.

```
*A:PE-7# configure router route-next-hop-policy
*A:PE-7>config>router>route-nh# info
      begin
      template "example3"
      nh-type tunnel
      exit
      commit
```

From the moment that route-next-hop policy template **example3** is applied to the OSPF interface towards PE-6, the LFA NH uses the RSVP tunnel. Note that the reference to the RSVP tunnel-ID (1) in the following show output corresponds with the tunnel-ID shown in the previous **show router tunnel-table 192.0.2.6** output:

```
*A:PE-7# configure router ospf area 0 interface "int-PE-7-PE-6" lfa-policy-map route-nh-
template "example3"
```

```
*A:PE-7# show router route-table alternative 192.0.2.6/32
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                Type    Proto    Age          Pref
      Next Hop[Interface Name]          Metric
      Alt-NextHop                      Alt-
                                          Metric
-----
```

Examples

```
192.0.2.6/32                                Remote  OSPF      00h01m49s  10
      192.168.67.1                          1000
      192.0.2.6 (LFA) (tunneled:RSVP:1)      65535
-----
*A:PE-7#

*A:PE-7# show router fib 1 nh-table-usage
=====
FIB Next-Hop Summary
=====
IPv4/IPv6                                Active                                Available
-----
IP Next-Hop                             10                                16383
Tunnel Next-Hop                          1                                993279
=====
*A:PE-7#
```

Example 4: Exclude Prefix

The objective is to force no LFA NH for LDP FEC prefix 192.0.2.5/32 where PE-7 is the PLR.

From the introduction of IP/LDP FRR implementation in SR-OS, it is possible to exclude an IGP interface, IGP area (OSPF) or IGP level (IS-IS) from the LFA SPF computation. The user also has the ability to exclude specific prefixes from the LFA SPF by using well-known prefix-lists and policy statements.

Translated into configuration commands, this becomes:

```
A:PE-7# configure router policy-options
      begin
      prefix-list "lo0-PE-5"
        prefix 192.0.2.5/32 exact
      exit
      policy-statement "PE-5-exclude-LFA"
        entry 10
          from
            prefix-list "lo0-PE-5"
          exit
          action accept
          exit
        exit
      exit
    exit
  commit
```

The configured policy statement is applied to the IGP protocol. From the moment that it is applied, the existing LFA NH entries for LDP FEC prefix 192.0.2.5/32 disappear instantly (compare with Example 1 above):

```
*A:PE-7# configure router ospf loopfree-alternate-exclude prefix-policy "PE-5-exclude-LFA"
```

```
*A:PE-7# show router ldp bindings active prefix 192.0.2.5/32
```

```
=====
Legend:  (S) - Static          (M) - Multi-homed Secondary Support
         (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP Prefix Bindings (Active)
=====
Prefix                Op   IngLbl  EgrLbl  EgrIntf/LspId  EgrNextHop
-----
192.0.2.5/32          Push  --      262143  1/1/4          192.168.57.1
192.0.2.5/32          Swap 262136  262143  1/1/4          192.168.57.1
-----
No. of Prefix Active Bindings: 2
=====
```

Conclusion

In production MPLS networks where IP FRR and/or LDP FRR is deployed it is possible that the existing calculated LFA NHs are not always taking the most optimal or desirable paths.

With LFA policies, operators have better control on the way in which LFA backup NHs are computed.

Different selection criteria can be part of the route-next-hop policy: IP admin-groups, IP SRLG groups, protection type preference and NH type preference.

Routing Protocols

In This Section

This section provides configuration information for the following topics:

- [Associating Communities with Static and Aggregate Routes on page 269](#)
- [IS-IS Link Bundling on page 301](#)

Associating Communities with Static and Aggregate Routes

In This Chapter

This section provides information about associating communities with static and aggregate routes configurations.

Topics in this section include:

- [Applicability on page 270](#)
- [Overview on page 272](#)
- [Configuration on page 274](#)
- [Conclusion on page 300](#)

Applicability

This example is applicable to all the 7750 SR, 7450 ESS in mixed-mode and 7950 XRS series and was tested on release 12.0.R1. There are no pre-requisites for this configuration.

Introduction

Border Gateway Protocol (BGP) Communities are optional, transitive attributes attached to BGP route prefixes to carry additional information about that route prefix. A number of route prefixes can have the same community attached such that it can be matched by a route policy. As a result, the presence of a community value can be used to influence and control route policy.

A BGP community is a 32-bit value that is written as two separate 16 bit numbers separated by a colon. The first number usually represents the Autonomous System (AS) number that defines or originates the community whilst the second is set by the network administrator.

Knowledge of RFC 4271 (BGP-4) and RFC 1997 (BGP Communities Attribute) is assumed throughout this document, as well as knowledge of Multi-Protocol BGP (MP-BGP) and RFC 4364 (BGP/MPLS IP VPNs).

Overview

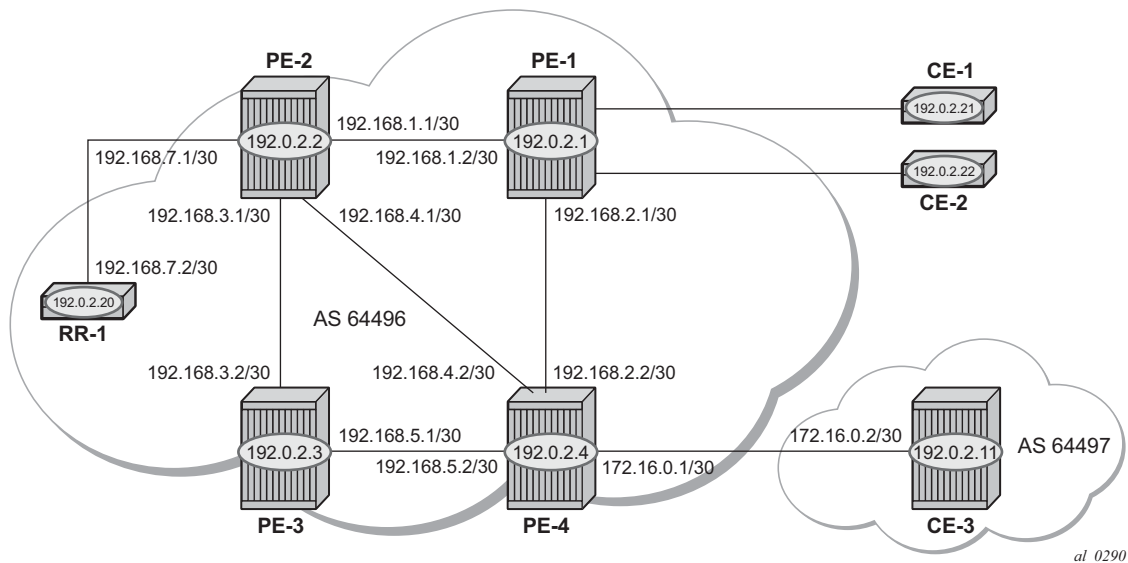


Figure 54: Network Topology

The network topology is displayed in [Figure 54](#). The setup uses 7750/7450/7710 Service Router (SR) nodes. PE-1 to PE-4 and the Route Reflector (RR-1) are located in the same Autonomous System (AS); AS6696. CE-3 is in a separate AS 64497 and peers using eBGP with its directly connected neighbor, PE-4.

The objectives are:

- To configure static-routes in a VPRN in PE-1 with various community values – including well-known communities – export them to other PEs within the same AS, and then via eBGP to CE-3. During this process, the community values for each route will be examined to ensure that the transitive nature of the attribute is maintained.
- To associate a community with an aggregate route that represents a larger number of composite prefixes. The aggregate will be advertised in place of the composite prefixes.

The following configuration tasks should be completed as a pre-requisite:

- Full mesh ISIS or OSPF between each of the PE routers and route reflector.
- iBGP between the RR and all PEs.
- eBGP between PE-4 and CE-3.
- Link-layer LDP between each PE.

Associating Communities with Static and Aggregate Routes

It is possible to add a single community value to a static and aggregate route without using a route policy.

The community value can be in the 4-byte format comprising of a 2-byte AS value, followed by a 2 byte decimal value, separated by a colon. It can also be the name of a well-known standard community; no-export, no-advertise, no-export-subconfed.

Any community added can be matched using a route policy.

The purpose of this example is to provision static and aggregate IPv4 route prefixes and associate a community with each route. These routes are then redistributed into the BGP protocol and advertised to other BGP speakers.

This is shown for IPv4 routes within a VPRN. Well-known, standard communities will also be configured to show that the correct behavior is observed.

Configuration

The first step is to configure an iBGP session between each of the PEs and the Route Reflector (RR). The address family negotiated between peers is vpn-ipv4.

The configuration for PE-1 is:

```
configure router bgp
  group internal
    family vpn-ipv4
    type internal
    neighbor 192.0.2.20
  exit
exit
exit all
```

The configuration for the other PEs is very similar. The IP addresses can be derived from [Figure 54](#).

The configuration for the RR is:

```
configure router bgp
  cluster 0.0.0.1
  group rr_clients
    type internal
    family vpn-ipv4
    neighbor 192.0.2.1
  exit
    neighbor 192.0.2.2
  exit
    neighbor 192.0.2.3
  exit
    neighbor 192.0.2.4
  exit
exit
exit all
```

On RR-1, show that BGP sessions with each PE are established, and have correctly negotiated the VPN IPv4 address family capability.

A:RR-1# show router bgp summary

```
=====
BGP Router ID:192.0.2.20      AS:64496      Local AS:64496
=====
BGP Admin State      : Up      BGP Oper State      : Up
Total Peer Groups    : 1      Total Peers          : 4
Total BGP Paths       : 18     Total Path Memory    : 3336
Total IPv4 Remote Rts : 0      Total IPv4 Rem. Active Rts : 0
Total McIPv4 Remote Rts : 0     Total McIPv4 Rem. Active Rts : 0
Total McIPv6 Remote Rts : 0     Total McIPv6 Rem. Active Rts : 0
Total IPv6 Remote Rts : 0      Total IPv6 Rem. Active Rts : 0
Total IPv4 Backup Rts : 0      Total IPv6 Backup Rts : 0
```

Associating Communities with Static and Aggregate Routes

```

Total Supressed Rts      : 0          Total Hist. Rts          : 0
Total Decay Rts          : 0

Total VPN Peer Groups    : 0          Total VPN Peers          : 0
Total VPN Local Rts      : 0
Total VPN-IPv4 Rem. Rts  : 8          Total VPN-IPv4 Rem. Act. Rts: 0
Total VPN-IPv6 Rem. Rts  : 0          Total VPN-IPv6 Rem. Act. Rts: 0
Total VPN-IPv4 Bkup Rts  : 0          Total VPN-IPv6 Bkup Rts   : 0

Total VPN Supp. Rts      : 0          Total VPN Hist. Rts      : 0
Total VPN Decay Rts      : 0

Total L2-VPN Rem. Rts    : 0          Total L2VPN Rem. Act. Rts : 0
Total MVPN-IPv4 Rem Rts  : 0          Total MVPN-IPv4 Rem Act Rts : 0
Total MDT-SAFI Rem Rts   : 0          Total MDT-SAFI Rem Act Rts  : 0
Total MSPW Rem Rts       : 0          Total MSPW Rem Act Rts     : 0
Total RouteTgt Rem Rts   : 0          Total RouteTgt Rem Act Rts  : 0
Total McVpnIPv4 Rem Rts  : 0          Total McVpnIPv4 Rem Act Rts : 0
Total MVPN-IPv6 Rem Rts  : 0          Total MVPN-IPv6 Rem Act Rts : 0
Total EVPN Rem Rts       : 0          Total EVPN Rem Act Rts     : 0
Total FlowIpv4 Rem Rts   : 0          Total FlowIpv4 Rem Act Rts  : 0
Total FlowIpv6 Rem Rts   : 0          Total FlowIpv6 Rem Act Rts  : 0

```

=====

BGP Summary

Neighbor

	AS	PktRcvd	InQ	Up/Down	State	Rcv/Act/Sent	(Addr Family)
		PktSent	OutQ				
192.0.2.1							
	64496	2089	0	17h20m25s	7/0/8	(VpnIPv4)	
		2091	0				
192.0.2.2							
	64496	2083	0	17h20m16s	0/0/8	(VpnIPv4)	
		2091	0				
192.0.2.3							
	64496	2082	0	17h19m44s	0/0/8	(VpnIPv4)	
		2089	0				
192.0.2.4							
	64496	2084	0	17h20m19s	1/0/8	(VpnIPv4)	
		2091	0				

A:RR-1#

VPRN: IPv4

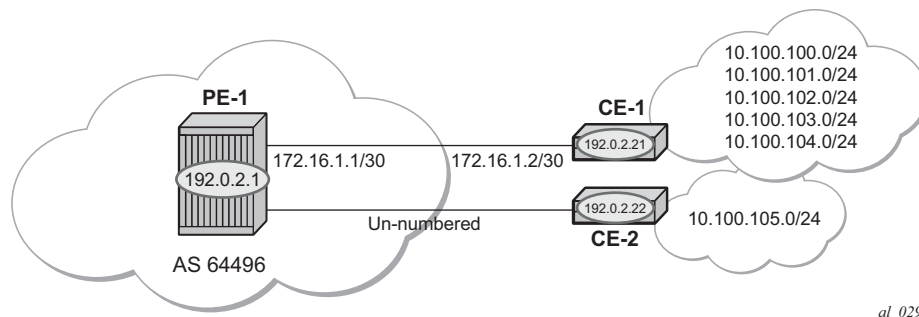


Figure 55: CE Connections for Next-Hops

The VPRN configuration for PE-1 is shown below:

```
A:PE-1# configure service vprn 3
-----
route-distinguisher 64496:3
auto-bind ldp
vrf-target target:64496:3
interface "int-PE-1-CE-1" create
  address 172.16.1.1/30
  sap 1/2/1:1.0 create
  exit
exit
interface "int-PE-1-CE-2" create
  unnumbered "loop1"
  sap 1/2/2:1.0 create
  exit
exit
interface "loop1" create
  address 192.0.2.100/32
  loopback
exit
```

LDP is used as the label-switching protocol for next-hop resolution.

The configuration is very similar for the other PEs.

PE-4 is configured with an interface towards CE-3 that supports eBGP, as follows:

```
A:PE-4# configure service vprn 3
A:PE-4>config>service>vprn# info
-----
autonomous-system 64496
route-distinguisher 64496:3
auto-bind ldp
vrf-target target:64496:3
```


Associating Communities with Static and Aggregate Routes

```
interface "int-PE-4-CE-3" create
    address 172.16.0.1/30
    sap 1/1/4:3 create
    exit
exit
bgp
    export "PE-4-VPN-BGP"
    group "VPRN-3-ext"
        peer-as 64497
        neighbor 172.16.0.2
    exit
    exit
    no shutdown
exit
no shutdown
```

Static Routes with Communities

A static route has a number of next-hop options – direct connected IP address, black-hole, indirect IP address and interface-name.

Figure 55 shows a pair of Customer Edge (CE) routers connected to PE1. The link to CE-1 is a numbered link. The link to CE-2 is an un-numbered link. The loopback interface address is used as a reference address for the un-numbered Ethernet interface.

Beyond CE-1 are a number of /24 subnets. Static routes to these individual subnets are created on PE-1 using a static route with a next-hop type of “interface address” or an “indirect address”. The indirect address is learned using a static route.

Beyond CE-2 is a single /24 subnet. A static route to this subnet is created using an interface-name next-hop type.

There are a number of well-known, standard communities:

- no-export: the route is not advertised to any external peer. This should be observed in the route tables of all BGP speakers in the originating AS, but not in those in neighboring ASs.
- no-advertise: the route is not advertised to any peer. This should not be observed in any router as BGP-learned route.

The requirement for each subnet is

- 10.100.100.0/24 must not be advertised outside of the AS. This must be associated with the standard, well-known community no-export. The community value is encoded as 65535:65281 (0xFFFFF01), but the CLI requires the keyword **no-export**.

```
A:PE-1>config>service vprn 3
      static-route 10.100.100.0/24 next-hop 172.16.1.2 community no-export
```

- 10.100.101.0/24 must be advertised with a community of 64496:101

```
A:PE-1>config> service vprn 3
      static-route 10.100.101.0/24 next-hop 172.16.1.2 community 64496:101
```

- 10.100.102.0/24 must not be advertised to any BGP peer. This must be associated with the standard, well-known community **no-advertise**. The community value is encoded as 65535:65282 (0xFFFFF02), but the CLI requires the keyword **no-advertise**.

```
A:PE-1>config> service vprn 3
      static-route 10.100.102.0/24 next-hop 172.16.1.2 community no-advertise
```

- 10.100.103.0/24 must be advertised with a community of 64496:103 and a route tag of 10.

Associating Communities with Static and Aggregate Routes

```
A:PE-1>config> service vprn 3
    static-route 10.100.103.0/24 next-hop 172.16.1.2 tag 10 community 64496:103
```

- 10.100.104.0/24 must be advertised with a community of 64496:104. It is reachable via 192.0.2.21 which, in turn, is reachable via 172.16.1.2. This is using a static route which does not need to be advertised – hence it is associated with the **no-advertise** community.

```
A:PE-1>config> service vprn 3
    static-route 10.100.104.0/24 indirect 192.0.2.21 community 64496:104
    static-route 192.0.2.21/32 next-hop 172.16.1.2 community no-advertise
```

- 10.100.105.0/24 must be advertised with a community of 64496:105. It is reachable via the un-numbered interface to CE-2.

```
A:PE-1>config> service vprn 3
    static-route 10.100.105.0/24 next-hop "int-PE-1-CE-2" community 64496:105
```

On PE-1 configure static routes that match the static routes from [Figure 55](#), and the conditions from above.

Note that the default behavior of a VPRN is to export all static and connected routes into a BGP labelled route with the appropriate route-target extended community configured in the vrf-target statement. A single community string can be added using the static-route community commands shown above. If multiple communities are required, then a VRF-export policy should be used. This is outside the scope of this note.

Examine the BGP table of PE-1 to establish that routes have been exported correctly in VPN IPv4 towards RR-1.

```
A:PE-1# show router bgp neighbor 192.0.2.20 advertised-routes vpn-ipv4
=====
BGP Router ID:192.0.2.1      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP VPN-IPv4 Routes
=====
Flag  Network                               LocalPref  MED
      Nexthop                             Path-Id    Label
      As-Path
-----
i     64496:3:10.100.100.0/24              100        None
      192.0.2.1                          None       262142
      No As-Path
i     64496:3:10.100.101.0/24              100        None
      192.0.2.1                          None       262142
      No As-Path
i     64496:3:10.100.103.0/24              100        None
      192.0.2.1                          None       262142
```

Static Routes with Communities

```

        No As-Path
i      64496:3:10.100.104.0/24          100      None
        192.0.2.1                      None      262142
        No As-Path
i      64496:3:10.100.105.0/24          100      None
        192.0.2.1                      None      262142
        No As-Path
i      64496:3:172.16.1.0/30           100      None
        192.0.2.1                      None      262142
        No As-Path
i      64496:3:192.0.2.100/32          100      None
        192.0.2.1                      None      262142
        No As-Path

```

```
-----
Routes : 7
=====
```

```
A:PE-1#
```

Note that there are only seven exported routes. The route prefixes associated with the **no-advertise** community are not present, as expected.

Examining the BGP table of PE-4 shows the presence of the expected routes, with the correct community values.

The prefix 10.100.100.0/24 is a member of community **no-export**. This is correctly advertised to PE-4.

```
A:PE-4# show router bgp routes vpn-ipv4 10.100.100.0/24 detail
```

```
=====
BGP Router ID:192.0.2.4      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP VPN-IPv4 Routes
=====
-----
Original Attributes

Network      : 10.100.100.0/24
Nexthop      : 192.0.2.1
Route Dist.  : 64496:3          VPN Label      : 262142
Path Id      : None
From         : 192.0.2.20
Res. Nexthop : n/a
Local Pref.  : 100
Aggregator AS : None           Interface Name : int-PE-4-PE-1
Atomic Aggr. : Not Atomic      Aggregator    : None
AIGP Metric  : None           MED           : None
Connector    : None
Community    : no-export target:64496:3
Cluster      : 0.0.0.1
Originator Id : 192.0.2.1      Peer Router Id : 192.0.2.20
Fwd Class    : None           Priority       : None
Flags        : Used Valid Best IGP

```

Associating Communities with Static and Aggregate Routes

```
Route Source      : Internal
AS-Path           : No As-Path
Neighbor-AS       : N/A
VPRN Imported     : 3
```

Modified Attributes

```
Network           : 10.100.100.0/24
Nexthop           : 192.0.2.1
Route Dist.       : 64496:3          VPN Label       : 262142
Path Id           : None
From              : 192.0.2.20
Res. Nexthop      : n/a
Local Pref.       : 100              Interface Name  : int-PE-4-PE-1
Aggregator AS     : None             Aggregator      : None
Atomic Aggr.      : Not Atomic       MED             : None
AIGP Metric       : None
Connector         : None
Community         : no-export target:64496:3
Cluster           : 0.0.0.1
Originator Id     : 192.0.2.1        Peer Router Id  : 192.0.2.20
Fwd Class         : None             Priority         : None
Flags             : Used Valid Best IGP
Route Source      : Internal
AS-Path           : No As-Path
Neighbor-AS       : N/A
VPRN Imported     : 3
```

```
-----
Routes : 1
=====
```

```
A:PE-4#
```

The prefix 10.100.101.0/24 is a member of community 64496:101. This is correctly advertised to PE-4.

```
A:PE-4# show router bgp routes vpn-ipv4 10.100.101.0/24 detail
```

```
=====
BGP Router ID:192.0.2.4      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
```

```
=====
BGP VPN-IPv4 Routes
=====
```

Original Attributes

```
Network           : 10.100.101.0/24
Nexthop           : 192.0.2.1
Route Dist.       : 64496:3          VPN Label       : 262142
Path Id           : None
From              : 192.0.2.20
Res. Nexthop      : n/a
Local Pref.       : 100              Interface Name  : int-PE-4-PE-1
Aggregator AS     : None             Aggregator      : None
```

Static Routes with Communities

```
Atomic Aggr.      : Not Atomic          MED           : None
AIGP Metric       : None
Connector         : None
Community         : 64496:101 target:64496:3
Cluster           : 0.0.0.1
Originator Id     : 192.0.2.1           Peer Router Id : 192.0.2.20
Fwd Class         : None                Priority       : None
Flags             : Used Valid Best IGP
Route Source      : Internal
AS-Path           : No As-Path
Neighbor-AS       : N/A
VPRN Imported     : 3
```

Modified Attributes

```
Network           : 10.100.101.0/24
Nexthop           : 192.0.2.1
Route Dist.       : 64496:3            VPN Label      : 262142
Path Id           : None
From              : 192.0.2.20
Res. Nexthop      : n/a
Local Pref.       : 100                Interface Name : int-PE-4-PE-1
Aggregator AS     : None                Aggregator     : None
Atomic Aggr.      : Not Atomic          MED           : None
AIGP Metric       : None
Connector         : None
Community         : 64496:101 target:64496:3
Cluster           : 0.0.0.1
Originator Id     : 192.0.2.1           Peer Router Id : 192.0.2.20
Fwd Class         : None                Priority       : None
Flags             : Used Valid Best IGP
Route Source      : Internal
AS-Path           : No As-Path
Neighbor-AS       : N/A
VPRN Imported     : 3
```

```
-----
Routes : 1
=====
```

```
A:PE-4#
```

The prefix 10.100.103.0/24 is a member of community 64496:103. This is correctly advertised to PE-4.

```
A:PE-4# show router bgp routes vpn-ipv4 10.100.103.0/24 detail
```

```
=====
BGP Router ID:192.0.2.4      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP VPN-IPv4 Routes
=====
-----
Original Attributes
```

Associating Communities with Static and Aggregate Routes

```

Network      : 10.100.103.0/24
Nexthop      : 192.0.2.1
Route Dist.  : 64496:3          VPN Label      : 262142
Path Id      : None
From         : 192.0.2.20
Res. Nexthop : n/a
Local Pref.  : 100              Interface Name : int-PE-4-PE-1
Aggregator AS : None            Aggregator    : None
Atomic Aggr. : Not Atomic       MED            : None
AIGP Metric  : None
Connector    : None
Community    : 64496:103 target:64496:3
Cluster      : 0.0.0.1
Originator Id : 192.0.2.1       Peer Router Id : 192.0.2.20
Fwd Class    : None            Priority       : None
Flags        : Used Valid Best IGP
Route Source : Internal
AS-Path      : No As-Path
Neighbor-AS  : N/A
VPRN Imported : 3

```

Modified Attributes

```

Network      : 10.100.103.0/24
Nexthop      : 192.0.2.1
Route Dist.  : 64496:3          VPN Label      : 262142
Path Id      : None
From         : 192.0.2.20
Res. Nexthop : n/a
Local Pref.  : 100              Interface Name : int-PE-4-PE-1
Aggregator AS : None            Aggregator    : None
Atomic Aggr. : Not Atomic       MED            : None
AIGP Metric  : None
Connector    : None
Community    : 64496:103 target:64496:3
Cluster      : 0.0.0.1
Originator Id : 192.0.2.1       Peer Router Id : 192.0.2.20
Fwd Class    : None            Priority       : None
Flags        : Used Valid Best IGP
Route Source : Internal
AS-Path      : No As-Path
Neighbor-AS  : N/A
VPRN Imported : 3

```

```

-----
Routes : 1

```

```

=====
A:PE-4#

```

The prefix 10.100.104.0/24 is a member of community 64496:104. This is correctly advertised to PE-4.

```

A:PE-4# show router bgp routes vpn-ipv4 10.100.104.0/24 detail

```

```

=====
BGP Router ID:192.0.2.4      AS:64496      Local AS:64496
=====
Legend -

```

Static Routes with Communities

```
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP VPN-IPv4 Routes
=====
-----
Original Attributes

Network      : 10.100.104.0/24
Nexthop      : 192.0.2.1
Route Dist.  : 64496:3          VPN Label      : 262142
Path Id      : None
From         : 192.0.2.20
Res. Nexthop : n/a
Local Pref.  : 100              Interface Name : int-PE-4-PE-1
Aggregator AS : None           Aggregator    : None
Atomic Aggr. : Not Atomic      MED           : None
AIGP Metric  : None
Connector    : None
Community    : 64496:104 target:64496:3
Cluster      : 0.0.0.1
Originator Id : 192.0.2.1      Peer Router Id : 192.0.2.20
Fwd Class    : None           Priority       : None
Flags        : Used Valid Best IGP
Route Source : Internal
AS-Path      : No As-Path
Neighbor-AS  : N/A
VPRN Imported : 3

Modified Attributes

Network      : 10.100.104.0/24
Nexthop      : 192.0.2.1
Route Dist.  : 64496:3          VPN Label      : 262142
Path Id      : None
From         : 192.0.2.20
Res. Nexthop : n/a
Local Pref.  : 100              Interface Name : int-PE-4-PE-1
Aggregator AS : None           Aggregator    : None
Atomic Aggr. : Not Atomic      MED           : None
AIGP Metric  : None
Connector    : None
Community    : 64496:104 target:64496:3
Cluster      : 0.0.0.1
Originator Id : 192.0.2.1      Peer Router Id : 192.0.2.20
Fwd Class    : None           Priority       : None
Flags        : Used Valid Best IGP
Route Source : Internal
AS-Path      : No As-Path
Neighbor-AS  : N/A
VPRN Imported : 3

-----
Routes : 1
=====
A:PE-4#
```


Associating Communities with Static and Aggregate Routes

The prefix 10.100.105.0/24 is a member of community 64496:105. This is correctly advertised to PE-4.

```
A:PE-4# show router bgp routes vpn-ipv4 10.100.105.0/24 detail
=====
BGP Router ID:192.0.2.4          AS:64496          Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP VPN-IPv4 Routes
=====
-----
Original Attributes

Network      : 10.100.105.0/24
Nexthop      : 192.0.2.1
Route Dist.  : 64496:3          VPN Label      : 262142
Path Id      : None
From         : 192.0.2.20
Res. Nexthop : n/a
Local Pref.  : 100              Interface Name : int-PE-4-PE-1
Aggregator AS : None           Aggregator     : None
Atomic Aggr. : Not Atomic      MED            : None
AIGP Metric  : None
Connector    : None
Community    : 64496:105 target:64496:3
Cluster      : 0.0.0.1
Originator Id : 192.0.2.1      Peer Router Id : 192.0.2.20
Fwd Class    : None           Priority        : None
Flags        : Used Valid Best IGP
Route Source : Internal
AS-Path      : No As-Path
Neighbor-AS  : N/A
VPRN Imported : 3

Modified Attributes

Network      : 10.100.105.0/24
Nexthop      : 192.0.2.1
Route Dist.  : 64496:3          VPN Label      : 262142
Path Id      : None
From         : 192.0.2.20
Res. Nexthop : n/a
Local Pref.  : 100              Interface Name : int-PE-4-PE-1
Aggregator AS : None           Aggregator     : None
Atomic Aggr. : Not Atomic      MED            : None
AIGP Metric  : None
Connector    : None
Community    : 64496:105 target:64496:3
Cluster      : 0.0.0.1
Originator Id : 192.0.2.1      Peer Router Id : 192.0.2.20
Fwd Class    : None           Priority        : None
Flags        : Used Valid Best IGP
Route Source : Internal
AS-Path      : No As-Path
Neighbor-AS  : N/A
```

Static Routes with Communities

```
VRPN Imported : 3
```

```
Routes : 1
```

```
A:PE-4#
```

Examine the route table of PE-4 – looking specifically at the BGP-learned routes, the same seven routes are present as valid routes.

```
A:PE-4# show router 3 route-table protocol bgp-vpn
```

```
Route Table (Service: 3)
```

Dest Prefix[Flags] Next Hop[Interface Name]	Type	Proto	Age Metric	Pref
10.100.100.0/24 192.0.2.1 (tunneled)	Remote	BGP VPN	17h24m48s 0	170
10.100.101.0/24 192.0.2.1 (tunneled)	Remote	BGP VPN	17h24m48s 0	170
10.100.103.0/24 192.0.2.1 (tunneled)	Remote	BGP VPN	17h24m48s 0	170
10.100.104.0/24 192.0.2.1 (tunneled)	Remote	BGP VPN	17h24m48s 0	170
10.100.105.0/24 192.0.2.1 (tunneled)	Remote	BGP VPN	17h24m48s 0	170
172.16.1.0/30 192.0.2.1 (tunneled)	Remote	BGP VPN	17h24m48s 0	170
192.0.2.100/32 192.0.2.1 (tunneled)	Remote	BGP VPN	17h24m48s 0	170

```
No. of Routes: 7
```

```
Flags: n = Number of times nexthop is repeated
```

```
B = BGP backup route available
```

```
L = LFA nexthop available
```

```
A:PE-4#
```

Examine the route table of CE-3 – looking specifically at the BGP-learned routes, six routes are present as valid routes, as expected.

```
A:CE-3# show router route-table protocol bgp
```

```
Route Table (Router: Base)
```

Dest Prefix[Flags] Next Hop[Interface Name]	Type	Proto	Age Metric	Pref
10.100.101.0/24 172.16.0.1	Remote	BGP	17h32m32s 0	170
10.100.103.0/24 172.16.0.1	Remote	BGP	17h32m32s 0	170
10.100.104.0/24 172.16.0.1	Remote	BGP	17h32m32s 0	170
10.100.105.0/24 172.16.0.1	Remote	BGP	17h32m32s 0	170
172.16.1.0/30	Remote	BGP	17h32m32s	170

Associating Communities with Static and Aggregate Routes

```

172.16.0.1 0
192.0.2.100/32 Remote BGP 17h32m32s 170
172.16.0.1 0
-----
No. of Routes: 6
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
=====
A:CE-3#

```

The prefix 10.100.100.0/24 is not received from PE-4 as it is a member of the **no-export** community.

```

A:CE-3# show router bgp routes community 64496:100
=====
BGP Router ID:192.0.2.11      AS:64497      Local AS:64497
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP IPv4 Routes
=====
Flag  Network                               LocalPref  MED
      Nexthop                             Path-Id    Label
      As-Path
-----
No Matching Entries Found
=====
A:CE-3#

```

Static route 10.100.101.0/24 is received with the correct community 64496:101.

```

A:CE-3# show router bgp routes community 64496:101
=====
BGP Router ID:192.0.2.11      AS:64497      Local AS:64497
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP IPv4 Routes
=====
Flag  Network                               LocalPref  MED
      Nexthop                             Path-Id    Label
      As-Path
-----
u*>i  10.100.101.0/24                       None       None
      172.16.0.1                          None       -
      64496
-----
Routes : 1
=====

```

Static Routes with Communities

A:CE-3#

Static route 10.100.103.0/24 is received with the correct community 64496:103.

```
A:CE-3# show router bgp routes community 64496:103
=====
BGP Router ID:192.0.2.11      AS:64497      Local AS:64497
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP IPv4 Routes
=====
Flag   Network                               LocalPref  MED
      Nexthop                             Path-Id    Label
      As-Path
-----
u*>i  10.100.103.0/24                       None       None
      172.16.0.1                         None       -
      64496
-----
Routes : 1
=====
A:CE-3#
```

Static route 10.100.104.0/24 is received with the correct community 64496:104.

```
A:CE-3# show router bgp routes community 64496:104
=====
BGP Router ID:192.0.2.11      AS:64497      Local AS:64497
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP IPv4 Routes
=====
Flag   Network                               LocalPref  MED
      Nexthop                             Path-Id    Label
      As-Path
-----
u*>i  10.100.104.0/24                       None       None
      172.16.0.1                         None       -
      64496
-----
Routes : 1
=====
A:CE-3#
```

Static route 10.100.105.0/24 is received with the correct community 64496:105.

```
A:CE-3# show router bgp routes community 64496:105
=====
BGP Router ID:192.0.2.11      AS:64497      Local AS:64497
```

Associating Communities with Static and Aggregate Routes

```
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP IPv4 Routes
=====
Flag   Network                               LocalPref  MED
      Nexthop                             Path-Id    Label
      As-Path
-----
u*>i   10.100.105.0/24                       None       None
      172.16.0.1                          None       -
      64496
-----
Routes : 1
=====
A:CE-3#
```

Aggregate Routes with Communities

An aggregate route can be configured to represent a larger number of prefixes. For example, a set of prefixes 10.101.0.0/24 to 10.101.8.0/24 can be represented as a single aggregate prefix of 10.101.0.0/21.

This is due to the fact that the third octet in the range 0 to 15 can be represented by the 8 bits 00000000 to 00000111. The first 5 bits of this octet are common, along with the previous 2 octets, giving a prefix where the first 21 bits are common. Therefore the aggregate can be written as 10.101.0.0/21.

In order to illustrate the configuration of an aggregate, consider following.

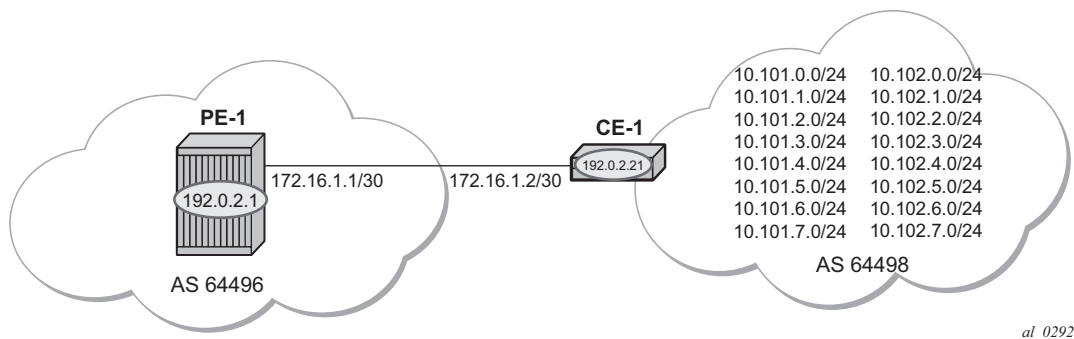


Figure 56: CE-1 Connectivity

Figure 56 shows a CE router (CE-1), in AS 64498, that advertises a series of contiguous prefixes via BGP.

- 10.101.0.0/24 to 10.101.7.0/24
- 10.102.0.0/24 to 10.102.7.0/24

Instead of advertising all of these prefixes out of the VPRN towards an external CE, an aggregate route can be configured that summarizes each set of 8 prefixes and a community can be directly associated with each aggregate route.

The configuration for a VPRN on PE-1, including the external BGP configuration is as follows:

```
B:PE-1>config>service>vprn# info
-----
autonomous-system 64496
route-distinguisher 64496:4
auto-bind mpls
vrf-target target:64496:4
interface "int-PE-1-CE-1" create
```

Associating Communities with Static and Aggregate Routes

```
address 172.16.1.1/30
sap 1/2/1:2.0 create
exit
exit
bgp
    group "external"
        family ipv4
        type external
        peer-as 64498
        neighbor 172.16.1.2
    exit
exit
no shutdown
exit
no shutdown
```

The neighbor relationship shows:

```
*A:PE-1# show router 4 bgp neighbor
```

```
=====
BGP Neighbor
=====
-----
Peer   : 172.16.1.2
Group  : external
-----
Peer AS           : 64498           Peer Port         : 179
Peer Address      : 172.16.1.2
Local AS          : 64496           Local Port        : 50709
Local Address     : 172.16.1.1
Peer Type         : External
State             : Established      Last State         : Active
Last Event        : recvKeepAlive
Last Error        : Unrecognized Error
Local Family      : IPv4
Remote Family     : IPv4
Hold Time         : 90               Keep Alive         : 30
Min Hold Time     : 0
Active Hold Time  : 90               Active Keep Alive  : 30
Cluster Id        : None
Preference        : 170              Num of Update Flaps : 0
Recd. Paths       : 1
IPv4 Recd. Prefixes : 16             IPv4 Active Prefixes : 16
VPN-IPv4 Suppressed Pfxs : 0         VPN-IPv4 Suppr. Pfxs : 0
VPN-IPv4 Recd. Pfxs : 0              VPN-IPv4 Active Pfxs : 0
Mc IPv4 Recd. Pfxs. : 0              Mc IPv4 Active Pfxs. : 0
Mc IPv4 Suppr. Pfxs : 0              IPv6 Suppressed Pfxs : 0
IPv6 Recd. Prefixes : 0              IPv6 Active Prefixes : 0
VPN-IPv6 Recd. Pfxs : 0              VPN-IPv6 Active Pfxs : 0
VPN-IPv6 Suppr. Pfxs : 0
Mc IPv6 Recd. Pfxs. : 0              Mc IPv6 Active Pfxs. : 0
Mc IPv6 Suppr. Pfxs : 0              L2-VPN Suppr. Pfxs  : 0
L2-VPN Recd. Pfxs  : 0              L2-VPN Active Pfxs  : 0
MVPN-IPv4 Suppr. Pfxs : 0            MVPN-IPv4 Recd. Pfxs : 0
MVPN-IPv4 Active Pfxs : 0            MDT-SAFI Suppr. Pfxs : 0
MDT-SAFI Recd. Pfxs : 0              MDT-SAFI Active Pfxs : 0
Flow-IPv4 Suppr. Pfxs : 0            Flow-IPv4 Recd. Pfxs : 0
Flow-IPv4 Active Pfxs : 0            Rte-Tgt Suppr. Pfxs : 0
```

Aggregate Routes with Communities

```

Rte-Tgt Recd. Pfxs      : 0
Backup IPv4 Pfxs        : 0
Mc Vpn Ipv4 Recd. Pf*   : 0
Backup Vpn IPv4 Pfxs    : 0
Input Queue             : 0
i/p Messages            : 10
i/p Octets               : 304
i/p Updates             : 1
MVPN-IPv6 Suppr. Pfxs   : 0
MVPN-IPv6 Active Pfxs   : 0
Flow-IPv6 Suppr. Pfxs   : 0
Flow-IPv6 Active Pfxs   : 0
Evpn Suppr. Pfxs        : 0
Evpn Active Pfxs        : 0
TTL Security            : Disabled
Graceful Restart         : Disabled
Restart Time             : n/a
Advertise Inactive       : Disabled
Advertise Label          : None
Auth key chain           : n/a
Disable Cap Nego         : Disabled
Flowspec Validate        : Disabled
Aigp Metric              : Disabled
Damp Peer Oscillatio*    : Disabled
GR Notification          : Disabled
Rem Idle Hold Time       : 00h00m00s
Next-Hop Unchanged       : None
Local Capability         : RtRefresh MPBGP 4byte ASN
Remote Capability        : RtRefresh MPBGP 4byte ASN
Local AddPath Capabi*    : Disabled
Remote AddPath Capab*    : Send - None
                        : Receive - None
Import Policy            : None Specified / Inherited
Export Policy            : None Specified / Inherited

```

```

-----
Neighbors : 1

```

```

=====
* indicates that the corresponding row element may have been truncated.

```

```

*A:PE-1#

```

The following output shows that 16 BGP routes are received by PE-1.

```

*A:PE-1# show router 4 bgp routes

```

```

=====
BGP Router ID:192.0.2.1      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP IPv4 Routes
=====
Flag  Network                      LocalPref  MED
      Nexthop                      Path-Id    Label
      As-Path
-----
u*>i  10.101.0.0/24                None       None

```


Associating Communities with Static and Aggregate Routes

```

172.16.1.2
64498
u*>i 10.101.1.0/24
172.16.1.2
64498
u*>i 10.101.2.0/24
172.16.1.2
64498
u*>i 10.101.3.0/24
172.16.1.2
64498
u*>i 10.101.4.0/24
172.16.1.2
64498
u*>i 10.101.5.0/24
172.16.1.2
64498
u*>i 10.101.6.0/24
172.16.1.2
64498
u*>i 10.101.7.0/24
172.16.1.2
64498
u*>i 10.102.0.0/24
172.16.1.2
64498
u*>i 10.102.1.0/24
172.16.1.2
64498
u*>i 10.102.2.0/24
172.16.1.2
64498
u*>i 10.102.3.0/24
172.16.1.2
64498
u*>i 10.102.4.0/24
172.16.1.2
64498
u*>i 10.102.5.0/24
172.16.1.2
64498
u*>i 10.102.6.0/24
172.16.1.2
64498
u*>i 10.102.7.0/24
172.16.1.2
64498
-----
Routes : 16
=====
*A:PE-1#

```

Aggregate Routes with Communities

PE-4 also has a VPRN 4 instance configured, so that it will receive the imported BGP routes. The configuration for PE-4 is:

```
A:PE-4>config>service>vprn# info
-----
autonomous-system 64496
route-distinguisher 64496:4
auto-bind mpls
vrf-target target:64496:4
interface "int-PE-4-CE-3" create
  address 172.16.0.5/30
  sap 1/1/4:4 create
  exit
exit
bgp
  group "VPRN-4-ext"
    peer-as 64497
    neighbor 172.16.0.6
    exit
  exit
  no shutdown
exit
no shutdown
```

Figure 57 shows the connectivity between PE-4 and CE-3. PE-4 will only forward a summarizing aggregate route towards CE-3.

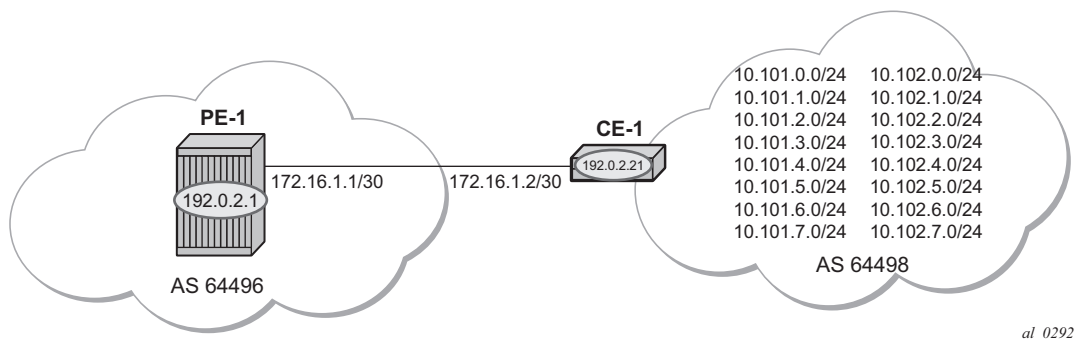


Figure 57: CE-3 Connectivity

PE-4 receives labelled BGP route prefixes from PE-1 via the route reflector and installs them in the FIB for router instance 4:

```
*A:PE-4# show router 4 route-table
=====
Route Table (Service: 4)
=====
Dest Prefix[Flags]                                Type      Proto    Age      Pref
      Next Hop[Interface Name]                                Metric
-----
```

Associating Communities with Static and Aggregate Routes

10.101.0.0/24	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	
10.101.1.0/24	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	
10.101.2.0/24	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	
10.101.3.0/24	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	
10.101.4.0/24	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	
10.101.5.0/24	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	
10.101.6.0/24	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	
10.101.7.0/24	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	
10.102.0.0/24	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	
10.102.1.0/24	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	
10.102.2.0/24	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	
10.102.3.0/24	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	
10.102.4.0/24	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	
10.102.5.0/24	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	
10.102.6.0/24	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	
10.102.7.0/24	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	
172.16.0.4/30	Local	Local		00h01m00s	0
int-PE-4-CE-3				0	
172.16.1.0/30	Remote	BGP	VPN	00h00m57s	170
192.0.2.1 (tunneled)				0	

No. of Routes: 18

Flags: n = Number of times nexthop is repeated
 B = BGP backup route available
 L = LFA nexthop available

=====

*A:PE-4#

The CE-3 configuration for an interface towards PE-4 is as follows:

```
*A:CE-3>config>service>ies# info
      interface "int-CE-3-PE-4-2" create
        address 172.16.0.6/30
        sap 1/1/2:4 create
      exit
exit
no shutdown
```

The BGP configuration of CE-3:

```
*A:CE-3>config>router>bgp# info
-----
      group "ext"
        peer-as 64496
        neighbor 172.16.0.5
      exit
exit
```

The BGP neighbor state for PE-4:

```
*A:PE-4# show router 4 bgp neighbor 172.16.0.6
=====
BGP Neighbor
=====
-----
Peer   : 172.16.0.6
Group  : VPRN-4-ext
-----
Peer AS           : 64497           Peer Port         : 179
Peer Address      : 172.16.0.6
Local AS          : 64496           Local Port        : 50539
Local Address     : 172.16.0.5
Peer Type         : External
State             : Established     Last State        : Active
Last Event        : recvKeepAlive
Last Error        : Unrecognized Error
Local Family      : IPv4
Remote Family     : IPv4
Hold Time         : 90              Keep Alive        : 30
Min Hold Time     : 0
Active Hold Time  : 90              Active Keep Alive  : 30
Cluster Id        : None
Preference        : 170             Num of Update Flaps : 0
Recd. Paths       : 0
IPv4 Recd. Prefixes : 0             IPv4 Active Prefixes : 0
IPv4 Suppressed Pfxs : 0            VPN-IPv4 Suppr. Pfxs : 0
VPN-IPv4 Recd. Pfxs : 0             VPN-IPv4 Active Pfxs : 0
Mc IPv4 Recd. Pfxs. : 0             Mc IPv4 Active Pfxs. : 0
Mc IPv4 Suppr. Pfxs : 0             IPv6 Suppressed Pfxs : 0
IPv6 Recd. Prefixes : 0             IPv6 Active Prefixes : 0
VPN-IPv6 Recd. Pfxs : 0             VPN-IPv6 Active Pfxs : 0
VPN-IPv6 Suppr. Pfxs : 0
Mc IPv6 Recd. Pfxs. : 0             Mc IPv6 Active Pfxs. : 0
```

Associating Communities with Static and Aggregate Routes

```

Mc IPv6 Suppr. Pfxs : 0
L2-VPN Recd. Pfxs : 0
MVPN-IPv4 Suppr. Pfxs: 0
MVPN-IPv4 Active Pfxs: 0
MDT-SAFI Recd. Pfxs : 0
Flow-IPv4 Suppr. Pfxs: 0
Flow-IPv4 Active Pfxs: 0
Rte-Tgt Recd. Pfxs : 0
Backup IPv4 Pfxs : 0
Mc Vpn Ipv4 Recd. Pf*: 0
Backup Vpn IPv4 Pfxs : 0
Input Queue : 0
i/p Messages : 4
i/p Octets : 102
i/p Updates : 0
MVPN-IPv6 Suppr. Pfxs: 0
MVPN-IPv6 Active Pfxs: 0
Flow-IPv6 Suppr. Pfxs: 0
Flow-IPv6 Active Pfxs: 0
Evpn Suppr. Pfxs : 0
Evpn Active Pfxs : 0
TTL Security : Disabled
Graceful Restart : Disabled
Restart Time : n/a
Advertise Inactive : Disabled
Advertise Label : None
Auth key chain : n/a
Disable Cap Nego : Disabled
Flowspec Validate : Disabled
Aigp Metric : Disabled
Damp Peer Oscillatio*: Disabled
GR Notification : Disabled
Rem Idle Hold Time : 00h00m00s
Next-Hop Unchanged : None
Local Capability : RtRefresh MPBGP 4byte ASN
Remote Capability : RtRefresh MPBGP 4byte ASN
Local AddPath Capabi*: Disabled
Remote AddPath Capab*: Send - None
: Receive - None
Import Policy : None Specified / Inherited
Export Policy : PE-4-VPN-Agg
-----
Neighbors : 1
=====
* indicates that the corresponding row element may have been truncated.
*A:PE-4#

```

In order to advertise a summarizing aggregate route with an associated community string, an aggregate route is required. In this case, the 10.101.x.0/24 group of prefixes will be associated with community 64496:101. The 10.102.x.0/24 group of prefixes will be associated with the standard community **no-export**, so that it will not be advertised to any external peer.

The configuration required is:

```

*A:PE-4>config>service>vprn#
    aggregate 10.101.0.0/21 community 64496:101

```

Aggregate Routes with Communities

```
aggregate 10.102.0.0/21 community no-export
```

An export policy is required to allow the advertising of the aggregate route. Note that no community is applied using this policy.

```
*A:PE-4>config>router>policy-options# begin
    policy-statement "PE-4-VPN-Agg"
        entry 10
            from
                protocol aggregate
            exit
            action accept
            exit
        exit
    exit
commit
```

This is applied as an export policy within the group context of the BGP configuration of the VPRN.

```
*A:PE-4>config>service>vprn#
    bgp
        group "VPRN-4-ext"
            export "PE-4-VPN-Agg"
        exit
    no shutdown
exit
```

The aggregate route 10.101.0.0/21 is received at CE-3 via BGP. The community that was associated with this prefix is seen – 64496:101. Note that the route is seen as an aggregate, with PE-4 as the aggregating router (192.0.2.4). Note also that the “Atomic Aggregate” attribute is present, meaning that PE-4 has not advertised any details of the AS Paths of the composite routes.

```
A:CE-3# show router bgp routes 10.101.0.0/21 hunt
=====
BGP Router ID:192.0.2.11      AS:64497      Local AS:64497
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP IPv4 Routes
=====
-----
RIB In Entries
-----
Network      : 10.101.0.0/21
Nexthop      : 172.16.0.5
Path Id      : None
From         : 172.16.0.5
Res. Nexthop : 172.16.0.5
Local Pref.  : None
Aggregator AS : 64496
Interface Name : int-CE-3-PE-4
Aggregator    : 192.0.2.4
```

Associating Communities with Static and Aggregate Routes

```
Atomic Aggr.      : Atomic                      MED              : None
AIGP Metric       : None
Connector         : None
Community         : 64496:101
Cluster           : No Cluster Members
Originator Id     : None                        Peer Router Id : 192.0.2.4
Fwd Class         : None                        Priority        : None
Flags             : Used Valid Best IGP
Route Source      : External
AS-Path           : 64496
Neighbor-AS       : 64496
```

```
-----
RIB Out Entries
-----
-----
```

```
Routes : 1
```

```
=====
A:CE-3#
```

The aggregate route 10.102.0.0/21 is not received at CE-3, as PE-4 does not advertise it, due to the fact that it is associated with the “no-export” community.

```
A:CE-3# show router bgp routes 10.102.0.0/21 hunt
```

```
=====
BGP Router ID:192.0.2.11      AS:64497      Local AS:64497
=====
```

```
Legend -
```

```
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
```

```
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
```

```
=====
BGP IPv4 Routes
=====
```

```
No Matching Entries Found
=====
```

```
A:CE-3#
```

Conclusion

Community strings can be added to static and aggregate routes. This example shows the configuration of communities with both static and aggregate routes, together with the associated show outputs which can be used to verify and troubleshoot them.

IS-IS Link Bundling

In This Chapter

This section provides information about IS-IS link bundling.

Topics in this section include:

- [Applicability on page 302](#)
- [Overview on page 303](#)
- [Configuration on page 307](#)
- [Conclusion on page 322](#)

Applicability

This example is applicable to all 7750 SR, 7450 ESS and 7950 XRS systems with IOM3-XPs or IMMs using chassis mode D.

The configuration was tested on release 13.0.R3.

Overview

Intermediate System to Intermediate System (IS-IS) Link Bundling allows for the grouping of a number of IS-IS interfaces into a single virtual link, called an IS-IS link group. It is used in conjunction with Equal Cost Multipath (ECMP) to dynamically change the metric of parallel IS-IS links if one or more links fail or suffer some sort of performance degradation.

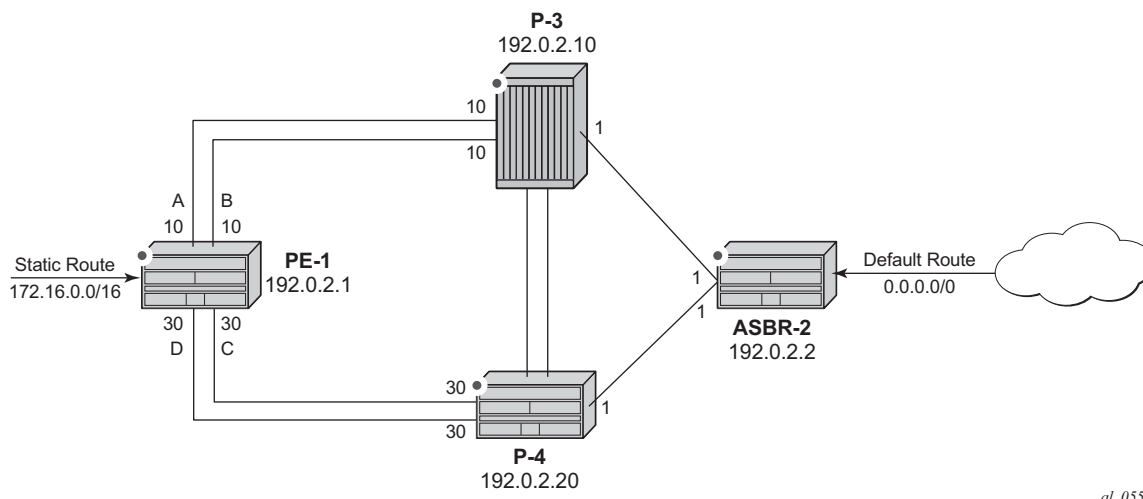


Figure 58: Link Bundle Schematic

Consider the network in [Figure 58](#), where a Provider Edge router PE-1 connects to a core network comprised of two Provider (P) routers and a single Autonomous System Border Router (ASBR). The links between PE-1 and P-3, and PE-1 and P-4 are 10 Gigabit Ethernet links. The links between ASBR-2 and P-3 and P-4 are both 100Gig links. The link metrics are as shown in [Figure 58](#).

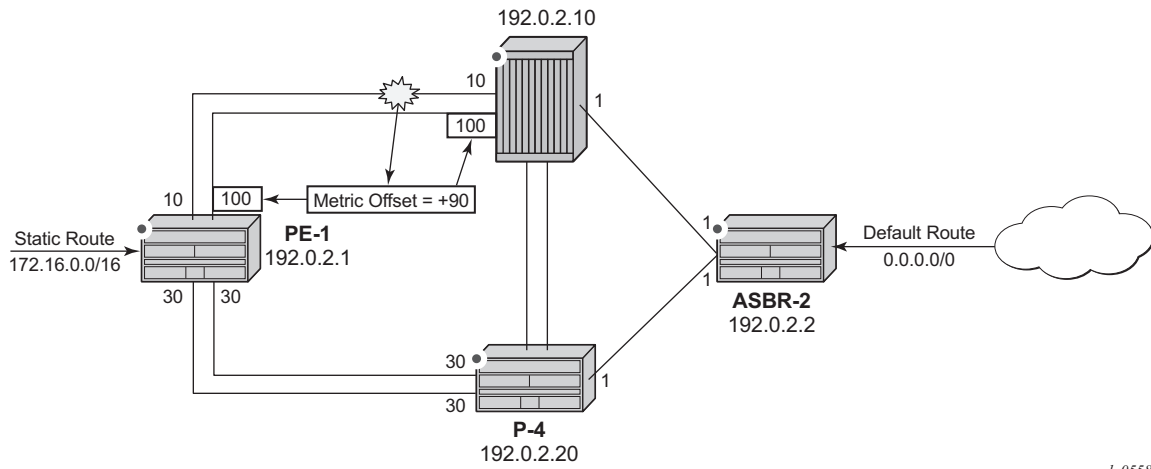
In order to maximize the use of link bandwidth ECMP is enabled on all routers and set to a value of 2 so that IP traffic flowing between PE1 and P-3, and PE-1 and P-4, is load balanced across the two links.

A default route is injected into the ASBR-2 router and re-distributed via a policy statement into IS-IS so that traffic flowing from PE-1 to the ASBR is resolved by this route. Traffic flows between PE-1 and ASBR-2 using the path with the lowest IS-IS metric, via P-3 with a metric of 11. The second path PE-1 to ASBR-2 via P-4 has the same bandwidth but a higher IS-IS metric of 31.

Traffic in the reverse direction flows toward a user subnet described by a static route configured on PE-1 which is redistributed into IS-IS using a policy statement. Once again, the shortest path between ASBR-2 and PE-1 is via P-3, so the bi-directional traffic flow is symmetric.

If one of the links between PE-1 and P-3 fails, traffic still flows via P-3 as the IS-IS metric is unchanged, but this now has less bandwidth than the second path via P-4. It is desirable to make use of the additional bandwidth of the second path, but this requires a change in metric. This can be achieved using IS-IS link bundling.

IS-IS link bundling allows for the creation of a group of IS-IS links, where the failure of a member link allows the metric of the remaining members of the link group to be increased by an offset value.



al_0558

Figure 59: Effect of Single Link Failure on Bundle Group

Using [Figure 59](#) as an example, the links between PE-1 and P-3 are included in a bundle group. To illustrate the change in metrics, a default static route is configured on ASBR-2 and re-distributed into IS-IS, and the path to this route is monitored at PE-1. Similarly, a static route to subnet 172.16.0.0/16 is configured on PE-1 and redistributed into IS-IS and viewed on ASBR-2.

Should one of the links between PE-1 and P-3 fail, the metric of the remaining members can be increased by an offset, for example 90, so that the metric of the remaining link becomes $10 + 90 = 100$. The IS-IS metric between PE-1 and ASBR-2 via P-3 is now 101. Note that the metric offset is applied to each remaining IS-IS interface individually and is advertised within the IS-IS database as the default cost in the TE-IS neighbors Type Length Variable (TLV).

The path between PE-1 and ASBR-2 via P-4 now has the lowest IS-IS metric, and any affected routers within the IS-IS area will try and re-route the traffic based on the new metric.

The fundamentals of this feature are:

- The treatment of all member links in a link group bundle as a single virtual interface.

- The increase in metric by a given offset value of each remaining individual link within the group when a failure of one or more links occurs.
 - The application of the offset occurs when the number of active links drops below a configured threshold.
 - The offset is removed when the number of active links within the link group bundle reaches the configured reversion threshold.
 - A link bundle is required on a router for the thresholds and offsets to apply.

Consider a second and subsequent failure where a link between PE-1 and P-4 also fails, so that there is only one active IS-IS interface between PE-1 and each of its neighboring P routers. This is shown in [Figure 60](#).

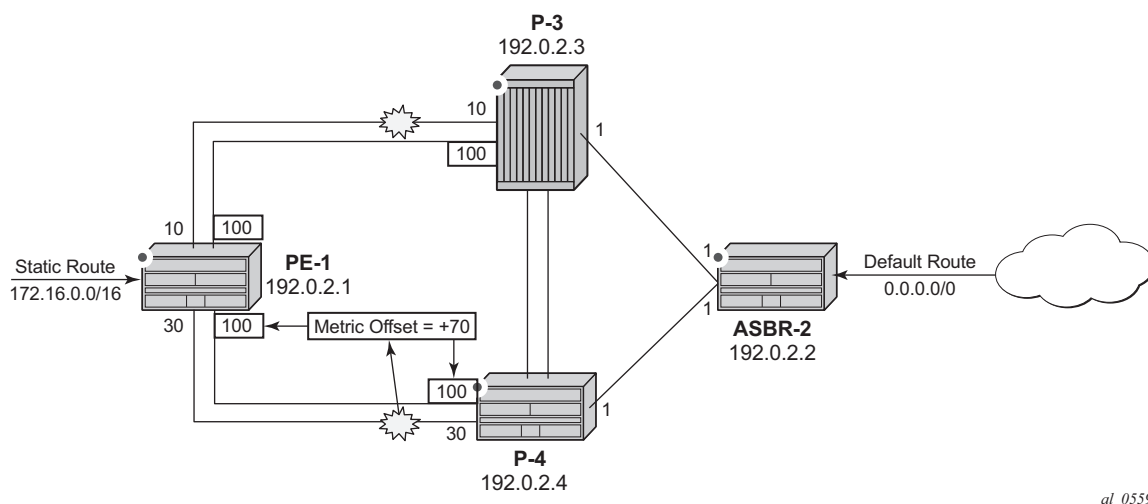


Figure 60: Double Link Failure

In this case, the metric for the remaining link between PE-1 and P-4 can be increased by an offset value of +70 so that the IS-IS metric PE-1 to P-4 becomes 100, the same as that between PE-1 and P-3 when a link has failed.

PE-1 now sees two equal cost paths to the default route – one via P-3 and one via P-4, so there are still two 10Gigabit Ethernet links across which the traffic can be load shared.

This can be summarized using the following table, where ABCD are the 4 links as per [Figure 58](#) and link status is Up (U) or Down (D).

Overview

ABCD Status	A (metric,status)	B (metric,status)	C (metric,status)	D (metric,status)
UUUU	10 Transmit	10 Transmit	30 Idle	30 Idle
UDUU	100 Idle	Down	30 Transmit	30 Transmit
UDUD	100 Transmit	Down	100 Transmit	Down
UUUD	10 Transmit	10 Transmit	100 Idle	Down

Configuration

The test topology is shown in [Figure 61](#).

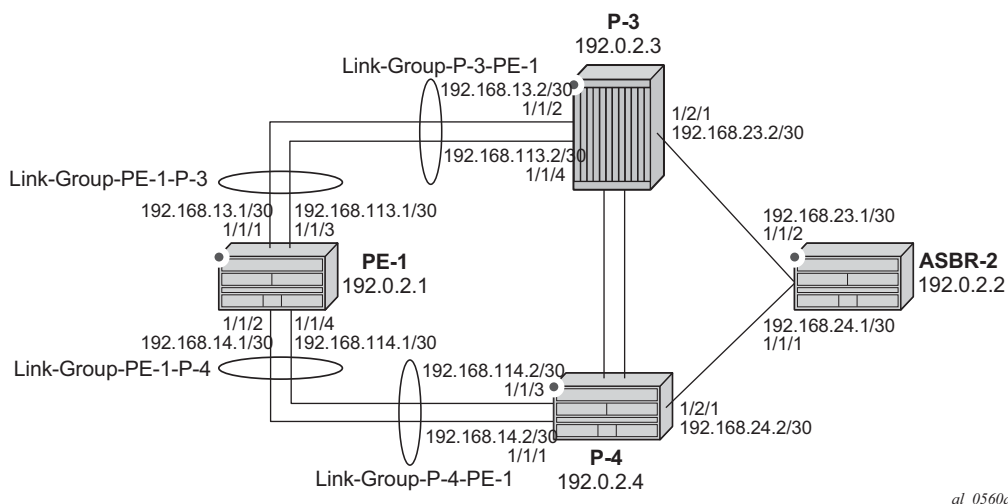


Figure 61: Test Topology

The PE-1 router configuration commands are shown below.

```
*A:PE-1# configure router
  interface "int-PE-1-P-3-1"
    address 192.168.13.1/30
    port 1/1/1
  exit
  interface "int-PE-1-P-3-2"
    address 192.168.113.1/30
    port 1/1/3
  exit
  interface "int-PE-1-P-4-1"
    address 192.168.14.1/30
    port 1/1/2
  exit
  interface "int-PE-1-P-4-2"
    address 192.168.114.1/30
    port 1/1/4
  exit
  interface "system"
    address 192.0.2.1/32
  exit
  ecmp 2
```

The IP router configuration for the remaining routers can be derived from [Figure 61](#).

The IS-IS network is a level 1 network.

Configuration

The IS-IS configuration for PE-1, including the interface metrics is shown below.

```
*A:PE-1# configure router
isis
  level-capability level-1
  area-id 49.0001
  advertise-passive-only
  level 1
    wide-metrics-only
  exit
  interface "system"
    passive
  exit
  interface "int-PE-1-P-3-1"
    interface-type point-to-point
    level 1
      metric 10
    exit
  exit
  interface "int-PE-1-P-3-2"
    interface-type point-to-point
    level 1
      metric 10
    exit
  exit
  interface "int-PE-1-P-4-1"
    interface-type point-to-point
    level 1
      metric 30
    exit
  exit
  interface "int-PE-1-P-4-2"
    interface-type point-to-point
    level 1
      metric 30
    exit
  exit
exit
```

Once again, the IS-IS configuration for the remaining routers can be derived from [Figure 61](#).

The following configuration is for the static route and export policy on ASBR-2. The configuration of the static route on PE-1 is similar.

```
*A:ASBR-2# configure router static-route 0.0.0.0/0 black-hole

*A:ASBR-2# configure router
policy-options
  begin
  policy-statement "STATIC-ISIS"
    entry 10
      from
        protocol static
      exit
      to
        level 1
      exit
      action accept
```



```
metric set igp
exit
exit
exit
commit
exit

*A:ASBR-2# configure router isis export "STATIC-ISIS"
```

Link Group Configuration

PE-1 contains 2 link groups. The first link group contains the IS-IS interfaces toward P-3. The second contains the interfaces toward P-4.

Each link-group is configured using a unique name, which is unique per router, and the IS-IS interface names are configured within the group as group members.

The metric offset value is the amount by which the IS-IS metric of active member links are increased when the number of links drops below a configured threshold.

The IS-IS link group configuration for PE-1 for the interfaces toward P-3 is as follows:

```
*A:PE-1# configure router
isis
  link-group "Link-Group-PE-1-P-3"
  level 1
    ipv4-unicast-metric-offset 90
    member "int-PE-1-P-3-1"
    member "int-PE-1-P-3-2"
    revert-members 2
    oper-members 2
  exit
exit
exit
```

Similarly, the IS-IS link group for PE-1 for the interfaces toward P-4 is:

```
*A:PE-1# configure router
isis
  link-group "Link-Group-PE-1-P-4"
  level 1
    ipv4-unicast-metric-offset 70
    member "int-PE-1-P-4-1"
    member "int-PE-1-P-4-2"
    revert-members 2
    oper-members 2
  exit
exit
exit
```

Within the link-group two thresholds are configured:

- oper-members threshold
- revert-members threshold

If the number of operational links in the link-group drops below the oper-members value, then all interfaces associated with that IS-IS link group have their interface metric increased by the configured offset value. As a result, IS-IS then tries to reroute traffic over lower cost paths.

If the number of operational links in the link-group equals the revert-members threshold value, then all interfaces associated with that IS-IS link group have their interface metric decreased by the configured offset value.

In this configuration, there is a requirement to increase the metric of each interface within a link-group when a single interface fails. This means that the oper-members value is set to 2. In normal working circumstances when both interfaces are active, the metric used is the configured interface metric. This means that the revert-members value must also be set to 2.

Note that it is not possible to set the oper-members threshold to a value higher than that of the revert-members.

For completeness, the IS-IS configuration of each P-router is as follows.

P-3

```
*A:P-3# configure router
  isis
    level-capability level-1
    area-id 49.0001
    advertise-passive-only
    level 1
      wide-metrics-only
    exit
    interface "system"
      passive
    exit
    interface "int-P-3-PE-1-1"
      interface-type point-to-point
      level 1
        metric 10
      exit
    exit
    interface "int-P-3-PE-1-2"
      interface-type point-to-point
      level 1
        metric 10
      exit
    exit
    interface "int-P-3-PE-2"
      interface-type point-to-point
      level 1
        metric 1
      exit
    exit
    link-group "Link-Group-P-3-PE-1"
      level 1
        ipv4-unicast-metric-offset 90
        member "int-P-3-PE-1-1"
        member "int-P-3-PE-1-2"
```

Link Group Configuration

```
        revert-members 2
        oper-members 2
    exit
exit
exit
```

P-4

```
*A:P-4# configure router
isis
    level-capability level-1
    area-id 49.0001
    advertise-passive-only
    level 1
        wide-metrics-only
    exit
    interface "system"
        passive
    exit
    interface "int-P-4-PE-1-1"
        interface-type point-to-point
        level 1
            metric 30
        exit
    exit
    interface "int-P-4-PE-1-2"
        interface-type point-to-point
        level 1
            metric 30
        exit
    exit
    interface "int-P-4-PE-2"
        interface-type point-to-point
        level 1
            metric 1
        exit
    exit
    link-group "Link-Group-P-4-PE-1"
        level 1
            ipv4-unicast-metric-offset 70
            member "int-P-4-PE-1-1"
            member "int-P-4-PE-1-2"
            revert-members 2
            oper-members 2
        exit
    exit
exit
```

An overview of all of the link groups can be shown using the following commands, in this case on node PE-1.

First, the Link-Group Status is shown:

```
*A:PE-1# show router isis link-group-status
=====
Router Base ISIS Instance 0 Link-Group Status
=====
Link-group          Mbrs   Oper   Revert Active Level   State
                   Mbr    Mbr    Mbr    Mbr
-----
Link-Group-PE-1-P-3    2     2     2     2     L1    normal
Link-Group-PE-1-P-4    2     2     2     2     L1    normal
=====
*A:PE-1#
```

Now, the output for the individual link group members is shown:

For "Link-Group-PE-1-P-3" at PE-1:

```
*A:PE-1# show router isis link-group-member-status level 1 "Link-Group-PE-1-P-3"
=====
Router Base ISIS Instance 0 Link-Group Member
=====
Link-group          I/F name          Level   State
-----
Link-Group-PE-1-P-3  int-PE-1-P-3-1    L1      Up
Link-Group-PE-1-P-3  int-PE-1-P-3-2    L1      Up
-----
Legend: BER = bitErrorRate
=====
*A:PE-1#
```

Link Group Configuration

For "Link-Group-PE-1-P-4" at PE-1:

```
*A:PE-1# show router isis link-group-member-status level 1 "Link-Group-PE-1-P-4"
=====
Router Base ISIS Instance 0 Link-Group Member
=====
Link-group          I/F name          Level    State
-----
Link-Group-PE-1-P-4  int-PE-1-P-4-1     L1       Up
Link-Group-PE-1-P-4  int-PE-1-P-4-2     L1       Up
-----
Legend: BER = bitErrorRate
=====
*A:PE-1#
```

For P-3, the following outputs show the link-group and link-group member status.

```
*A:P-3# show router isis link-group-status
=====
Router Base ISIS Instance 0 Link-Group Status
=====
Link-group          Mbrs   Oper   Revert Active Level  State
                   Mbr     Mbr     Mbr     Mbr
-----
Link-Group-P-3-PE-1  2       2       2       2       L1    normal
=====
*A:P-3#

*A:P-3# show router isis link-group-member-status level 1 "Link-Group-P-3-PE-1"
=====
Router Base ISIS Instance 0 Link-Group Member
=====
Link-group          I/F name          Level    State
-----
Link-Group-P-3-PE-1  int-P-3-PE-1-1     L1       Up
Link-Group-P-3-PE-1  int-P-3-PE-1-2     L1       Up
-----
Legend: BER = bitErrorRate
=====
*A:P-3#
```

Routing Table PE-1

In a normal working state, the routing table for PE-1 contains the default route for forwarding traffic toward ASBR-2. As ECMP is set to a value of 2 two entries are available with next-hops pointing toward P-3, as shown below. Note that the metric for each path is 11.

```
*A:PE-1# show router route-table 0.0.0.0/0
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type   Proto   Age           Pref
      Next Hop[Interface Name]                      Metric
-----
0.0.0.0/0                                           Remote  ISIS    00h02m27s    15
      192.168.13.2                                   11
0.0.0.0/0                                           Remote  ISIS    00h02m27s    15
      192.168.113.2                                  11
-----
No. of Routes: 2
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====
*A:PE-1#
```

Failure of link member PE-1 to P-3

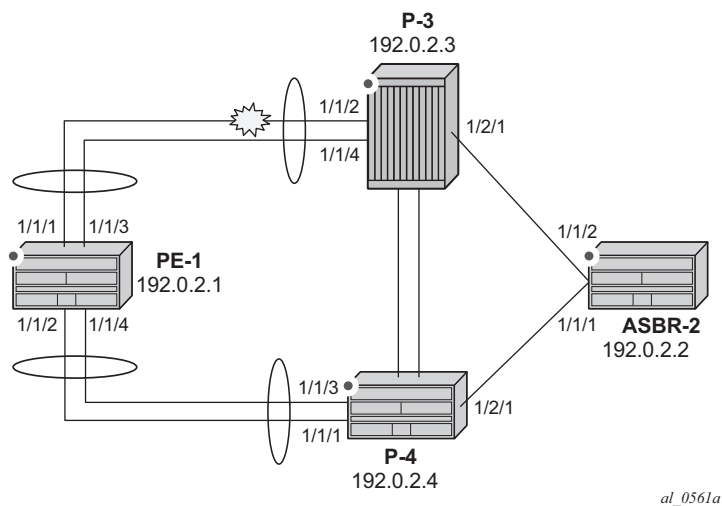


Figure 62: Link Failure

Link Group Configuration

One of the links between PE-1 and P-3 is put into a failed state by shutting down port 1/1/2 on P-3, as per [Figure 62](#).

```
*A:P-3# configure port 1/1/2 shutdown
```

The route-table on PE-1 shows that the metric for the default route prefix, 0.0.0.0/0, has increased from 11 to 31, and the next-hops are now interface addresses on P-4.

```
A:PE-1# show router route-table 0.0.0.0/0
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type   Proto   Age           Pref
  Next Hop[Interface Name]                        Metric
-----
0.0.0.0/0                                           Remote  ISIS    00h01m14s    15
      192.168.14.2                                   31
0.0.0.0/0                                           Remote  ISIS    00h01m14s    15
      192.168.114.2                                   31
-----
No. of Routes: 2
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====
*A:PE-1#
```

The link-group status shows that the number of active members has fallen below the oper-members threshold and as a result, the metric offset has been applied.

```
*A:PE-1# show router isis link-group-status
=====
Router Base ISIS Instance 0 Link-Group Status
=====
Link-group      Mbrs   Oper   Revert Active Level   State
                Mbr    Mbr    Mbr    Mbr
-----
Link-Group-PE-1-P-3  2      2      2      1      L1      Offset-Applied
Link-Group-PE-1-P-4  2      2      2      2      L1      normal
=====
*A:PE-1#
```


Finally, the status of an individual link group member can be shown.

```
*A:PE-1# show router isis link-group-member-status "Link-Group-PE-1-P-3"
=====
Router Base ISIS Instance 0 Link-Group Member
=====
Link-group          I/F name          Level    State
-----
Link-Group-PE-1-P-3  int-PE-1-P-3-1    L1       If-Down
Link-Group-PE-1-P-3  int-PE-1-P-3-2    L1       Up
-----
Legend: BER = bitErrorRate
=====
*A:PE-1#
```

By examining the IS-IS database on PE-1, it can be seen that the link metric (TE-IS neighbor) toward P-3 has a metric of 100, comprised of the original metric of 10 plus the offset of 90.

```
*A:PE-1# show router isis database PE-1 detail
=====
Router Base ISIS Instance 0 Database
=====

Displaying Level 1 database
-----
LSP ID       : PE-1.00-00                      Level      : L1
Sequence     : 0x9                             Checksum   : 0x9312  Lifetime   : 1016
Version      : 1                               Pkt Type   : 18     Pkt Ver    : 1
Attributes: L1                               Max Area   : 3
SysID Len    : 6                               Used Len   : 156   Alloc Len  : 1492

TLVs :
  Area Addresses:
    Area Address : (3) 49.0001
  Supp Protocols:
    Protocols    : IPv4
  IS-Hostname    : PE-1
  Router ID      :
    Router ID    : 192.0.2.1
  I/F Addresses :
    I/F Address  : 192.0.2.1
    I/F Address  : 192.168.13.1
    I/F Address  : 192.168.14.1
    I/F Address  : 192.168.113.1
    I/F Address  : 192.168.114.1
  TE IS Nbrs :
    Nbr         : P-3.00
    Default Metric : 100
    Sub TLV Len  : 12
    IF Addr      : 192.168.113.1
    Nbr IP       : 192.168.113.2
  TE IS Nbrs :
    Nbr         : P-4.00
    Default Metric : 30
    Sub TLV Len  : 12
    IF Addr      : 192.168.14.1
```

Link Group Configuration

```
Nbr IP      : 192.168.14.2
TE IS Nbrs  :
Nbr         : P-4.00
Default Metric : 30
Sub TLV Len  : 12
IF Addr     : 192.168.114.1
Nbr IP      : 192.168.114.2
TE IP Reach :
Default Metric : 0
Control Info:   , preflen 16
Prefix      : 172.16.0.0
Default Metric : 0
Control Info:   , preflen 32
Prefix      : 192.0.2.1
```

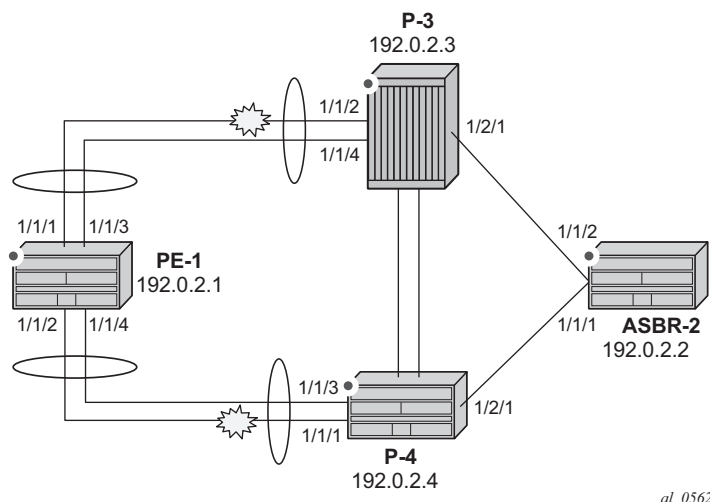
Level (1) LSP Count : 1

Displaying Level 2 database

Level (2) LSP Count : 0

=====

```
*A:PE-1#
```

Failure of link member PE-1 to P-4:**Figure 63: Second Link Failure**

If a link between PE-1 and P-4 now fails, simulated by shutting down port 1/1/1 on P-4, then the metric offset is applied to the link groups on PE-1 and P-4 as the number of active links has dropped below the oper-members threshold for the link groups Link-Group-PE-1-P-4 on PE-1 and Link-Group-P-4-PE-1 on P-4.

```
*A:P-4# configure port 1/1/1 shutdown
```

The routing table for PE-1 now shows that there are still two equal cost paths for the default route prefix advertised by ASBR-2, as shown in the following output:

```
*A:PE-1# show router route-table 0.0.0.0/0
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type  Proto  Age           Pref
Next Hop[Interface Name]                        Metric
-----
0.0.0.0/0                                           Remote  ISIS   00h01m16s    15
192.168.113.2                                       101
0.0.0.0/0                                           Remote  ISIS   00h01m16s    15
192.168.114.2                                       101
-----
No. of Routes: 2
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
```

Link Group Configuration

```
=====
*A:PE-1#
```

Note that the metric for each routing table entry is 101, comprising of a cost of 100 for the PE-1 to P router link, where the link-group offset has been applied, and the cost of 1 for the P router to ASBR-2 router link.

By examining the IS-IS database on the PE-1 router, the updated metric for the link to neighbors P-3 and P-4 can be seen with the offset applied. These are seen in the “TE-IS Nbrs” TLV in the following output.

```
*A:PE-1# show router isis database PE-1 detail
```

```
=====
Router Base ISIS Instance 0 Database
```

```
=====
Displaying Level 1 database
```

```
-----
LSP ID       : PE-1.00-00                               Level       : L1
Sequence     : 0xa                                       Checksum     : 0x5ebc   Lifetime     : 1105
Version      : 1                                         Pkt Type    : 18       Pkt Ver      : 1
Attributes: L1                                         Max Area    : 3
SysID Len    : 6                                         Used Len    : 131     Alloc Len    : 1492
```

```
TLVs :
```

```
Area Addresses:
```

```
Area Address : (3) 49.0001
```

```
Supp Protocols:
```

```
Protocols    : IPv4
```

```
IS-Hostname   : PE-1
```

```
Router ID    :
```

```
Router ID    : 192.0.2.1
```

```
I/F Addresses :
```

```
I/F Address  : 192.0.2.1
```

```
I/F Address  : 192.168.13.1
```

```
I/F Address  : 192.168.14.1
```

```
I/F Address  : 192.168.113.1
```

```
I/F Address  : 192.168.114.1
```

```
TE IS Nbrs   :
```

```
Nbr          : P-3.00
```

```
Default Metric : 100
```

```
Sub TLV Len    : 12
```

```
IF Addr       : 192.168.113.1
```

```
Nbr IP        : 192.168.113.2
```

```
TE IS Nbrs   :
```

```
Nbr          : P-4.00
```

```
Default Metric : 100
```

```
Sub TLV Len    : 12
```

```
IF Addr       : 192.168.114.1
```

```
Nbr IP        : 192.168.114.2
```

```
TE IP Reach   :
```

```
Default Metric : 0
```

```
Control Info:   , prefLen 16
```

```
Prefix        : 172.16.0.0
```

```
Default Metric : 0
```

```
Control Info:   , prefLen 32
```

```
Prefix        : 192.0.2.1
```

```
Level (1) LSP Count : 1
```

Displaying Level 2 database

Level (2) LSP Count : 0

Level (2) LSP Count : 0

*A:PE-1#

Conclusion

IS-IS link bundling allows service providers to configure multiple IS-IS interfaces as a single link group for ECMP purposes and allow link metric increases if an interface within the bundle group fails. This example provides the configuration for IS-IS link bundling, together with the associated commands and outputs which can be used for verifying and troubleshooting.

In This Section

This section provides MPLS configuration information for the following topics:

- [Automatic Bandwidth Adjustment in P2P LSPs on page 325](#)
- [Automatic Creation of RSVP-TE LSPs on page 365](#)
- [BGP Anycast on page 387](#)
- [IGP Shortcuts on page 431](#)
- [Inter-Area TE Point-to-Point LSPs on page 493](#)
- [LDP over RSVP Using OSPF as IGP on page 521](#)
- [MPLS LDP FRR using ISIS as IGP on page 563](#)
- [MPLS Transport Profile on page 589](#)
- [Point-to-Point LSPs on page 617](#)
- [RSVP Signaled Point-to-Multipoint LSPs on page 651](#)
- [Segment Routing with IS-IS Control Plane on page 701](#)
- [Shared Risk Link Groups for RSVP-Based LSP on page 725](#)

Automatic Bandwidth Adjustment in P2P LSPs

In This Chapter

This section provides information about automatic bandwidth adjustment in P2P LSPs.

Topics in this section include:

- [Applicability on page 326](#)
- [Overview on page 327](#)
- [Configuration on page 340](#)
- [Conclusion on page 364](#)

Applicability

This note is applicable to all 7x50 chassis. At a minimum chassis mode C must be used (and for 7450 ESS, at least chassis mode D). The feature was first introduced in 8.0.R4. For this configuration example, it was tested on 13.0.R2.

Overview

Automatic Bandwidth Adjustment refers to the capability of an ingress Label Edge Router (iLER) to dynamically adjust the bandwidth of an RSVP LSP (Resource Reservation Protocol Label Switched Path) tunnel based on active measurement of the traffic rate into the tunnel. The bandwidth assigned to an RSVP LSP tunnel is taken into account by the control plane, to verify that sufficient bandwidth is available for a new LSP or for an increase or decrease in bandwidth for an existing LSP. The actual bandwidth in the data plane is not capped by this setting ¹.

Auto-Bandwidth adjustment uses the existing LSP Egress Statistics feature to track the bandwidth on a specific LSP. When egress statistics are enabled, the Control Processing Module (CPM) collects statistics from all IOMs forwarding traffic belonging to the LSP (whether the traffic is currently leaving the ingress LER via the primary path, a secondary path, or an FRR detour/bypass path). The egress statistics have counts for the number of packets and bytes forwarded per LSP on a per-forwarding class, per priority (in-profile versus out-of-profile) basis.

For the actual bandwidth adjustment, Make-Before-Break (MBB) is used. No traffic interruption is noticed. If an auto-bandwidth attempt fails, there will be 5 retries² and, if they all fail, the bandwidth remains unchanged. The next attempt may occur with the next trigger.

Retries follow the retry-limit (5 in this case), retry-timer (by default 30s), and exponential back-off timer, if enabled in MPLS.

Auto-bandwidth adjustment can be triggered in four different ways:

1. Periodic trigger

The iLER determines at the end of each adjust-interval whether to attempt an auto-bandwidth adjustment.

2. Overflow or underflow trigger

The measured bandwidth of an LSP has increased or decreased significantly since the start of the current adjust-interval. It may be preferable to adjust the bandwidth of the LSP after a number of overflow/underflow samples, rather than wait for the adjust-interval to end (default: 24 h).

3. Manual trigger

An operator launches a tools command to trigger an auto-bandwidth adjustment.

4. Active path change

1. QoS mechanisms can be set up to filter and police the traffic in the data plane, but that is beyond the scope of this example.

2. Numerals will be used in this document, even when less than 10, for consistency, and because they often appear in code examples.

Overview

The LSP has a primary and one or more secondary paths. When there is a change from the primary path to a secondary path without the LSP going down, an auto-bandwidth MBB is triggered. When the primary path becomes active again, another auto-bandwidth MBB is triggered.

Periodic Trigger

Figure 64 shows the different time intervals and bandwidths defined in the auto-bandwidth adjustment implementation. In this example, there will be an auto-bandwidth attempt when the adjust-interval elapses (periodic trigger). If the auto-bandwidth algorithm is met, the current bandwidth is increased. The parameters are explained after the figure.

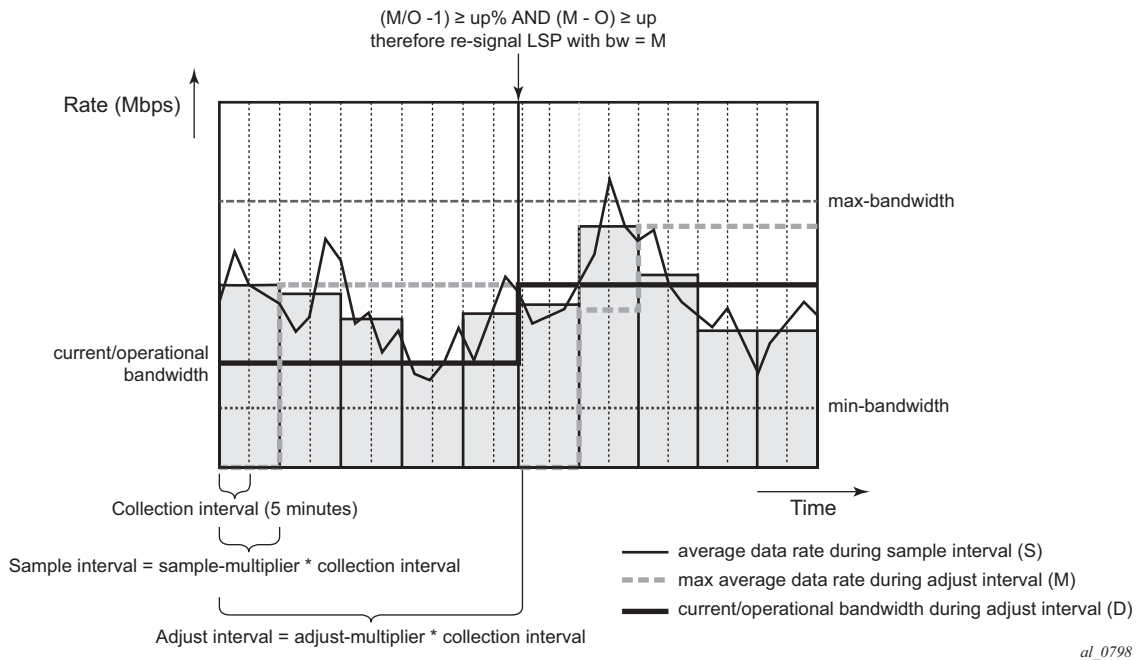


Figure 64: Auto-Bandwidth Adjustment Implementation

The time intervals are:

- Collection interval in minutes. This is a global parameter to be set in an accounting policy. Range: 5 to 120 minutes³. Default: 5 minutes.

```
*A:PE-1# configure log accounting-policy 10 collection-interval
- collection-interval <minutes>
- no collection-interval
```

```
<minutes> : [1..120]
```

```
*A:PE-1# configure log accounting-policy 10 collection-interval 1
MAJOR: LOG #1076 Except for policies using a record type of SAA or PM the minimum interval
is 5 mins
```

3. For this kind of record type, the minimum interval is 5 minutes. For policies using a record type of SAA or PM, the minimum is 1 minute.

- Sample interval: sample-multiplier * collection interval
- Sample-multiplier is configurable globally in the MPLS context or per LSP. Default value: 1. In [Figure 64](#), the sample multiplier equals 2 for a sample interval of 2 * 5 minutes = 10 minutes.
- Adjust-interval: adjust-multiplier * collection interval
 - Alcatel-Lucent recommends that the adjust-multiplier is an integer multiple of the sample-multiplier.
 - Adjust-multiplier is configurable globally in the MPLS context or per LSP. Default value: 288 (288 * 5 minutes = 1440 minutes = 24 h). In [Figure 64](#), the adjust multiplier equals 10 for an adjust-interval of 10 * 5 minutes = 50 minutes.

```
*A:PE-1# configure router mpls auto-bandwidth-multipliers
- auto-bandwidth-multipliers sample-multiplier <number1> adjust-multiplier <number2>
- no auto-bandwidth-multipliers
```

```
<number1>          : [1..511]
<number2>          : [1..16383]
```

```
*A:PE-1# configure router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth multipliers
- multipliers sample-multiplier <num1> adjust-multiplier <num2>
- no multipliers
```

```
<num1>             : [1..511]
<num2>             : [1..16383]
```

The different bandwidths are:

- Minimum bandwidth: configured minimum bandwidth in Mbps that the auto-bandwidth adjustment can signal for the LSP. Granularity: 1 Mbps⁴. Default: 0 Mbps.

```
*A:PE-1# configure router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth min-bandwidth
- min-bandwidth <mbps>
- no min-bandwidth
```

```
<mbps>             : [0..100000]
```

- Maximum bandwidth: configured maximum bandwidth in Mbps that the auto-bandwidth adjustment can signal for the LSP. Granularity: 1 Mbps. Default: 100 Mbps.

```
*A:PE-1# configure router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth max-bandwidth
- max-bandwidth <mbps>
- no max-bandwidth
```

```
<mbps>             : [0..100000]
```

4. For consistency, “Mbps” is used in this section, rather than “Mb/s”.

- Current bandwidth or operational bandwidth (O): currently reserved bandwidth in Mbps for the LSP in the control plane. This is the operational bandwidth that is maintained in the Management Information Base (MIB) and is the bandwidth that will be auto-adjusted. Granularity: 1 Mbps.
- Sampled bandwidth (S): average data rate for the last sample interval.
- Measured bandwidth (M): maximum averaged (per sample interval) data rate in the current adjust-interval. The SR OS keeps track of the maximum average data rate of each LSP since the last reset of the adjust-count.
- Signaled bandwidth: bandwidth in Mbps that is provided to the CSPF algorithm and signaled in the RSVP SENDER_TSPEC and FLOWSPEC objects, when an auto-bandwidth adjustment is attempted. Granularity: 1 Mbps.

The other auto-bandwidth parameters for periodically triggered auto-bandwidth adjustment are:

- Up% (adjust-up in percent): minimum increase in bandwidth from current to measured bandwidth, expressed as a percentage of the current bandwidth. Default: 5%.
- Up (adjust-up bw): minimum increase in bandwidth as absolute bandwidth in Mbps. Up = measuredBW – currentBW. Granularity: 1 Mbps. Default: 0 Mbps.

```
*A:PE-1>config>router>mpls>lsp>auto-bandwidth# adjust-up
- adjust-up <percent> [bw <mbps>]
- no adjust-up
```

```
<percent>          : [0..100]
<mbps>             : [0..100000]
```

- Down% (adjust-down in percent): minimum decrease in bandwidth from current to measured bandwidth, expressed as a percentage of the current bandwidth. Default: 5%.
- Down (adjust-down bw): minimum decrease in bandwidth as absolute bandwidth in Mbps. Down = currentBW – measuredBW. Granularity: 1 Mbps. Default: 0 Mbps.

```
*A:PE-1>config>router>mpls>lsp>auto-bandwidth# adjust-down
- adjust-down <percent> [bw <mbps>]
- no adjust-down
```

```
<percent>          : [0..100]
<mbps>             : [0..100000]
```

In [Figure 64](#), the minimum and maximum bandwidths mark the bandwidth range where auto-bandwidth adjustments are allowed. The sample interval is two collection intervals long (2 * 5 minutes = 10 minutes). The adjust-interval is 10 collection intervals long (10 * 5 minutes = 50 minutes). Initially, the current bandwidth (O) equals the configured bandwidth for the primary path. It is good practice to give that same value to the minimum bandwidth for auto-bandwidth. The system doesn't confirm this and these bandwidths are independent from each other.

In this example, the sampled bandwidth exceeds the current bit rate in most of the sample intervals. The maximum sampled bandwidth in the current adjust-interval corresponds to the

measured bandwidth (M). When auto-bandwidth adjustment is triggered at the end of the adjust-interval, this measured bandwidth will be signaled and, after a successful adjustment, will be the new current bandwidth. After the auto-bandwidth adjustment, a new adjust-interval starts and the measured bandwidth is reset to 0. As long as the first sample interval of the new adjust-interval is not finished, the measured bandwidth equals 0 and auto-adjustment would be impossible even when triggered manually.

The auto-bandwidth attempt follows these rules:

- When $\text{measuredBW} \geq \text{currentBW}$
 - if $\{(\text{measuredBW} / \text{currentBW} - 1) \geq \text{up\%}\} \ \&\& \ \{(\text{measuredBW} - \text{currentBW}) \geq \text{up}\}$
then $\text{signaledBW} = \max\{(\min(\text{measuredBW}, \text{maxBW})), \text{minBW}\}$
- When $\text{measuredBW} \leq \text{currentBW}$
 - if $\{(1 - \text{measuredBW}/\text{currentBW}) \geq \text{down\%}\} \ \&\& \ \{(\text{currentBW} - \text{measuredBW}) \geq \text{down}\}$
then $\text{signaledBW} = \min\{(\max(\text{measuredBW}, \text{minBW})), \text{maxBW}\}$

CLI configured bandwidths have a granularity of 1 Mbps, while the threshold calculations with measured bandwidth are performed at full precision. This means that the signaled bandwidth in the RSVP message is rounded up to the nearest integer multiple of 1 Mbps.

Overflow/Underflow Trigger

Auto-bandwidth adjustment can also be triggered by overflow or underflow. When the bandwidth changes drastically, the bandwidth can be auto-adjusted after a number of consecutive overflow/underflow samples. In this case, there is no need to wait for the adjust-interval to end (by default: 24 h).

The parameters used in case of overflow are:

- Overflow sample: a sample interval counts as an overflow sample if the sampled bandwidth is higher than the current bandwidth by at least the configured overflow thresholds.
- Overflow-limit/overflow-count: an auto-bandwidth adjustment occurs after this number of consecutive overflow samples.
- Threshold%: minimum difference between sampled bandwidth and current bandwidth, expressed as a percentage of the current bandwidth.
- Threshold bw: minimum difference between sampled bandwidth and current bandwidth in Mbps. Default value: 0.

```
A:PE-1# configure router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth overflow-limit
- overflow-limit <number> threshold <percent> [bw <mbps>]
- no overflow-limit

<number>          : [1..10]
<percent>         : [0..100]
<mbps>            : [0..100000]
```

The rules for overflow-triggered auto-bandwidth adjustment are as follows:

- Overflow sample: $\{(sampledBW / currentBW - 1) \geq threshold\% \} \&\& \{(sampledBW - currentBW) \geq thresholdBW\}$
- The signaled bandwidth will be:
 - if $(measuredBW \geq maxBW)$ then $signaledBW = maxBW$
 - if $(measuredBW \leq minBW)$ then $signaledBW = minBW^5$
 - else $signaledBW = measuredBW$

Underflow triggers were introduced in 12.0.R1. The parameters used in case of underflow are:

5. This is impossible in case of overflow. The measured bandwidth will never be lower than the minimum bandwidth then.

Overflow/Underflow Trigger

- Underflow sample: a sample interval counts as an underflow sample if the sampled bandwidth is lower than the current bandwidth by at least the configured underflow thresholds.
- Underflow-limit/underflow-count: an auto-bandwidth adjustment occurs after this number of consecutive underflow samples.
- Threshold%: minimum difference between current bandwidth and sampled bandwidth, expressed as a percentage of the current bandwidth.
- Threshold bw: minimum difference between current bandwidth and sampled bandwidth in Mbps. Default value: 0.
- Maximum underflow bandwidth (MU): maximum sampled bandwidth in the consecutive underflow samples.

```
A:PE-1# configure router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth underflow-limit
- underflow-limit <number> threshold <percent> [bw <mbps>]
- no underflow-limit
```

```
<number>          : [1..10]
<percent>         : [0..100]
<mbps>           : [0..1000000]
```

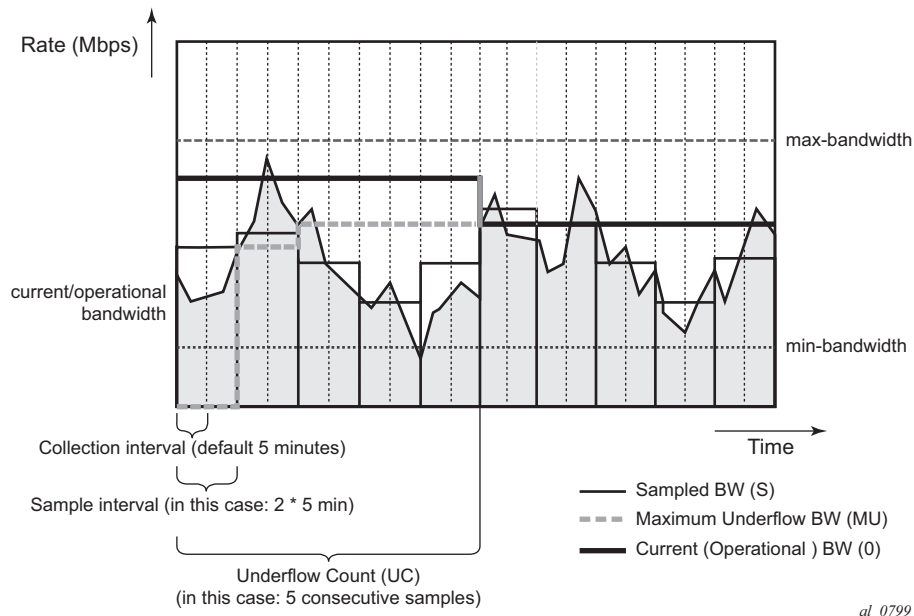


Figure 65: Underflow-Triggered Auto-Bandwidth Implementation

In [Figure 65](#), the adjust-interval is not displayed. It is assumed to be the default of 288 collection intervals (24 h). The figure only shows five consecutive underflow samples. The underflow-limit equals 5. In each of the samples, the sample bandwidth is below the underflow threshold. The maximum sampled bandwidth of these five samples corresponds to the maximum underflow bandwidth. This bandwidth will be signaled when auto-bandwidth adjustment is triggered because the underflow count is reached.

The rules for underflow-triggered auto-bandwidth adjustment are as follows:

- Underflow sample:
 - $\{(1 - \text{sampledBW} / \text{currentBW}) \geq \text{threshold}\} \ \&\& \ \{(\text{currentBW} - \text{sampledBW}) \geq \text{thresholdBW}\}$
- Underflow count/underflow limit: after that many consecutive underflow samples, an auto-bandwidth adjustment is triggered.
- The signaled bandwidth will be:
 - if $(\text{maxUnderflowBW} \geq \text{maxBW})$ then $\text{signaledBW} = \text{maxBW}$ ⁶
 - if $(\text{maxUnderflowBW} \leq \text{minBW})$ then $\text{signaledBW} = \text{minBW}$
 - else $\text{signaledBW} = \text{maxUnderflowBW}$

If the adjustment is successful, the sample counter within the adjust-interval is reset, along with other parameters, such as the maximum underflow bandwidth, the measured bandwidth, and the underflow count. The next adjust-interval will elapse in 24 h.

If the adjustment fails, there will be 5 retries. If they all fail, only the underflow count and the maximum underflow bandwidth are reset. The current adjust-interval continues.

6. This is impossible in case of underflow. The maximum underflow bandwidth can never be equal to or greater than the maximum bandwidth then.

Manual Trigger

Besides the periodic trigger and the overflow/underflow trigger, an operator can launch a tools command to trigger an auto-bandwidth adjustment.

```
A:PE-1# tools perform router mpls adjust-autobandwidth
```

This tools command can be launched with or without explicit LSP name. In the latter case, all active LSPs are attempted for auto-bandwidth.

```
A:PE-1# tools perform router mpls adjust-autobandwidth lsp "LSP-PE-1-PE-2"
- adjust-autobandwidth [lsp <lsp-name> [force [bandwidth <mbps>]]]
```

```
<lsp-name>      : [64 chars max]
<force>         : keyword
<mbps>          : [0..100000]
```

```
A:PE-1# tools perform router mpls adjust-autobandwidth lsp "LSP-PE-1-PE-2"
```

This command (without the keyword force) triggers a new auto-bandwidth calculation according to the rules of periodic triggered type. If the LSP already has the correct reserved bandwidth, the following message is returned.

```
A:PE-1# tools perform router mpls adjust-autobandwidth lsp "LSP-PE-1-PE-2"
MINOR: CLI lsp LSP-PE-1-PE-2 active path is already at the requested value 12 Mbps.
```

If the keyword force is added without a specific value for the bandwidth, there is no threshold checking. The bandwidth can also be adjusted if the difference in bandwidth is below the thresholds. The granularity remains 1 Mbps.

```
A:PE-1# tools perform router mpls adjust-autobandwidth lsp "LSP-PE-1-PE-2" force
```

The rules for the signaled bandwidth are unchanged:

- if (measuredBW \geq maxBW) then signaledBW = maxBW
- if (measuredBW \leq minBW) then signaledBW = minBW
- else signaledBW = measuredBW

If the keyword force with bandwidth (in Mbps) option is given, the signaled bandwidth is set to this configured bandwidth, even if it is a value below the minimum or higher than the maximum bandwidth.

```
A:PE-1# tools perform router mpls adjust-autobandwidth lsp "LSP-PE-1-PE-2" force bandwidth
30
```

After a manually triggered auto-bandwidth MBB, no counters are reset. The ongoing adjust-interval is not aborted.

A clear command resets all counters and timers associated with auto-bandwidth adjustment on a specified LSP.

```
A:PE-1# clear router mpls lsp-autobandwidth "LSP-PE-1-PE-2"
```

This command clears the parameters that are shown in bold in the following example. The parameters will be explained in the configuration section.

```
A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth
=====
MPLS LSP (Auto Bandwidth)
=====
-----
Type : Originating
-----
LSP Name      : LSP-PE-1-PE-2
Auto BW       : Enabled
Auto BW Min   : 2 Mbps
AB Up Thresh  : 10 percent
AB Up BW      : 1 Mbps
AB Curr BW    : 2 Mbps
AB Adj Mul    : 288
AB Adj Time   : 1440 Mins
AB Adj Cnt   : 0
AB Last Adj   : 04/28/2015 08:10:19
ABMaxAvgRt  : 0 Mbps
AB Ovfl Lmt   : 1
ABOvflThres   : 10 percent
AB UndflLmt   : 3
ABUndflThrs   : 10 percent
ABMaxUndflBW  : 0 Mbps
AB Adj Cause  : underflow
Be Weight     : 50 percent
L1 Weight     : 100 percent
Nc Weight     : 100 percent
H1 Weight     : 100 percent
AB OpState    : Up
Auto BW Max   : 20 Mbps
AB Down Thresh : 5 percent
AB Down BW    : 0 Mbps
AB Samp Intv   : 5 Mins
AB Samp Mul    : 1
AB Samp Time   : 5 Mins
AB Samp Cnt  : 0
AB Next Adj    : 1440 Mins
AB Lst AvgRt : 0 Mbps
AB Ovfl Cnt  : 0
AB Ovfl BW     : 2 Mbps
AB Undrfl Cnt : 0
AB Undrfl BW   : 2 Mbps
AB Monitor BW  : False
Af Weight     : 80 percent
L2 Weight     : 100 percent
Ef Weight     : 100 percent
H2 Weight     : 100 percent
=====
A:PE-1#
```

Passive Monitoring

The system offers the option to measure the bandwidth of an LSP without taking any action to adjust the bandwidth reservation.

```
A:PE-1# configure router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth monitor-bandwidth
```

Auto-Bandwidth Based on Forwarding Class

From 11.0.R4 onward, the bandwidth can be calculated as a weighted sum of all the traffic in the eight forwarding classes. By default, all forwarding classes have the same weight: 100%, but that sampling weight is configurable.

```
A:PE-1# configure router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth fc
- fc <fc-name> sampling-weight <sampling-weight>
- no fc <fc-name>

<fc-name>          : be|l2|af|l1|h2|ef|h1|nc
<sampling-weight>  : [0..100]
```

Active Path Change

From 12.0.R4 onward, auto-bandwidth adjustment is also supported on LSPs that have secondary paths. If the secondary path is standby, an auto-bandwidth MBB can be triggered when the active path changes from primary to secondary. The secondary/standby path is only initialized at its configured bandwidth when it is established, and the bandwidth is adjusted only when it becomes active. This happens when the primary path goes down or becomes degraded. When another path becomes active, the bandwidth used to signal the auto-bandwidth MBB will be the operational bandwidth of the previous path.

The definition for current bandwidth is modified for this feature:

- Current bandwidth: last known reserved bandwidth for the LSP. This may be for a different path than the active one.
Auto-bandwidth adjustment will only take place on the active path. When the active path changes, the current bandwidth is updated to the operational bandwidth of the new active path.
- For a secondary path that is signaled as standby, if the active path for an LSP changes without the LSP going down, an auto-bandwidth MBB is triggered on the new active path. The signaled bandwidth is the operational bandwidth of the previous path. The reserved bandwidth of the new active path will be its configured bandwidth until the MBB succeeds.

- For a secondary path where the active path goes down, the LSP will go down temporarily until the secondary path is set up. When the LSP goes down, all statistics and counters are cleared, so the previous path operational bandwidth is lost. There will be no immediate bandwidth adjustment on the secondary path.

The following rules apply to determine the signaled bandwidth of the new active path.

- For a path that is operationally down, signaledBW = configuredBW.
- For the first 5 MBB attempts on the path that just became active, signaledBW = currentBW (operational bandwidth of the previous path).

For the remaining MBB attempts, signaledBW = operationalBW.

- For all MBBs other than auto-bandwidth MBB on the active path, MBB signaledBW = operationalBW.
- For an MBB on the inactive (standby) path, MBB signaledBW = configuredBW.

When the system reverts from a secondary standby path to the primary path, a Delayed Retry MBB is attempted to bring the bandwidth on the standby path back to the configured bandwidth. MBB is attempted once, and if it fails, the standby is torn down. A Delayed Retry MBB has the highest priority among MBBs, so it will take precedence over any other MBB in progress on the standby path, such as configuration change or pre-emption.

Configuration

Figure 66 shows the lab setup. The focus will be on the RSVP LSP from PE-1 to PE-2.

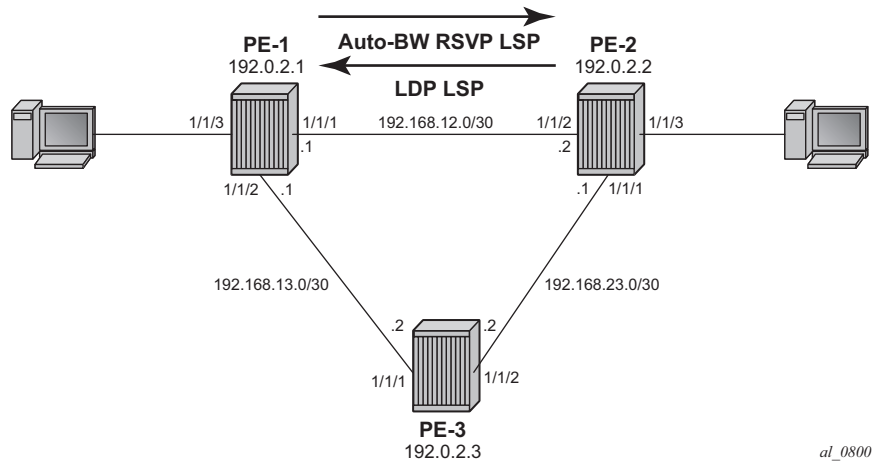


Figure 66: Lab Setup for Auto-Bandwidth Point-to-Point LSPs

Base Configuration

The cards, MDAs and ports need to be configured.

Configure the interfaces on all nodes. For PE-1:

```
configure router
  interface "int-PE-1-PE-2"
    address 192.168.12.1/30
    port 1/1/1
  exit
  interface "int-PE-1-PE-3"
    address 192.168.13.1/30
    port 1/1/2
  exit
  interface "system"
    address 192.0.2.1/32
  exit
```

As an IGP, OSPF or IS-IS can be used. In this example, OSPF is configured. Traffic engineering should be enabled. For PE-1:

```
configure router
```



```

ospf
  traffic-engineering
  area 0.0.0.0
    interface "system"
    exit
    interface "int-PE-1-PE-2"
      interface-type point-to-point
    exit
    interface "int-PE-1-PE-3"
      interface-type point-to-point
    exit
  exit
exit

```

Optionally, enable LDP on all interfaces. Link-layer LDP is not a pre-requisite for using auto-bandwidth RSVP LSPs. In this example, the SDP from PE-2 to PE-1 uses an LDP LSP, but it could have been an RSVP-TE LSP instead. For PE-2:

```

configure router ldp
  interface-parameters
    interface "int-PE-2-PE-1"
    exit
    interface "int-PE-2-PE-3"
    exit
  exit
  targeted-session
exit

```

Enable MPLS and RSVP on all nodes.

```

configure router mpls no shutdown
configure router rsvp no shutdown

```

Add all interfaces to the MPLS context. They will automatically be added to the RSVP context. For PE-1:

```

configure router
  mpls
    interface "int-PE-1-PE-2"
    exit
    interface "int-PE-1-PE-3"
    exit
  exit

```

Configure a path with no explicitly defined hops and LSP LSP-PE-1-PE-2 on PE-1:

```

configure router mpls
  path "loose"
  no shutdown
  exit
  lsp "LSP-PE-1-PE-2"
  to 192.0.2.2

```

Base Configuration

```
        primary "loose"  
        exit  
        no shutdown  
    exit
```

In the example, traffic needs to be injected into the LSP tunnel. For that, a VPLS service is created. For PE-1, an SDP using the RSVP LSP to PE-2 is created.

```
configure service  
    sdp 212 mpls create  
        description "SDP-PE-1-PE-2-overRSVP-TE"  
        far-end 192.0.2.2  
        lsp "LSP-PE-1-PE-2"  
        no shutdown  
    exit
```

On PE-2, an SDP using LDP is created toward PE-1.

```
configure service  
    sdp 121 mpls create  
        description "SDP-PE-2-PE-1-overLDP"  
        far-end 192.0.2.1  
        ldp  
        no shutdown  
    exit
```

On PE-1 and PE-2, a VPLS is created. For PE-1:

```
configure service vpls 100 customer 1 create  
    sap 1/1/3 create  
    exit  
    spoke-sdp 212:100 create  
    exit  
    no shutdown  
    exit
```

The configuration on PE-2 is similar.

Pre-requisites for Auto-Bandwidth LSP Configuration

Enable Constrained Shortest Path First (CSPF) on the LSP by adding the keyword **cspf**.

```
configure router mpls lsp "LSP-PE-1-PE-2" cspf
```

The bandwidth of the LSP will be adjusted in a Make-Before-Break (MBB) manner. Enable MBB on the LSP by adding the keyword **adaptive** to the primary path.

```
configure router mpls lsp "LSP-PE-1-PE-2" primary "loose" adaptive
```

Enter a value for the bandwidth in Mbps for the primary path. It is good practice to configure the same value as for the minimum bandwidth in the auto-bandwidth settings.

```
configure router mpls lsp "LSP-PE-1-PE-2" primary "loose" bandwidth 2
```

Auto-Bandwidth LSP Configuration

MPLS auto-bandwidth adjustment allows the ingress LER to dynamically adjust the bandwidth of an RSVP tunnel based on active measurements of the traffic rate into the tunnel. Therefore, LSP egress statistics need to be enabled on the iLER.

Auto-bandwidth adjustment requires an accounting policy to be defined and operational. The accounting policy specifies the collection interval for LSP statistics collection, which is fundamental to the auto-bandwidth algorithm. The minimum interval for this type of collection is 5 minutes, which is the default value.

```
configure log
    accounting-policy 10
        record combined-mpls-lsp-egress
        to no-file
        no shutdown
    exit
```

An accounting policy of record type **combined-mpls-lsp-egress**⁷ doesn't need a reference to a specific file ID⁸:

```
configure log accounting-policy 10 to no-file
```

-
7. From the moment auto-bandwidth is enabled with an LSP context, the record combined-mpls-lsp-egress inside the accounting policy will also take bandwidth measurements.
 8. When **to no-file** is configured, no LSP statistics are stored anymore. The MPLS auto-bandwidth feature retrieves its LSP stats information directly from the statistics module.

Auto-Bandwidth LSP Configuration

However, the accounting policy can reference a file and, therefore, a CF card. An additional CF card may be required in each node as a storage location. For releases prior to 10.0.R4, this is mandatory.

```
configure log
    file-id 66
        location cf3:
        rollover 15 retention 1
    exit
    accounting-policy 66
        record combined-mpls-lsp-egress
        to file 66
        no shutdown
    exit
```

In the remainder of the example, the accounting policy will reference to no-file.

After the accounting policy has been created, egress statistics can be enabled on the LSP.

```
configure router mpls lsp "LSP-PE-1-PE-2"
    egress-statistics
    no shutdown
    collect-stats
    accounting-policy 10
    exit
```

The system does not verify whether egress statistics have been enabled on the LSP. When a user configures auto-bandwidth adjustment, but without enabling egress statistics, no auto-bandwidth measurements and adjustments are performed. The operational state of auto-bandwidth (AB OpState) is down.

Enable auto-bandwidth with default settings by adding the keyword auto-bandwidth to the LSP.

```
configure router mpls lsp LSP-PE-1-PE-2 auto-bandwidth
```

The actual values are shown in the following output. They are explained after the output.

```
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth
=====
MPLS LSP (Auto Bandwidth)
=====
-----
Type : Originating
-----
LSP Name      : LSP-PE-1-PE-2
Auto BW       : Enabled
Auto BW Min   : 0 Mbps
AB Up Thresh  : 5 percent
AB Up BW      : 0 Mbps
AB Curr BW    : 2 Mbps
AB Adj Mul    : 288+
AB Adj Time   : 1440 Mins
AB OpState    : Up
Auto BW Max   : 100000 Mbps
AB Down Thresh : 5 percent
AB Down BW    : 0 Mbps
AB Samp Intv  : 5 Mins
AB Samp Mul   : 1+
AB Samp Time  : 5 Mins
```

```

AB Adj Cnt : 0
AB Last Adj : n/a
ABMaxAvgRt : 0 Mbps
AB Ovfl Lmt : 0
ABOvflThres : 0 percent
AB UndflLmt : 0
ABUndflThrs : 0 percent
ABMaxUndflBW: 0 Mbps
AB Adj Cause: none
Be Weight : 100 percent
L1 Weight : 100 percent
Nc Weight : 100 percent
H1 Weight : 100 percent

AB Samp Cnt : 0
AB Next Adj : 1440 Mins
AB Lst AvgRt : 0 Mbps
AB Ovfl Cnt : 0
AB Ovfl BW : 0 Mbps
AB Undrfl Cnt : 0
AB Undrfl BW : 0 Mbps
AB Monitor BW : False
Af Weight : 100 percent
L2 Weight : 100 percent
Ef Weight : 100 percent
H2 Weight : 100 percent
=====
*A:PE-1#

```

The plus sign (+) indicates that the value is inherited from the global MPLS settings (AB Adj Mul: 288+ and AB Samp Mul: 1+). The sample-multiplier and the adjust-multiplier can both be configured globally in the MPLS context or overruled by the settings per LSP. In this example, nothing has been configured in the MPLS context or in the LSP. Therefore, the default values as defined in the MPLS context are applicable.

Auto-Bandwidth – Periodic Trigger (Normal)

The default collection interval is 5 minutes. The sample-multiplier is 1, by default. The sample interval equals $1 * 5 \text{ minutes} = 5 \text{ minutes}$. The adjust-multiplier is 288, by default 288. The adjustment interval equals $288 * 5 \text{ minutes} = 1440 \text{ minutes}$ (24 hours).

The auto-bandwidth settings for the LSP are modified as follows:

```

configure router mpls lsp LSP-PE-1-PE-2
    auto-bandwidth
        multipliers sample-multiplier 1 adjust-multiplier 3
        adjust-up 10 bw 1
        adjust-down 5 bw 0 ## default
        max-bandwidth 20
        min-bandwidth 2
    exit

```

In the example, the bandwidth of the LSP can be auto-adjusted every 15 minutes (after 3 intervals of 5 minutes). For a decrease in bandwidth, the default settings apply and no explicit command is required in this example. That means that the current bandwidth will be reduced when the difference in bandwidth is at least 5%. There is no absolute decrease (in Mbps) defined. For an increase in bandwidth, there will only be an adjustment when the increase is at least 10% and at least 1 Mbps. The minimum bandwidth is 2 Mbps. This equals the bandwidth set in the path in the LSP (recommended). The maximum bandwidth equals 20 Mbps. The system will not compare the minimum or maximum bandwidth to the configured bandwidth for the path.

Auto-Bandwidth – Periodic Trigger (Normal)

Display the actual auto-bandwidth data after 5, 10, and 15, minutes.

There are different bandwidths displayed:

- The AB Curr BW is the operational bandwidth during the adjustment interval. It is initially the configured bandwidth of the path in the LSP, but it can be auto-adjusted. This bandwidth is taken into account in the control plane when an LSP is set up or modified in case of MBB. The real data rate in the data plane may exceed this operational bandwidth.
- The ABMaxAvgRt is the measured bandwidth, meaning the maximum averaged bandwidth (calculated every sample interval of 5 minutes) in the adjustment interval of 15 minutes (AB Adj Time: 15 Min).
- The AB Lst AvgRt is the sampled bandwidth, averaged over the latest sample interval of 5 minutes (AB Samp Intv: 5 Mins).

After 5 minutes, one collection interval has elapsed within the adjust-interval (AB Adj Cnt = 1) and the next adjustment time is in 10 minutes (AB Next Adj = 10 Min). The current bandwidth equals 2 Mbps, while the measured and the sampled bandwidths are much higher: 12 Mbps.

```
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth
=====
MPLS LSP (Auto Bandwidth)
=====
Type : Originating
-----
LSP Name      : LSP-PE-1-PE-2
Auto BW       : Enabled
Auto BW Min   : 2 Mbps
AB Up Thresh  : 10 percent
AB Up BW      : 1 Mbps
AB Curr BW    : 2 Mbps
AB Adj Mul    : 3
AB Adj Time   : 15 Mins
AB Adj Cnt    : 1
AB Last Adj   : n/a
ABMaxAvgRt    : 12 Mbps
AB Ovfl Lmt   : 0
ABOvflThres   : 0 percent
AB UndflLmt   : 0
ABUndflThrs   : 0 percent
ABMaxUndflBW  : 0 Mbps
AB Adj Cause  : none
Be Weight     : 100 percent
L1 Weight     : 100 percent
Nc Weight     : 100 percent
H1 Weight     : 100 percent
AB OpState    : Up
Auto BW Max   : 20 Mbps
AB Down Thresh : 5 percent
AB Down BW    : 0 Mbps
AB Samp Intv  : 5 Mins
AB Samp Mul   : 1
AB Samp Time  : 5 Mins
AB Samp Cnt   : 0
AB Next Adj   : 10 Mins
AB Lst AvgRt  : 12 Mbps
AB Ovfl Cnt   : 0
AB Ovfl BW    : 0 Mbps
AB Undrfl Cnt : 0
AB Undrfl BW  : 0 Mbps
AB Monitor BW : False
Af Weight     : 100 percent
L2 Weight     : 100 percent
Ef Weight     : 100 percent
H2 Weight     : 100 percent
=====
*A:PE-1#
```

After 10 minutes, another collection interval has elapsed in the adjust-interval (AB Adj Cnt = 2) and the next adjustment time is in 5 minutes (AB Next Adj = 5 Min).

Automatic Bandwidth Adjustment in P2P LSPs

```
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth
=====
MPLS LSP (Auto Bandwidth)
=====
-----
Type : Originating
-----
LSP Name      : LSP-PE-1-PE-2
Auto BW       : Enabled
Auto BW Min   : 2 Mbps
AB Up Thresh  : 10 percent
AB Up BW      : 1 Mbps
AB Curr BW    : 2 Mbps
AB Adj Mul    : 3
AB Adj Time   : 15 Mins
AB Adj Cnt    : 2
AB Last Adj   : n/a
ABMaxAvgRt    : 12 Mbps
AB Ovfl Lmt   : 0
ABOvflThres   : 0 percent
AB UndflLmt   : 0
ABUndflThrs   : 0 percent
ABMaxUndflBW  : 0 Mbps
AB Adj Cause  : none
Be Weight     : 100 percent
L1 Weight     : 100 percent
Nc Weight     : 100 percent
H1 Weight     : 100 percent
AB OpState    : Up
Auto BW Max   : 20 Mbps
AB Down Thresh : 5 percent
AB Down BW    : 0 Mbps
AB Samp Intv  : 5 Mins
AB Samp Mul   : 1
AB Samp Time  : 5 Mins
AB Samp Cnt   : 0
AB Next Adj   : 5 Mins
AB Lst AvgRt  : 12 Mbps
AB Ovfl Cnt   : 0
AB Ovfl BW    : 0 Mbps
AB Undrfl Cnt : 0
AB Undrfl BW  : 0 Mbps
AB Monitor BW : False
Af Weight     : 100 percent
L2 Weight     : 100 percent
Ef Weight     : 100 percent
H2 Weight     : 100 percent
=====
*A:PE-1#
```

After 15 minutes, auto-bandwidth adjustment occurs. **AB Adj Cause** is normal for periodically triggered adjustments. The next adjustment interval will elapse in 15 minutes. The measured bandwidth **ABMaxAvgRt** is reset to 0 after a successful adjustment.

```
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth
=====
MPLS LSP (Auto Bandwidth)
=====
-----
Type : Originating
-----
LSP Name      : LSP-PE-1-PE-2
Auto BW       : Enabled
Auto BW Min   : 2 Mbps
AB Up Thresh  : 10 percent
AB Up BW      : 1 Mbps
AB Curr BW    : 13 Mbps
AB Adj Mul    : 3
AB Adj Time   : 15 Mins
AB Adj Cnt    : 0
AB Last Adj   : 04/24/2015 12:57:25
ABMaxAvgRt    : 0 Mbps
AB Ovfl Lmt   : 0
ABOvflThres   : 0 percent
AB UndflLmt   : 0
ABUndflThrs   : 0 percent
AB OpState    : Up
Auto BW Max   : 20 Mbps
AB Down Thresh : 5 percent
AB Down BW    : 0 Mbps
AB Samp Intv  : 5 Mins
AB Samp Mul   : 1
AB Samp Time  : 5 Mins
AB Samp Cnt   : 0
AB Next Adj   : 15 Mins
AB Lst AvgRt  : 13 Mbps
AB Ovfl Cnt   : 0
AB Ovfl BW    : 0 Mbps
AB Undrfl Cnt : 0
AB Undrfl BW  : 0 Mbps
```

Auto-Bandwidth – Periodic Trigger (Normal)

```
ABMaxUndflBW: 0 Mbps
AB Adj Cause: normal
Be Weight : 100 percent
L1 Weight : 100 percent
Nc Weight : 100 percent
H1 Weight : 100 percent
AB Monitor BW : False
Af Weight : 100 percent
L2 Weight : 100 percent
Ef Weight : 100 percent
H2 Weight : 100 percent
=====
*A:PE-1#
```

The periodic trigger type rules for auto-bandwidth are:

- When $\text{measuredBW} \geq \text{currentBW}$
if $\{(\text{measuredBW} / \text{currentBW} - 1) \geq \text{up\%}\} \&\& \{(\text{measuredBW} - \text{currentBW}) \geq \text{up}\}$
then $\text{signaledBW} = \max\{(\min(\text{measuredBW}, \text{maxBW})), \text{minBW}\}$

In this case, the measuredBW (13 Mbps) is greater than the currentBW (2 Mbps). The increase is at least 10% (up%) and at least 1 Mbps (up). The bandwidth will be adjusted. The new bandwidth that will be signaled is calculated as follows:

```
signaledBW = max{(min(measuredBW, maxBW)), minBW}
signaledBW = max {(min (13 Mbps, 20 Mbps)), 2 Mbps}
            = max {13 Mbps, 2 Mbps}
            = 13 Mbps
```

Whenever an auto-bandwidth adjustment is performed, a message is stored in log 99.

```
*A:PE-1# show log log-id 99 application "mpls"
=====
Event Log 99
=====
Description : Default System Log
Memory Log contents [size=500 next event=88 (not wrapped)]

87 2015/04/24 12:57:25.84 UTC WARNING: MPLS #2014 Base VR 1:
"LSP path LSP-PE-1-PE-2::loose resigaled as result of autoBandwidth MBB"
```

When the maximum bandwidth is modified to a value that is lower than the current bandwidth, an adjustment occurs at the end of the adjustment interval.

```
configure router mpls lsp LSP-PE-1-PE-2
    auto-bandwidth
    max-bandwidth 10
exit
```

The current bandwidth will be reduced to 10 Mbps (for the same measured bandwidth).

```
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth
=====
MPLS LSP (Auto Bandwidth)
=====
-----
```



```

Type : Originating
-----
LSP Name      : LSP-PE-1-PE-2
Auto BW       : Enabled
Auto BW Min   : 2 Mbps
AB Up Thresh  : 10 percent
AB Up BW      : 1 Mbps
AB Curr BW   : 10 Mbps
AB Adj Mul    : 3
AB Adj Time   : 15 Mins
AB Adj Cnt    : 2
AB Last Adj   : 04/24/2015 14:27:25
ABMaxAvgRt    : 13 Mbps
AB Ovfl Lmt   : 0
ABOvflThres   : 0 percent
AB UndflLmt   : 0
ABUndflThrs   : 0 percent
ABMaxUndflBW  : 0 Mbps
AB Adj Cause  : normal
Be Weight     : 100 percent
L1 Weight     : 100 percent
Nc Weight     : 100 percent
H1 Weight     : 100 percent
AB OpState    : Up
Auto BW Max  : 10 Mbps
AB Down Thresh : 5 percent
AB Down BW     : 0 Mbps
AB Samp Intv   : 5 Mins
AB Samp Mul    : 1
AB Samp Time   : 5 Mins
AB Samp Cnt    : 0
AB Next Adj    : 5 Mins
AB Lst AvgRt   : 13 Mbps
AB Ovfl Cnt    : 0
AB Ovfl BW     : 0 Mbps
AB Undrfl Cnt  : 0
AB Undrfl BW   : 0 Mbps
AB Monitor BW  : False
Af Weight     : 100 percent
L2 Weight     : 100 percent
Ef Weight     : 100 percent
H2 Weight     : 100 percent
=====
*A:PE-1#

```

Auto-Bandwidth - Passive Monitoring

When passive monitoring is enabled, no automatic bandwidth adjustments occurs. When the maximum bandwidth is again raised to 20 Mbps, the bandwidth will not be auto-adjusted even if the measured bandwidth is high enough.

```

configure router mpls lsp LSP-PE-1-PE-2 auto-bandwidth max-bandwidth 20
configure router mpls lsp LSP-PE-1-PE-2 auto-bandwidth monitor-bandwidth

```

The system monitors the bandwidth, but without taking action at the end of the adjust-interval.

```

*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth
=====
MPLS LSP (Auto Bandwidth)
=====
Type : Originating
-----
LSP Name      : LSP-PE-1-PE-2
Auto BW       : Enabled
Auto BW Min   : 2 Mbps
AB Up Thresh  : 10 percent
AB Up BW      : 1 Mbps
AB Curr BW   : 10 Mbps
AB Adj Mul    : 3
AB Adj Time   : 15 Mins
AB Adj Cnt    : 0
AB OpState    : Up
Auto BW Max  : 20 Mbps
AB Down Thresh : 5 percent
AB Down BW     : 0 Mbps
AB Samp Intv   : 5 Mins
AB Samp Mul    : 1
AB Samp Time   : 5 Mins
AB Samp Cnt    : 0

```

Auto-Bandwidth – Overflow and Underflow Trigger Type

```
AB Last Adj : 04/24/2015 14:27:25      AB Next Adj      : 15 Mins
ABMaxAvgRt  : 0 Mbps                  AB Lst AvgRt     : 13 Mbps
AB Ovfl Lmt : 0                      AB Ovfl Cnt      : 0
ABOvflThres : 0 percent              AB Ovfl BW       : 0 Mbps
AB UndflLmt : 0                      AB Undrfl Cnt    : 0
ABUndflThrs : 0 percent              AB Undrfl BW     : 0 Mbps
ABMaxUndflBW: 0 Mbps
AB Adj Cause: normal                  AB Monitor BW   : True
Be Weight   : 100 percent             Af Weight       : 100 percent
L1 Weight   : 100 percent             L2 Weight       : 100 percent
Nc Weight   : 100 percent             Ef Weight       : 100 percent
H1 Weight   : 100 percent             H2 Weight       : 100 percent
=====
*A:PE-1#
```

Note the value for the parameter **AB Monitor BW: True**

For the remainder of the example, there is no passive monitoring. The settings are restored to normal:

```
configure router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth no monitor-bandwidth
```

Auto-Bandwidth – Overflow and Underflow Trigger Type

With default settings, the adjustment interval is 24 hours. If the bandwidth changes significantly since the start of the current adjust-interval, overflow and underflow triggers can be used. This will speed up the auto-bandwidth adjustment. Overflow triggers are supported from 8.0.R4 onward and underflow triggers from 12.0.R1 onward.

Stop auto-bandwidth in order to force a MBB attempt toward the configured primary path bandwidth (2Mbps in this example).

```
configure router mpls lsp "LSP-PE-1-PE-2" no auto-bandwidth
```

Check the operational bandwidth of the LSP.

```
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" detail
=====
MPLS LSPs (Originating) (Detail)
=====
-----
Type : Originating
-----
LSP Name      : LSP-PE-1-PE-2
LSP Type      : RegularLsp
From          : 192.0.2.1
Adm State     : Up
LSP Up Time   : 2d 15:28:36
Transitions   : 3
LSP Tunnel ID : 1
To            : 192.0.2.2
Oper State    : Up
LSP Down Time : 0d 00:00:00
Path Changes  : 3
```

Automatic Bandwidth Adjustment in P2P LSPs

```

Retry Limit : 0
Signaling : RSVP
Hop Limit : 255
Adaptive : Enabled
FastReroute : Disabled
Egress Stats: Enabled
CSPF : Enabled
Metric : N/A
Load Balanc*: N/A
Include Grps:
None
Least Fill : Disabled

Retry Timer : 30 sec
Resv. Style : SE
Negotiated MTU : 1564
ClassType : 0
Oper FR : Disabled
Egress Oper St*: N/A
ADSPEC : Disabled
Use TE metric : Disabled

Exclude Grps :
None

Revert Timer: Disabled
Auto BW : Disabled
LdpOverRsvp : Enabled
IGP Shortcut: Enabled
IGP LFA : Disabled
BGPTransTun : Enabled
Oper Metric : 10
Prop Adm Grp: Disabled

Next Revert In : N/A

VprnAutoBind : Enabled
BGP Shortcut : Enabled
IGP Rel Metric : Disabled

Primary(a) : loose
Bandwidth : 2 Mbps
Up Time : 2d 15:28:36

```

```

=====
* indicates that the corresponding row element may have been truncated.
*A:PE-1#

```

Enable auto-bandwidth with similar settings as before and add overflow and underflow triggers. The multipliers are default. Therefore, a periodically triggered auto-adjustment will only take place once every 24 hours.

```

configure router mpls lsp LSP-PE-1-PE-2 auto-bandwidth
    multipliers sample-multiplier 1 adjust-multiplier 288
    adjust-up 10 bw 1
    max-bandwidth 20
    min-bandwidth 2
    overflow-limit 1 threshold 10 bw 2
    underflow-limit 3 threshold 10 bw 2
exit

```

The overflow count indicates the number of consecutive times that the overflow condition is detected at the end of a sample interval. Auto-bandwidth adjustment occurs after that number of overflow samples is reached, in this case, after the first overflow sample (overflow-limit = 1). The conditions for an overflow sample are:

```

{(sampledBW / currentBW - 1) ≥ threshold%} && {(sampledBW - currentBW) ≥ thresholdBW}
{(13 Mbps/2Mbps - 1) ≥ 0,1} && {(13Mbps - 2Mbps) ≥ 2Mbps}

```

The signaled bandwidth will be:

- if (measuredBW \geq maxBW) then signaledBW = maxBW
- if (measuredBW \leq minBW) then signaledBW = minBW⁹
else signaledBW = measuredBW
- if (13 Mbps \geq 20 Mbps) then signaledBW = 20 Mbps;
- if (13 Mbps \leq 2 Mbps) then signaledBW = 2 Mbps;
else signaledBW = 13 Mbps

Display the auto-bandwidth data. The **AB Adj Cause** is now **overflow**. The overflow limit is the configured value of 1 (**AB Ovfl Lmt**). The overflow count has been reset after the auto-bandwidth was adjusted (**AB Ovfl Cnt = 0**), along with the **ABMaxAvgRt** and the **AB Adj Cnt**. This is the start of a new adjust-interval of 24 hours.

```
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth
=====
MPLS LSP (Auto Bandwidth)
=====
Type : Originating
-----
LSP Name       : LSP-PE-1-PE-2
Auto BW        : Enabled
Auto BW Min    : 2 Mbps
AB Up Thresh   : 10 percent
AB Up BW       : 1 Mbps
AB Curr BW     : 13 Mbps
AB Adj Mul     : 288
AB Adj Time    : 1440 Mins
AB Adj Cnt     : 0
AB Last Adj    : 04/27/2015 07:42:25
ABMaxAvgRt     : 0 Mbps
AB Ovfl Lmt    : 1
ABOvflThres : 10 percent
AB UndflLmt    : 3
ABUndflThrs    : 10 percent
ABMaxUndflBW   : 0 Mbps
AB Adj Cause: overflow
Be Weight      : 100 percent
L1 Weight      : 100 percent
Nc Weight      : 100 percent
H1 Weight      : 100 percent
AB OpState     : Up
Auto BW Max    : 20 Mbps
AB Down Thresh : 5 percent
AB Down BW     : 0 Mbps
AB Samp Intv   : 5 Mins
AB Samp Mul    : 1
AB Samp Time   : 5 Mins
AB Samp Cnt    : 0
AB Next Adj    : 1440 Mins
AB Lst AvgRt   : 13 Mbps
AB Ovfl Cnt : 0
AB Ovfl BW  : 2 Mbps
AB Undrfl Cnt  : 0
AB Undrfl BW   : 2 Mbps
AB Monitor BW  : False
Af Weight      : 100 percent
L2 Weight      : 100 percent
Ef Weight      : 100 percent
H2 Weight      : 100 percent
=====
*A:PE-1#
```

At the end of a sample interval, the sampled bandwidth is reduced by at least 10% and at least 2 Mbps, and this becomes an underflow sample. The conditions for an underflow sample are:

-
9. This is impossible in case of overflow. The measured bandwidth will never be lower than the minimum bandwidth.

```
{(1 - sampledBW / currentBW) ≥ threshold%} && {(currentBW - sampledBW) ≥ thresholdBW}
{(1 - 10 Mbps / 13 Mbps) ≥ 0,1} && {(13Mbps - 10Mbps) ≥ 2Mbps}
```

In this case, the bandwidth dropped from 13 Mbps to 10 Mbps and the conditions for underflow are met. Since the underflow limit equals 3, an auto-bandwidth adjustment can only take place after the third consecutive underflow sample. The new bandwidth will equal the **maximum sampled underflow bandwidth (ABMaxUndflBW)**. This is the maximum sampled bandwidth in the three consecutive underflow samples.

The signaled bandwidth will be:

```
→ if (maxUnderflowBW ≥ maxBW) then signaledBW = maxBW10
→ if (maxUnderflowBW ≤ minBW) then signaledBW = minBW
   else signaledBW = maxUnderflowBW
→ if (10 Mbps ≥ 20 Mbps) then signaledBW = 20 Mbps;
→ if (10 Mbps ≤ 2 Mbps) then signaledBW = 2 Mbps;
   else signaledBW = 10 Mbps
```

The following output shows the auto-bandwidth data after two consecutive underflow samples (**AB Underfl Cnt: 2**). The maximum sampled underflow bandwidth equals 10 Mbps. No bandwidth adaptation can take place until there are three consecutive underflow samples (**AB UndflLmt: 3**).

```
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth
=====
MPLS LSP (Auto Bandwidth)
=====
-----
Type : Originating
-----
LSP Name      : LSP-PE-1-PE-2
Auto BW       : Enabled
Auto BW Min   : 2 Mbps
AB Up Thresh  : 10 percent
AB Up BW      : 1 Mbps
AB Curr BW    : 13 Mbps
AB Adj Mul    : 288
AB Adj Time   : 1440 Mins
AB Adj Cnt    : 3
AB Last Adj   : 04/27/2015 07:42:25
ABMaxAvgRt    : 12 Mbps
AB Ovfl Lmt   : 1
ABOvflThres   : 10 percent
AB UndflLmt : 3
ABUndflThrs : 10 percent
ABMaxUndflBW: 10 Mbps

AB OpState    : Up
Auto BW Max   : 20 Mbps
AB Down Thresh : 5 percent
AB Down BW    : 0 Mbps
AB Samp Intv   : 5 Mins
AB Samp Mul    : 1
AB Samp Time   : 5 Mins
AB Samp Cnt    : 0
AB Next Adj    : 1425 Mins
AB Lst AvgRt   : 10 Mbps
AB Ovfl Cnt    : 0
AB Ovfl BW     : 2 Mbps
AB Undrfl Cnt : 2
AB Undrfl BW  : 2 Mbps
```

10. This is impossible in case of underflow. The maximum underflow bandwidth can never be equal to or greater than the maximum bandwidth in this case.

Auto-Bandwidth – Overflow and Underflow Trigger Type

AB Adj Cause: overflow	AB Monitor BW : False
Be Weight : 100 percent	Af Weight : 100 percent
L1 Weight : 100 percent	L2 Weight : 100 percent
Nc Weight : 100 percent	Ef Weight : 100 percent
H1 Weight : 100 percent	H2 Weight : 100 percent

=====

*A:PE-1#

After a successful auto-bandwidth adjustment, the **ABMaxUndflBW** is reset, along with the **AB Adj Cnt**, **AB Underfl Cnt** and **ABMaxAvgRt**.

```
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth
=====
MPLS LSP (Auto Bandwidth)
=====
-----
Type : Originating
-----
LSP Name      : LSP-PE-1-PE-2
Auto BW       : Enabled
Auto BW Min   : 2 Mbps
AB Up Thresh  : 10 percent
AB Up BW      : 1 Mbps
AB Curr BW    : 10 Mbps
AB Adj Mul    : 288
AB Adj Time   : 1440 Mins
AB Adj Cnt    : 0
AB Last Adj   : 04/27/2015 08:02:25
ABMaxAvgRt   : 0 Mbps
AB Ovfl Lmt   : 1
ABOvflThres   : 10 percent
AB UndflLmt  : 3
ABUndflThrs  : 10 percent
ABMaxUndflBW: 0 Mbps
AB Adj Cause: underflow
Be Weight     : 100 percent
L1 Weight     : 100 percent
Nc Weight     : 100 percent
H1 Weight     : 100 percent
AB OpState    : Up
Auto BW Max   : 20 Mbps
AB Down Thresh : 5 percent
AB Down BW    : 0 Mbps
AB Samp Intv   : 5 Mins
AB Samp Mul    : 1
AB Samp Time   : 5 Mins
AB Samp Cnt    : 0
AB Next Adj    : 1440 Mins
AB Lst AvgRt   : 7 Mbps
AB Ovfl Cnt    : 0
AB Ovfl BW     : 2 Mbps
AB Undrfl Cnt : 0
AB Undrfl BW  : 2 Mbps
AB Monitor BW  : False
Af Weight      : 100 percent
L2 Weight      : 100 percent
Ef Weight      : 100 percent
H2 Weight      : 100 percent
=====
*A:PE-1#
```

If the overload or underload trigger condition is met at the end of an adjust-interval, the auto-bandwidth adjustment is normal, based on the periodic trigger. Overflow and underflow auto-bandwidth adjustments only take place when the adjust-interval is not yet completed.

Auto-Bandwidth – Manual Trigger Type

The auto-bandwidth adjustment can be triggered manually at all times by the following command (with or without the keyword **force**).

```
tools perform router mpls adjust-autobandwidth
tools perform router mpls adjust-autobandwidth lsp "LSP-PE-1-PE-2"
tools perform router mpls adjust-autobandwidth lsp "LSP-PE-1-PE-2" force
```

When no specific LSP is referred to, auto-bandwidth will be attempted on all LSPs. If the LSP already has the requested bandwidth, the following output is returned.

```
*A:PE-1# tools perform router mpls adjust-autobandwidth lsp "LSP-PE-1-PE-2"
MINOR: CLI No Thresholds crossed for lsp LSP-PE-1-PE-2.
```

By adding the keyword **force**, there is no check whether the thresholds are crossed. However, the granularity is 1 Mbps. In this case, it is not possible to signal a bandwidth that is at least 1 Mbps different, so the following error message is returned.

```
*A:PE-1# tools perform router mpls adjust-autobandwidth lsp "LSP-PE-1-PE-2" force
MINOR: CLI lsp LSP-PE-1-PE-2 active path is already at the requested value 13 Mbps.
```

If the first sample interval has not yet expired, the following error message is returned.

```
*A:PE-1# tools perform router mpls adjust-autobandwidth lsp "LSP-PE-1-PE-2"
MINOR: CLI No Autobandwidth Averages computed for lsp LSP-PE-1-PE-2.
```

If the tools command is launched after the first sample interval has expired (**ABMaxAvgRt** is filled in), the overflow trigger rules can be applied. In this example, the traffic is reduced and the thresholds are crossed.

The **AB Adj Cause** is manual.

```
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth
=====
MPLS LSP (Auto Bandwidth)
=====
Type : Originating
-----
LSP Name      : LSP-PE-1-PE-2
Auto BW       : Enabled
Auto BW Min   : 2 Mbps
AB Up Thresh  : 10 percent
AB Up BW      : 1 Mbps
AB Curr BW   : 4 Mbps
AB Adj Mul    : 288
AB Adj Time   : 1440 Mins
AB Adj Cnt    : 1
AB Last Adj   : 04/27/2015 09:02:36
ABMaxAvgRt    : 4 Mbps
AB OpState    : Up
Auto BW Max   : 20 Mbps
AB Down Thresh : 5 percent
AB Down BW    : 0 Mbps
AB Samp Intv  : 5 Mins
AB Samp Mul   : 1
AB Samp Time  : 5 Mins
AB Samp Cnt   : 0
AB Next Adj   : 1435 Mins
AB Lst AvgRt  : 4 Mbps
```


Automatic Bandwidth Adjustment in P2P LSPs

```

AB Ovfl Lmt : 1
ABOvflThres : 10 percent
AB UndflLmt : 3
ABUndflThrs : 10 percent
ABMaxUndflBW: 0 Mbps
AB Adj Cause: manual
Be Weight : 100 percent
L1 Weight : 100 percent
Nc Weight : 100 percent
H1 Weight : 100 percent

AB Ovfl Cnt : 0
AB Ovfl BW : 2 Mbps
AB Undrfl Cnt : 0
AB Undrfl BW : 2 Mbps
AB Monitor BW : False
Af Weight : 100 percent
L2 Weight : 100 percent
Ef Weight : 100 percent
H2 Weight : 100 percent
=====
*A:PE-1#

```

The counters are not reset after a manually triggered auto-bandwidth adjustment. The adjust-interval is not interrupted, the measured bandwidth and the maximum underflow bandwidth are not reset, and the overflow and underflow count are not reset.

Launch the tools command with the keyword **force** and a bandwidth value. This will set the current bandwidth to this value, even if the value is not within the allowed range between the minimum and maximum bandwidth.

```

tools perform router mpls adjust-autobandwidth lsp "LSP-PE-1-PE-2" force bandwidth 30
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth
=====
MPLS LSP (Auto Bandwidth)
=====
-----
Type : Originating
-----
LSP Name      : LSP-PE-1-PE-2
Auto BW       : Enabled
Auto BW Min   : 2 Mbps
AB Up Thresh  : 10 percent
AB Up BW      : 1 Mbps
AB Curr BW    : 30 Mbps
AB Adj Mul    : 288
AB Adj Time   : 1440 Mins
AB Adj Cnt    : 1
AB Last Adj   : 04/27/2015 09:05:42
ABMaxAvgRt    : 4 Mbps
AB Ovfl Lmt   : 1
ABOvflThres   : 10 percent
AB UndflLmt   : 3
ABUndflThrs   : 10 percent
ABMaxUndflBW  : 0 Mbps
AB Adj Cause  : manual
Be Weight     : 100 percent
L1 Weight     : 100 percent
Nc Weight     : 100 percent
H1 Weight     : 100 percent

AB OpState    : Up
Auto BW Max   : 20 Mbps
AB Down Thresh : 5 percent
AB Down BW    : 0 Mbps
AB Samp Intv   : 5 Mins
AB Samp Mul    : 1
AB Samp Time   : 5 Mins
AB Samp Cnt    : 0
AB Next Adj    : 1435 Mins
AB Lst AvgRt   : 4 Mbps
AB Ovfl Cnt    : 0
AB Ovfl BW     : 2 Mbps
AB Undrfl Cnt  : 0
AB Undrfl BW   : 2 Mbps
AB Monitor BW  : False
Af Weight     : 100 percent
L2 Weight     : 100 percent
Ef Weight     : 100 percent
H2 Weight     : 100 percent
=====
*A:PE-1#

```

Manually triggered auto-bandwidth adjustments are also performed using MBB procedures.

Auto-Bandwidth Adjustment Based on Forwarding Class Subset

With the configuration applied so far, there is no distinction between traffic from different Forwarding Classes (FCs). The average data rate is the sum of the traffic from all eight FCs. From 11.0.R4 onward, it is possible to provide a sampling weight for each Forwarding Class (FC) for each auto-bandwidth LSP. The average data rate is now the weighted sum of the traffic from all FCs.

```
*A:PE-1# configure router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth fc
- fc <fc-name> sampling-weight <sampling-weight>
- no fc <fc-name>

<fc-name>          : be|l2|af|l1|h2|ef|h1|nc
<sampling-weight>  : [0..100]

configure router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth fc be sampling-weight 50
configure router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth fc af sampling-weight 80

*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth
=====
MPLS LSP (Auto Bandwidth)
=====
-----
Type : Originating
-----
LSP Name      : LSP-PE-1-PE-2
Auto BW       : Enabled
Auto BW Min   : 2 Mbps
AB Up Thresh  : 10 percent
AB Up BW      : 1 Mbps
AB Curr BW    : 2 Mbps
AB Adj Mul    : 288
AB Adj Time   : 1440 Mins
AB Adj Cnt    : 0
AB Last Adj   : 04/27/2015 09:52:25
ABMaxAvgRt    : 0 Mbps
AB Ovfl Lmt   : 1
ABOvflThres   : 10 percent
AB UndflLmt   : 3
ABUndflThrs   : 10 percent
ABMaxUndflBW  : 0 Mbps
AB Adj Cause  : underflow
Be Weight     : 50 percent
L1 Weight     : 100 percent
Nc Weight     : 100 percent
H1 Weight     : 100 percent
AB OpState    : Up
Auto BW Max   : 20 Mbps
AB Down Thresh : 5 percent
AB Down BW    : 0 Mbps
AB Samp Intv   : 5 Mins
AB Samp Mul    : 1
AB Samp Time   : 5 Mins
AB Samp Cnt    : 0
AB Next Adj    : 1440 Mins
AB Lst AvgRt   : 0 Mbps
AB Ovfl Cnt    : 0
AB Ovfl BW     : 2 Mbps
AB Undrfl Cnt  : 0
AB Undrfl BW   : 2 Mbps
AB Monitor BW  : False
Af Weight     : 80 percent
L2 Weight     : 100 percent
Ef Weight     : 100 percent
H2 Weight     : 100 percent
=====
*A:PE-1#
```

The sampling-weight values can be changed while auto-bandwidth is enabled. The auto-bandwidth algorithm will be reset on the LSP at the end of the current collection interval. At that time, the current bandwidth will not be adjusted and the following counters will be reset to 0: sample count, adjust count, overflow count, underflow count, max average data rate, and max average underflow data rate.

Auto-Bandwidth on LSPs with Secondary Paths

This feature is supported from 12.0.R4 onward.

When the active path goes down or becomes degraded, the bandwidth used to signal the auto-bandwidth MBB will be the operational bandwidth of the previous active path. The parameter `current-bandwidth` requires a modified definition:

Current-bandwidth — The last known reserved bandwidth for the LSP (this may be for a different path than the active one).

When the active path changes, the current bandwidth is updated to the operational bandwidth of the new active path. While the auto-bandwidth MBB on the active path is in progress, a statistics sample might be triggered because the intervals aren't reset when the active path changes. It is possible that an auto-adjustment is needed. The in-progress auto-bandwidth MBB will be restarted with retry attempts to 0 and signaled bandwidth equal to the new measured bandwidth. If after five attempts, auto-bandwidth MBB fails, the current bandwidth and secondary **oper-bw** remain unchanged.

For a secondary/standby path, if the active path changes without the LSP going down, an auto-bandwidth MBB is triggered for the new active path. The bandwidth used to signal the MBB is the operational bandwidth of the previous active path (current bandwidth).

If the primary path is not currently active, but it has not gone down, then any MBB should use the configured bandwidth for the primary path.

Create two new strict paths and assign them to the LSP. The primary path is the direct strict path from PE-1 to PE-2. There are two secondary paths: **path-PE-1-PE-3-PE-2** and **loose**. The first one is standby, the latter is not.

```
configure router mpls
  path path-PE-1-PE-2
    hop 10 192.0.2.2 strict
    no shutdown
  exit
  path path-PE-1-PE-3-PE-2
    hop 10 192.0.2.3 strict
    hop 20 192.0.2.2 strict
    no shutdown
  exit
  lsp "LSP-PE-1-PE-2"
    to 192.0.2.2
    cspf
    fast-reroute facility
    no node-protect
  exit
  primary loose shutdown
  no primary loose
  primary path-PE-1-PE-2
    adaptive
    bandwidth 2
```

Auto-Bandwidth on LSPs with Secondary Paths

```
exit
secondary path-PE-1-PE-3-PE-2
    adaptive
    bandwidth 2
    standby
exit
secondary loose
    adaptive
    bandwidth 2
exit
no shutdown
exit
```

Stop the auto-bandwidth MBB to have the current bandwidth equal to the bandwidth configured for the primary path (2 Mbps).

```
configure router mpls lsp "LSP-PE-1-PE-2" no auto-bandwidth
```

Configure auto-bandwidth with the following settings:

```
configure router mpls lsp "LSP-PE-1-PE-2"
    auto-bandwidth
        multipliers sample-multiplier 1 adjust-multiplier 288
        adjust-up 10 bw 1
        max-bandwidth 20
        min-bandwidth 2
        overflow-limit 2 threshold 10
        underflow-limit 3 threshold 10
        fc be sampling-weight 50
        fc af sampling-weight 80
    exit
```

Initially, the current bandwidth is the configured bandwidth of the primary path: 2 Mbps, but in case of overflow, it will be increased after two overflow samples (10 minutes).

```
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth
=====
MPLS LSP (Auto Bandwidth)
=====
-----
Type : Originating
-----
LSP Name      : LSP-PE-1-PE-2
Auto BW       : Enabled
Auto BW Min   : 2 Mbps
AB Up Thresh  : 10 percent
AB Up BW      : 1 Mbps
AB Curr BW   : 6 Mbps
AB Adj Mul    : 288
AB Adj Time   : 1440 Mins
AB Adj Cnt    : 0
AB Last Adj   : 04/27/2015 12:27:25
ABMaxAvgRt    : 0 Mbps
AB Ovfl Lmt   : 2
AB OpState    : Up
Auto BW Max   : 20 Mbps
AB Down Thresh : 5 percent
AB Down BW    : 0 Mbps
AB Samp Intv  : 5 Mins
AB Samp Mul   : 1
AB Samp Time  : 5 Mins
AB Samp Cnt   : 0
AB Next Adj   : 1440 Mins
AB Lst AvgRt  : 6 Mbps
AB Ovfl Cnt   : 0
```

Automatic Bandwidth Adjustment in P2P LSPs

```

ABOvflThres : 10 percent
AB UndflLmt : 3
ABUndflThrs : 10 percent
ABMaxUndflBW: 0 Mbps
AB Adj Cause: overflow
Be Weight   : 50 percent
L1 Weight   : 100 percent
Nc Weight   : 100 percent
H1 Weight   : 100 percent
AB Ovfl BW   : 0 Mbps
AB Undrfl Cnt : 0
AB Undrfl BW : 0 Mbps
AB Monitor BW : False
Af Weight    : 80 percent
L2 Weight    : 100 percent
Ef Weight    : 100 percent
H2 Weight    : 100 percent
=====
*A:PE-1#

```

Shutdown port 1/1/1 on PE-1 to initiate a failure on the primary path.

```
*A:PE-1# configure port 1/1/1 shutdown
```

Verify that the secondary/standby path is now active.

```

*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" activepath
=====
MPLS LSP: LSP-PE-1-PE-2 (active paths)
=====
Legend :
# - Manually switched path
#F - Manually forced switched path
=====
LSP Name       : LSP-PE-1-PE-2
Path Name      : path-PE-1-PE-3-PE-2
To             : 192.0.2.2
LSP Id         : 5748
Active Path    : Standby
=====
*A:PE-1#

```

Check the auto-bandwidth data on the LSP. The current bandwidth for the LSP is the same as it used to be for the primary path. **AB Adj Cause** = activePathChange.

```

*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth
=====
MPLS LSP (Auto Bandwidth)
=====
Type : Originating
-----
LSP Name       : LSP-PE-1-PE-2
Auto BW        : Enabled
Auto BW Min    : 2 Mbps
AB Up Thresh   : 10 percent
AB Up BW       : 1 Mbps
AB Curr BW     : 6 Mbps
AB Adj Mul     : 288
AB Adj Time    : 1440 Mins
AB Adj Cnt     : 0
AB Last Adj    : 04/27/2015 13:17:25
ABMaxAvgRt     : 0 Mbps
AB Ovfl Lmt    : 2
AB OpState     : Up
Auto BW Max    : 20 Mbps
AB Down Thresh : 5 percent
AB Down BW     : 0 Mbps
AB Samp Intv   : 5 Mins
AB Samp Mul    : 1
AB Samp Time   : 5 Mins
AB Samp Cnt    : 0
AB Next Adj    : 1440 Mins
AB Lst AvgRt   : 6 Mbps
AB Ovfl Cnt    : 0

```

Auto-Bandwidth on LSPs with Secondary Paths

```
ABOvflThres : 10 percent          AB Ovfl BW      : 0 Mbps
AB UndflLmt : 3                   AB Undrfl Cnt   : 0
ABUndflThrs : 10 percent          AB Undrfl BW    : 0 Mbps
ABMaxUndflBW: 0 Mbps
AB Adj Cause: activePathChange    AB Monitor BW   : False
Be Weight    : 50 percent          Af Weight       : 80 percent
L1 Weight    : 100 percent         L2 Weight       : 100 percent
Nc Weight    : 100 percent         Ef Weight       : 100 percent
H1 Weight    : 100 percent         H2 Weight       : 100 percent
=====
*A:PE-1#
```

The original situation is restored.

```
configure port 1/1/1 no shutdown
```

The primary path comes up again.

```
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" activepath
=====
MPLS LSP: LSP-PE-1-PE-2 (active paths)
=====
Legend :
  # - Manually switched path
  #F - Manually forced switched path
=====
LSP Name      : LSP-PE-1-PE-2          LSP Id       : 5748
Path Name     : path-PE-1-PE-2        Active Path   : Primary
To            : 192.0.2.2
=====
*A:PE-1#
```

The auto-bandwidth data again shows **AB Adj Cause: activePathChange**, but with a different timestamp.

```
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-2" auto-bandwidth
=====
MPLS LSP (Auto Bandwidth)
=====
-----
Type : Originating
-----
LSP Name      : LSP-PE-1-PE-2
Auto BW       : Enabled
Auto BW Min   : 2 Mbps
AB Up Thresh  : 10 percent
AB Up BW      : 1 Mbps
AB Curr BW    : 6 Mbps
AB Adj Mul    : 288
AB Adj Time   : 1440 Mins
AB Adj Cnt    : 0
AB Last Adj   : 04/27/2015 13:27:14
ABMaxAvgRt    : 6 Mbps
AB Ovfl Lmt   : 2
AB OpState    : Up
Auto BW Max   : 20 Mbps
AB Down Thresh : 5 percent
AB Down BW    : 0 Mbps
AB Samp Intv  : 5 Mins
AB Samp Mul   : 1
AB Samp Time  : 5 Mins
AB Samp Cnt   : 0
AB Next Adj   : 1430 Mins
AB Lst AvgRt  : 6 Mbps
AB Ovfl Cnt   : 0
```

Automatic Bandwidth Adjustment in P2P LSPs

```
ABOvflThres : 10 percent          AB Ovfl BW      : 0 Mbps
AB UndflLmt : 3                   AB Undrfl Cnt   : 0
ABUndflThrs : 10 percent          AB Undrfl BW    : 0 Mbps
ABMaxUndflBW: 0 Mbps
AB Adj Cause: activePathChange   AB Monitor BW   : False
Be Weight    : 50 percent          Af Weight       : 80 percent
L1 Weight    : 100 percent         L2 Weight       : 100 percent
Nc Weight    : 100 percent         Ef Weight       : 100 percent
H1 Weight    : 100 percent         H2 Weight       : 100 percent
=====
*A:PE-1#
```

Conclusion

Auto-bandwidth adjustment can be enabled on point-to-point LSPs in order to make a realistic bandwidth reservation, based on active iLER traffic monitoring. A user has control over how the bytes count for the different FCs by providing a sampling-weight factor. This can influence the average data rate over the sample interval.

The bandwidth is taken into account in the control plane when LSPs are established or when they change their bandwidth using MBB. The bandwidth in the data plane is not restricted by this setting.

Auto-bandwidth adjustment can be triggered in different ways: periodically, in case of overflow/underflow, manually, and in case of an active path change. It is also possible to have passive monitoring where no adjustment is done.

Automatic Creation of RSVP-TE LSPs

In This Chapter

This section provides information about Automatic Creation of RSVP-TE LSPs.

Topics in this section include:

- [Applicability on page 366](#)
- [Overview on page 367](#)
- [Configuration on page 368](#)
- [Conclusion on page 385](#)

Applicability

This feature is applicable to 7750 SR-7/12, 7750 SR-12e, 7450 ESS-7/12, XRS-20/16c, and 7750-c4/12 with no hardware constraints as this is a control-plane feature only.

The configuration was tested on release 13.0.R2.

Overview

Automatic creation of RSVP-TE LSPs enables the automated creation of point-to-point RSVP-TE LSPs within a single IGP IS-IS level/OSPF area that can subsequently be used by services and/or IGP shortcuts. The feature is divided into two components; creation of an RSVP-TE LSP mesh, and creation of single-hop RSVP-TE LSPs which can be used together though in general it is likely that one or the other is used.

When creating an RSVP-TE LSP mesh, the mesh can be full or partial, the extent of which is governed by a prefix-list containing the system addresses of all nodes that should form part of the mesh. When using single-hop RSVP-TE LSPs, point-to-point LSPs are established to all directly connected neighbors. The purpose of these single-hop LSPs is to allow for ECMP load-balancing of traffic using LDP-over-RSVP, which is not possible using native RSVP LSPs.

The use of automatically created RSVP-TE LSPs avoids manual configuration of RSVP-TE LSP meshes. Even when provisioning tools (such as 5620 SAM) are used to automatically provision these LSPs, auto-mesh still provides a benefit by avoiding increased configuration file sizes.

The use of automatically created targeted LDP sessions is also described when using the automatically created RSVP LSPs for Layer 2 services.

Configuration

Test Topology

The test topology is shown in [Figure 67](#). All routers participate in a single IS-IS Level-2 area that has traffic-engineering enabled. MPLS and RSVP are enabled on every interface, but no LSPs are initially provisioned. All routers are BGP speakers and form part of Autonomous system 64496. PE-5 is a Route-Reflector and the remaining routers are IBGP clients for the VPN IPv4 and L2-VPN Address Families. The objective of this example and test topology is to demonstrate how to automatically create transport LSPs using RSVP or LDP-over-RSVP, and then create services that utilize those LSPs. The exchange of BGP routes is needed for those services.

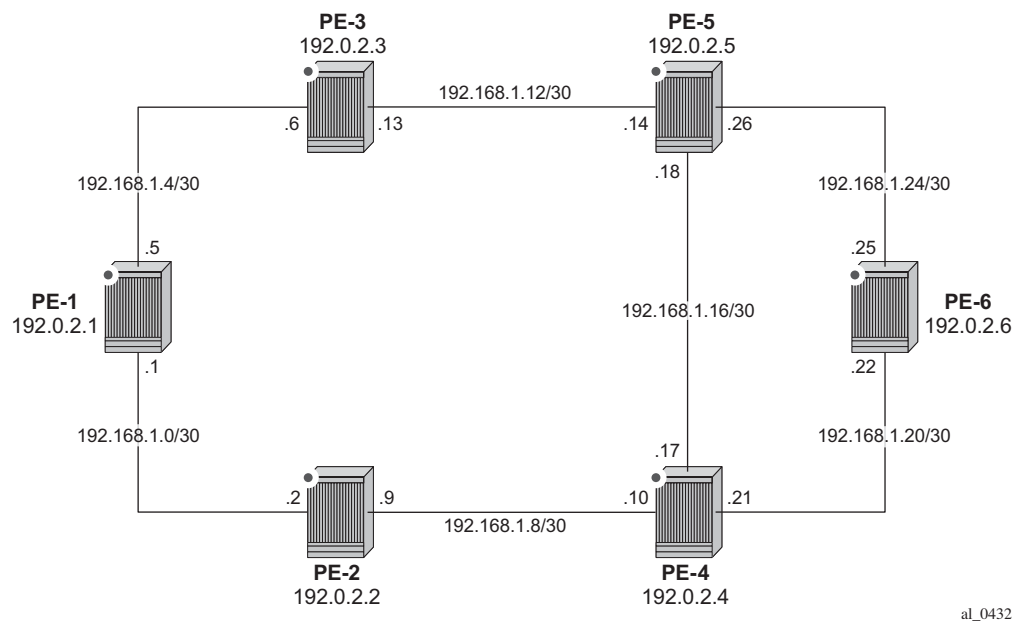


Figure 67: Test Topology

Automatic Creation of an RSVP-TE LSP Mesh

To start the process of automatically creating an RSVP-TE LSP mesh, the user must create a route policy referencing a prefix-list. This prefix-list contains the system addresses of all nodes that are required to be in the mesh, and can be entered as a series of /32 addresses, or simply as a range as shown configured below. This range encompasses all of the system addresses of the nodes in the test topology as the requirement is to make a full-mesh.

```
configure
router
  policy-options
    begin
    prefix-list "System-Addresses"
      prefix 192.0.2.0/24 prefix-length-range 32-32
    exit
    policy-statement "Remote-PEs"
      entry 10
        from
          prefix-list "System-Addresses"
        exit
        action accept
        exit
      exit
    exit
  exit
  commit
exit
```

After the route policy is created the user must create an **lsp-template** containing the common parameters which are used to establish all point-to-point LSPs within the mesh. For an RSVP-TE LSP mesh the **lsp-template** must be suffixed with the creation-time attribute **mesh-p2p**. Upon creation of the template, CSPF is automatically enabled (and cannot be disabled), and the template must reference a **default-path** before it can be placed in a **no shutdown** state. In the example contained in the following output, the template refers to a path named “loose” that has no strict or loose hops defined, meaning the system will dynamically calculate the path whilst considering other specified constraints. The **lsp-template** in this output also stipulates **fast-reroute facility** bypass protection. The default behavior is no node-protect, hence this configuration requests link protection only. Note that one-to-one protection is not supported for automatically created RSVP-TE LSPs; hence facility bypass is the only form of protection supported. Finally the template is placed in a no shutdown state.

Next, the user must associate the lsp-template with the previously defined route-policy, and this is accomplished using the **auto-lsp lsp-template** command. In this example, the lsp-template “Full-Mesh” is associated with the policy-statement “Remote-PEs” that in turn references a prefix-list containing all system addresses in the test topology. (Up to five policies can be associated with a given lsp-template at any one time.) If a policy associated with an lsp-template is modified to add or remove prefixes, the system immediately re-evaluates the policy/prefix-list to determine if one or more LSPs need to be established, or one or more LSPs need to be torn down.

Test Topology

```
configure
router
mpls
  path "loose"
  no shutdown
exit
lsp-template "Full-Mesh" mesh-p2p
  default-path "loose"
  cspf
  fast-reroute facility
exit
no shutdown
exit
auto-lsp lsp-template "Full-Mesh" policy "Remote-PEs"
no shutdown
exit
```

Once the **auto-lsp lsp-template** command is entered, the system commences the process of establishing the point-to-point LSPs. The prefixes defined in the prefix-list are checked, and if a prefix corresponds to a router-id that is present in the Traffic-Engineering database, the system instantiates a CSPF computed primary path to that prefix using the parameters specified in the lsp-template. With the previously defined configuration applied at PE-6, the existence of point-to-point RSVP LSPs to every node in the test topology can be verified as shown in the following output. The LSP name is automatically constructed as TemplateName-DestIPv4Address-TunnelId. The LSP name signaled in the Session Attribute object concatenates the LSP name with the path name (for example Full-Mesh-192.0.2.1-61455::loose).

```
*A:PE-6# show router mpls lsp
=====
MPLS LSPs (Originating)
=====
```

LSP Name	To	Tun Id	Fastfail Config	Adm	Opr
Full-Mesh-192.0.2.1-61455	192.0.2.1	61455	Yes	Up	Up
Full-Mesh-192.0.2.2-61456	192.0.2.2	61456	Yes	Up	Up
Full-Mesh-192.0.2.3-61457	192.0.2.3	61457	Yes	Up	Up
Full-Mesh-192.0.2.4-61458	192.0.2.4	61458	Yes	Up	Up
Full-Mesh-192.0.2.5-61459	192.0.2.5	61459	Yes	Up	Up

```
-----
LSPs : 5
=====
```

Recall that the lsp-template requested fast-reroute link protection. At PE-6 this protection can be verified by querying each primary LSP. In the following output, the primary LSP to PE-1 (Full-Mesh-192.0.2.1-61455) is signaled through PE-5 (192.0.2.5) and PE-3 (192.0.2.3), and the presence of the @ indicator after each hop denotes that link protection is available to the primary path.

```
*A:PE-6# show router mpls lsp path "Full-Mesh-192.0.2.1-61455" detail | match expression
"LSP Name|Actual Hops" post-lines 4
LSP Name      : Full-Mesh-192.0.2.1-61455          Path LSP ID : 44042
From          : 192.0.2.6                          To          : 192.0.2.1
Adm State     : Up                                Oper State  : Up
Path Name     : loose                             Path Type   : Primary
Path Admin    : Up                                Path Oper   : Up
Actual Hops   :
    192.168.1.25 (192.0.2.6) @                    Record Label : N/A
-> 192.168.1.26 (192.0.2.5) @                    Record Label : 262143
-> 192.168.1.13 (192.0.2.3) @                    Record Label : 262143
-> 192.168.1.5  (192.0.2.1)                        Record Label : 262143
```

Finally, it can be verified that the signaled LSPs are placed in the tunnel table and made available to the tunnel table manager so they can be used by applications and services.

```
*A:PE-6# show router tunnel-table
=====
Tunnel Table (Router: Base)
=====
Destination      Owner  Encap  TunnelId  Pref  Nexthop      Metric
-----
192.0.2.1/32     rsvp  MPLS   61455     7     192.168.1.26  300
192.0.2.2/32     rsvp  MPLS   61456     7     192.168.1.21  200
192.0.2.3/32     rsvp  MPLS   61457     7     192.168.1.26  200
192.0.2.4/32     rsvp  MPLS   61458     7     192.168.1.21  100
192.0.2.5/32     rsvp  MPLS   61459     7     192.168.1.26  100
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
```

Once the lsp-template is in use and LSPs are instantiated, it is necessary to place the template into a shutdown state to change any parameters that cannot be handled as a *Make-Before-Break* (MBB). This essentially includes all LSP parameters with the exception of bandwidth and fast-reroute without node-protection. Modification of any other parameters requires a shutdown of the lsp-template and a re-signal of the LSP once the lsp-template is placed in the no shutdown state again. It should be noted however that MBB is supported for timer-based and manual re-signaling of the automatically created LSPs.

Service and Application Verification

With the RSVP-TE LSP mesh in place, it is now possible to create services and applications to utilize those LSPs. These applications and services include Layer 2 and Layer 3 VPNs, resolution of BGP labeled routes and resolution of BGP, IGP, and static routes. However, the automatically created LSPs are not available for explicit binding in a statically provisioned SDP.

IGP Shortcuts

Figure 68 demonstrates the use of IGP shortcuts, prefix 172.16.32.0/20 is advertised to PE-1 from an external peer in AS 64510, which PE-1 subsequently advertises into IBGP, imposing Next-Hop-Self in the process. For more details on IGP shortcuts refer to [IGP Shortcuts on page 431](#).

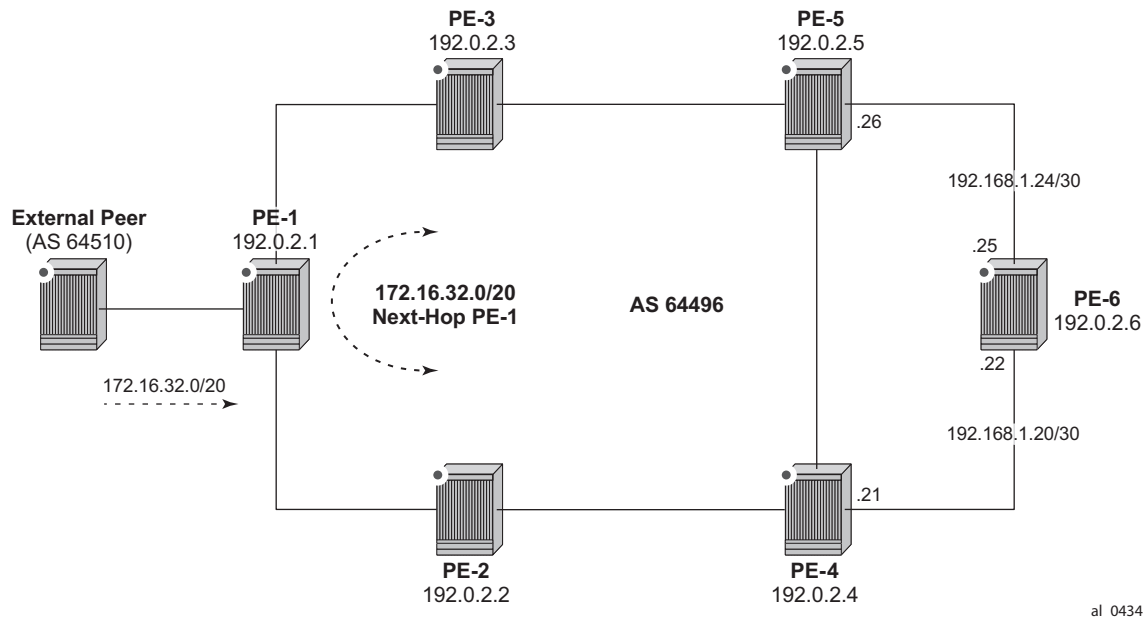


Figure 68: IGP Shortcuts with RSVP-TE Auto-Mesh

The objective is that PE-6 uses the automatically created LSP to PE-1 as an IGP shortcut (typically implemented in order to maintain a “BGP-free” core). IGP shortcuts for BGP are enabled under the main BGP context using the command **next-hop-resolution shortcut-tunnel** with options for **rsvp**, **ldp** or **bgp**¹. Since the test topology only has (automatically created) RSVP-TE LSPs, this option is selected.

```
configure router bgp next-hop-resolution shortcut-tunnel family ?
```

1. There are two more options: sr-isis and sr-ospf. These are related to segment routing, which is beyond the scope of this chapter.


```

- family <family>

<family>                : ipv4

[no] disallow-igp        - Allow/Disallow IGP shortcuts
      resolution        - Configure resolution state of BGP unlabelled routes to tunnels
      resolution-fil* + Configure specific tunnels to be used for resolving BGP unlabelled
routes

configure router bgp next-hop-resolution shortcut-tunnel family ipv4 resolution-filter ?
- resolution-filter

[no] bgp                  - Use BGP tunnelling for next hop resolution
[no] ldp                  - Use LDP tunnelling for next hop resolution
[no] rsvp                  - Use RSVP tunnelling for next hop resolution

configure router bgp
      next-hop-resolution shortcut-tunnel
      family ipv4 resolution-filter rsvp
      family ipv4 resolution filter
exit

```

Once the shortcuts are enabled, the route-table (and FIB) can be validated to ensure that the programmed Next-Hop is the advertising BGP speaker (as opposed to the IGP Next-Hop), and that traffic is tunneled to that Next-Hop through an RSVP LSP. In this case the RSVP LSP is the LSP with Tunnel-Id 61455, which is the LSP to PE-1.

```

*A:PE-6# show router route-table 172.16.32.0/20
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type    Proto    Age           Pref
      Next Hop[Interface Name]                                Metric
-----
172.16.32.0/20                                     Remote  BGP       00h08m16s    170
      192.0.2.1 (tunneled:RSVP:61455)                        0
-----
No. of Routes: 1
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====

```

Layer 3 VPN

Layer 3 VPNs can utilize the automatically created LSPs by using the **auto-bind-tunnel** feature configured with the **rsvp** option (possibly in combination with LDP). The option to include both rsvp and ldp, allows the system to use an RSVP LSP if one exists, and if not to revert to an LDP-based LSP. A VPRN is configured in this manner at PE-1 and PE-6. PE-1 is configured with a loopback address of 172.16.1.1/24 and advertises the VPN IPv4 prefix 172.16.1.0/24 into IBGP, whilst PE-6 is configured with a loopback address of 172.16.6.1/24 and advertises the VPN IPv4 prefix 172.16.6.0/24 into IBGP. The next output illustrates the configuration at PE-6. The only difference at PE-1 is the IP address assigned to the loopback interface.

```
configure
service
    vprn 1 customer 1 create
        route-distinguisher 64496:1
        auto-bind-tunnel
            resolution-filter
                ldp
                rsvp
            exit
        resolution filter
    exit
    vrf-target target:64496:1
    interface "loopback" create
        address 172.16.1.1/24
        loopback
    exit
    no shutdown
exitiv
```

Before a VPN IPv4 prefix is considered **valid** by a receiving SR OS PE router, it must be able to resolve the BGP Next-Hop to an LSP in the tunnel table (if not, the prefix is held in RIB-In and flagged as **invalid**). At PE-6 it is possible to verify that the VPN IPv4 prefix 172.16.1.0/24 received from PE-1 is correctly resolved simply by looking at the VPRN-specific route table. In the output below the VPN IPv4 prefix 172.16.1.0/24 with a Next-Hop of PE-1 (192.0.2.1) is correctly resolved to an RSVP LSP with a Tunnel Id of 61455.

```
*A:PE-6# show router 1 route-table 172.16.1.0/24
=====
Route Table (Service: 1)
=====
```

Dest Prefix[Flags]	Type	Proto	Age	Pref
Next Hop[Interface Name]			Metric	
172.16.1.0/24	Remote	BGP VPN	01h51m58s	170
192.0.2.1 (tunneled:RSVP:61455)			0	

```
-----
No. of Routes: 1
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====
```

Layer 2 VPN

As previously described, automatically created RSVP LSPs cannot be referenced by statically provisioned SDPs. Without the ability for SDPs to explicitly reference automatically created RSVP LSPs, there is little value in manually defining SDPs within Layer 2 service constructs (there is little point in referring to an SDP that cannot bind to the underlying RSVP mesh). Therefore, in order to deliver Layer 2 services there is a requirement to adopt a model within the service construct that permits automatic creation of SDP bindings, and this is achieved using a pseudowire-template dictating the characteristics of the SDP. The secondary effect of using pseudowire-templates to dynamically create SDPs is that these automatically created SDPs can currently only use LDP or BGP as a transport tunnel, not RSVP. The solution is to enable LDP-over-RSVP.

This can be implemented using static provisioning of peers as shown in the next output, or it can be done using automatic creation of T-LDP sessions. Regardless of the method, a reciprocal configuration must exist at both peer endpoints. The static per-peer configuration is applied in the **targeted-session** context and specifies the remote peer system IP address, and the keyword **tunneling**, which enables tunneling of LDP FECs over RSVP LSPs with a far-end address matching that of the T-LDP peer. At a global level, the **prefer-tunnel-in-tunnel** command is shown, but is only required when a next-hop router advertises a FEC over link-level LDP and T-LDP. In this case, by default the system would prefer the link-level LDP tunnel, so the **prefer-tunnel-in-tunnel** instructs the system to prefer an LDP-over-RSVP tunnel if it is available. Although link-layer LDP is not present in the test topology, the command is included because the presence of link-layer LDP is common.

```
configure
router
  ldp
    prefer-tunnel-in-tunnel
    interface-parameters
    exit
    targeted-session
      peer 192.0.2.1
        tunneling
      exit
    exit
  exit
no shutdown
exit
```

The next output provides an example using automatic creation of T-LDP sessions. Here, no explicit reference is made to specific peers, but rather a **peer-template** is configured containing the parameters which apply to all T-LDP sessions spawned by this template. In this example, only the **tunneling** command is required. A **peer-template-map** is then used to create a mapping between the **peer-template** (TLDP-Mesh) and a **policy** defining the IP addresses of remote nodes to which T-LDP sessions should be established. In this example, the policy “Remote-PEs” is the same policy previously used by the auto-created RSVP LSP mesh.

Test Topology

```
configure
  router
    ldp
      prefer-tunnel-in-tunnel
      interface-parameters
      exit
      targeted-session
        peer-template "TLDP-Mesh"
        tunneling
      exit
      peer-template-map peer-template "TLDP-Mesh" policy "Remote-PEs"
    exit
  no shutdown
exit
```

Regardless of whether T-LDP sessions are explicitly provisioned, or dynamically created using a peer-template, the result is that a targeted LDP session is established which can be used for advertising of address and service FECs, and which is capable of tunneling LDP over RSVP.

```
*A:PE-6# show router ldp targ-peer 192.0.2.1 detail
=====
LDP IPv4 Targeted Peers
=====
-----
192.0.2.1
-----
Admin State       : Up           Oper State        : Up
Last Oper Chg     : 0d 00:05:15
Hold Time         : 45           Hello Factor      : 3
Oper Hold Time    : 45
Hello Reduction   : Disabled     Hello Reduction Fact*: 3
Keepalive Timeout : 40           Keepalive Factor  : 4
Active Adjacencies : 1          Last Modified     : 05/04/15 10:37:57
Auto Created      : Yes
Creator           : template     Template Name      : TLDP-Mesh
Tunneling         : Enabled
Lsp Name          : None
Local LSR         : None
BFD Status        : Disabled
=====
No. of IPv4 Targeted Peers: 1
=====
* indicates that the corresponding row element may have been truncated.
*A:PE-6#
```

To create VPLS services using dynamically-created SDPs, BGP Auto-Discovery (BGP-AD) must be used together with LDP (or BGP) pseudowire signaling, for more details see [LDP VPLS using BGP-Auto Discovery on page 1127](#). In the following output PE-6 uses BGP-AD and LDP signaling. (The same configuration is applied at PE-1.) The **vpls-id** is configured in the **bgp-ad** context. The **vpls-id** is a network-wide identifier assigned to all VPLS Switch Instances (VSIs) belonging to the same VPLS, and is carried in VPLS Network Layer Reachability Information (NLRI) as an Extended Community attribute. A second parameter used for BGP-AD and carried in the VPLS NLRI is the VSI-ID, which uniquely identifies each VSI. The VSI-ID is

automatically derived from the global ASN, the VPLS service ID, and the system IP address. To automatically create SDPs, the **bgp** context of the VPLS service refers to a **pw-template** defining the parameters of the pseudowire. In this example, the use of the hash (entropy) label is enabled in the pseudowire template, and a **split-horizon-group**, SHG, is applied.

```
configure
  service
    pw-template 2 create
      hash-label
      split-horizon-group "SHG"
    exit
  exit
  vpls 2 customer 1 create
    bgp
      pw-template-binding 2
    exit
    exit
    bgp-ad
      vpls-id 64496:2
      no shutdown
    exit
    sap 1/1/4:2 create
    exit
    no shutdown
  exit
```

The service information (truncated to show only the relevant information) provides the BGP and BGP-AD operational parameters, and shows that an SDP of type **BgpAd** (17407:4294967294) has been automatically created. Both the SDP and the SAP are operationally up.

```
*A:PE-6# show service id 2 bgp
=====
BGP Information
=====
Vsi-Import           : None
Vsi-Export           : None
Route Dist           : None
Oper Route Dist      : 64496:2
Oper RD Type         : derivedVpls
Rte-Target Import    : None
Rte-Target Export    : None

PW-Template Id       : 2
PW-Template SHG      : None
Oper Group           : None
Mon Oper Group       : None
BFD Template         : None
BFD-Enabled          : no
BFD-Encap            : ipv4
Import Rte-Tgt       : None
-----
*A:PE-6#
*A:P
-----
BGP Auto-discovery Information
-----
Admin State          : Up
Vpls Id              : 64496:2
```

```

Prefix                : 192.0.2.6
-----
*A:PE-6#
*A:PE-6# show service id 2 base | match "Service Access" post-lines 10
Service Access & Destination Points
-----
Identifier                                Type      AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/4:2                             q-tag     1518    1518    Up   Up
sdp:17407:4294967294 SB(192.0.2.1)      BgpAd     0       1548    Up   Up
=====
*A:PE-6#

```

To create Epipe services using dynamically created SDPs, two options exist. Either LDP FEC 129 signaling can be used, which in turn dictates the presence of pseudowire routing information, or BGP-VPWS based signaling can be used, for more details see [BGP Virtual Private Wire Services on page 905](#). This example illustrates the use of BGP-VPWS, but in either case, only single-segment pseudowires are supported. The next output shows the configuration requirements for a basic BGP-based Epipe service at PE-6. Once again a **pw-template** is used to define the characteristics of the pseudowire, and this template is referenced in the **bgp** context of the Epipe service. The **bgp** context is also where the **route-distinguisher** and **route-target** values are configured, which are carried in the VPWS NLRI and Extended Communities respectively. The **ve-name**, **ve-id**, and **remote-ve-name** are all configured in the **bgp-vpws** context. The **ve-id** is carried in the VPWS NLRI, and when a PE router receives a VPWS NLRI to try to establish an Epipe service, the **ve-id** from the NLRI is validated against the **ve-id** configured in the **remote-ve-name**. These must match before the Epipe becomes operational.

```

configure
  service
    pw-template 3 create
      hash-label
    exit
    epipe 3 customer 1 create
      bgp
        route-distinguisher 64496:3
        route-target export target:64496:3 import target:64496:3
        pw-template-binding 3
      exit
    exit
    bgp-vpws
      ve-name "PE-6"
      ve-id 6
    exit
    remote-ve-name "PE-1"
      ve-id 1
    exit
    no shutdown
  exit
  sap 1/1/4:3 create
  exit
  no shutdown
exit

```

The basic service information is truncated to show only the relevant information is used to verify that the service is operational. SDP (17407:4294967293) has been automatically created of type **BgpVpws**. Both the SDP and the SAP are operationally up.

```
*A:PE-6# show service id 3 base | match "Service Access" post-lines 10
Service Access & Destination Points
```

Identifier	Type	AdmMTU	OprMTU	Adm	Opr
sap:1/1/4:3	q-tag	1518	1518	Up	Up
sdp:17406:4294967293 SB(192.0.2.1)	BgpVpws	0	1548	Up	Up

Automatic Creation of RSVP Single-Hop LSPs

As previously discussed, the purpose of a single-hop LSP mesh is to allow for ECMP load-balancing of traffic using LDP-over-RSVP. ECMP load-balancing could be implemented using LDP over a partial or full mesh of RSVP-TE LSPs, but the use of single-hop LSPs additionally allows for load-balancing across a number of parallel RSVP LSPs between nodes. To illustrate ECMP load-balancing over multiple parallel RSVP LSPs the test topology of [Figure 67](#) is modified to include a parallel link between PE-6 and PE-5 as shown in [Figure 69](#). In addition, all routers are enabled for ECMP=2.

```
configure router ecmp 2
```

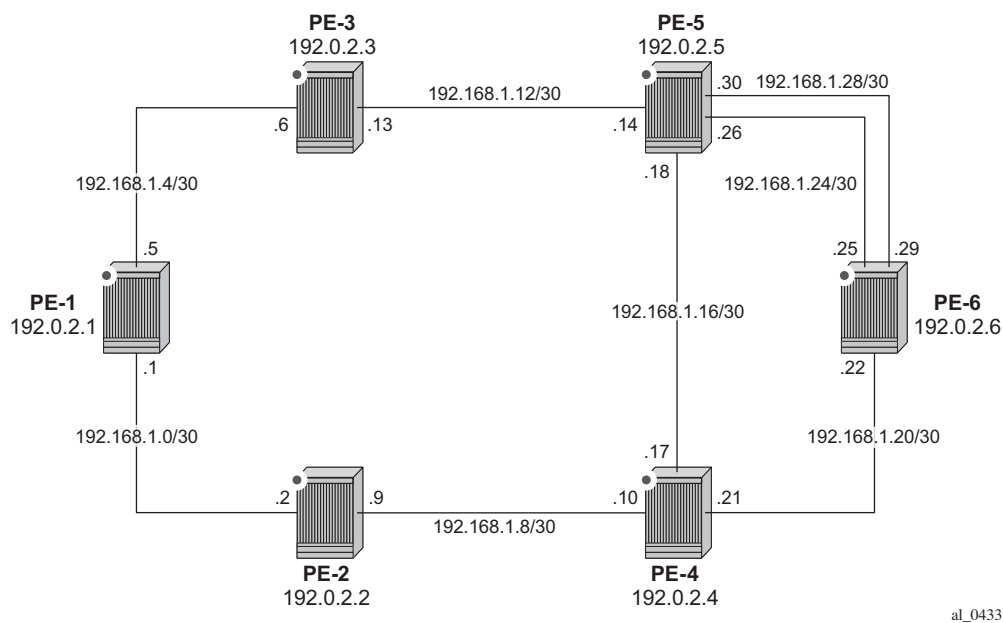


Figure 69: Test Topology for Single-Hop LDP-over-RSVP with ECMP

Unlike the automatically created RSVP-TE LSP mesh previously discussed, the automatically created single-hop RSVP-TE LSPs have no requirement for a prefix-list to be referenced containing the prefixes of the remote nodes that form part of the mesh. In the case of automatically created single-hop LSPs the TE database keeps track of each TE link which comes up to a directly connected IGP neighbor. The system then establishes a single-hop LSP with a destination address matching the router-id of the neighbor and with a strict hop consisting of the address of the interface used by the TE link.

The first requirement is to create an **lsp-template** containing the common parameters used to establish each single-hop LSP. For a single-hop LSP mesh the **lsp-template** must be suffixed with the creation-time attribute **one-hop-p2p**. Upon creation of the template, **cspf** is automatically enabled (and cannot be disabled), and the **hop-limit** is set to a value of **2**. The hop-limit defines the number of nodes the LSP may traverse, and since these are single-hop LSPs to adjacent neighbors a limit of 2 is sufficient. The template must also reference a **default-path** before it can be placed in the no shutdown state. The example below references a path named “loose” that has no strict or loose hops defined. When the RSVP PATH message is actually generated to create the one-hop LSP, it contains one strict-hop to the interface address of the neighbor; and as destination the system address of the adjacent node.

The next requirement is to trigger the creation of single-hop LSPs, and this is achieved using the **auto-lsp lsp-template** command. In this example, the lsp-template “Single-Hop” is referenced, and the command is completed with the keyword **one-hop** to indicate the creation of single-hop LSPs. Unlike an RSVP-TE mesh there is no requirement to reference a route-policy. In the example, the auto-lsp with lsp-template “Full-Mesh” is stopped.

```
configure router mpls no auto-lsp lsp-template "Full-Mesh"

configure
  router
    mpls
      path "loose"
        no shutdown
      exit
      lsp-template "Single-Hop" one-hop-p2p
        default-path "loose"
        cspf
        hop-limit 2
        no shutdown
      exit
      auto-lsp lsp-template "Single-Hop" one-hop
        no shutdown
    exit
  exit
exit
```

Once the **auto-lsp lsp-template** command is entered, the system starts the process of establishing the single-hop LSPs. A check is made of the TE database for every TE link to a directly connected IGP neighbor, and a single-hop LSP is established across each TE link. The output below is taken from PE-6 and shows the automatically created single-hop LSPs. The LSP names are automatically constructed as TemplateName-DestIPv4Address-TunnelId. The LSP name signaled in the Session Attribute object concatenates the LSP name with the path name (for example Single-Hop-192.0.2.4-61499::loose). Recall from [Figure 69](#) that PE-6 has a single TE-enabled link to PE-4, and two TE-enabled links to PE-5, hence with ECMP=2 there is one LSP to PE-4 (192.0.2.4) and two LSPs to PE-5 (192.0.2.5). Note that if ECMP=1 only one single-hop LSP would be signaled to PE-5.

```

*A:PE-6# show router mpls lsp
=====
MPLS LSPs (Originating)
=====
LSP Name                               To                Tun      Fastfail  Adm  Opr
                                Id              Config
-----
Single-Hop-192.0.2.4-61448             192.0.2.4         61448    No        Up   Up
Single-Hop-192.0.2.5-61449             192.0.2.5         61449    No        Up   Up
Single-Hop-192.0.2.5-61450             192.0.2.5         61450    No        Up   Up
-----
LSPs : 3
=====
*A:PE-6#

```

As the purpose of single-hop LSPs is to enable ECMP load-balancing using LDP-over-RSVP, there is a requirement to configure T-LDP sessions between RSVP LSP endpoints. This can be implemented using static peer provisioning, or it can be done using automatic creation of T-LDP sessions, both of which have been previously described and they are therefore not repeated. In this example, the automatic creation of T-LDP sessions approach is used, and T-LDP sessions are created to adjacent neighbors that are capable of tunneling inside RSVP.

```

*A:PE-6# show router ldp session
=====
LDP IPv4 Sessions
=====
Peer LDP Id          Adj Type  State          Msg Sent  Msg Recv  Up Time
-----
192.0.2.1:0          Targeted  Established    1009      1008      0d 01:29:07
192.0.2.2:0          Targeted  Established    1027      1026      0d 01:30:44
192.0.2.3:0          Targeted  Established    1026      1026      0d 01:30:45
192.0.2.4:0          Targeted  Established    1029      1026      0d 01:30:45
192.0.2.5:0          Targeted  Established    1029      1030      0d 01:30:58
-----
No. of IPv4 Sessions: 5
=====
LDP IPv6 Sessions
=====
Peer LDP Id
Adj Type          State          Msg Sent  Msg Recv  Up Time
-----
No Matching Entries Found
=====
*A:PE-6#

```

To validate the ECMP load-balancing capability, PE-5 is configured to advertise prefix 172.16.5.0/24 to PE-6. In turn, PE-6 is configured for **ibgp-multipath** to enable load-balancing over IGP links to the BGP Next-Hop address, **next-hop-resolution shortcut-tunnel resolution-filter ldp** to enable tunneling of traffic destined towards the BGP Next-Hop in MPLS, and **ecmp 2**.

```

configure router bgp
    ibgp-multipath
    next-hop-resolution
    shortcut-tunnel
    family ipv4
        resolution-filter
        ldp
    exit
    resolution filter
exit
exit
exit
no shutdown

```

The prefix 172.16.5.0/24 advertised by PE-5 is learned at PE-6 and installed in the RIB/FIB with PE-5's system address (192.0.2.5) as next-hop.

```

*A:PE-6# show router bgp routes 172.16.5.0/24
=====
BGP Router ID:192.0.2.6          AS:64496          Local AS:64496
=====
Legend -
Status codes  : u - used, s - suppressed, h - history, d - decayed, * - valid
                  l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP IPv4 Routes
=====
Flag  Network                      LocalPref  MED
      Nexthop (Router)              Path-Id    Label
      As-Path
-----
u*>i  172.16.5.0/24                  100        None
      192.0.2.5                      None        -
      No As-Path
-----
Routes : 1
=====
*A:PE-6

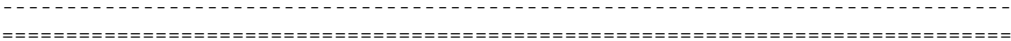
```

Checking the FIB for the Next-Hop address 192.0.2.5, it can be verified that both links are installed as Next-Hop addresses, meaning that ECMP load-balancing is effective.

```

*A:PE-6# show router fib 1 192.0.2.5/32
=====
FIB Display
=====
Prefix [Flags]                      Protocol
      Nexthop
-----
192.0.2.5/32                        ISIS
      192.168.1.26 (int-PE-6-PE-5)
      192.168.1.30 (int-PE-6-PE-5-2nd)
-----
Total Entries : 1

```



Conclusion

Automatic creation of RSVP-TE LSPs provides a good solution for reducing the amount of provisioning activity required when configuring RSVP LSPs. However, there are some constraints with regard to the way that services are deployed on top of those LSPs. SDPs cannot explicitly reference automatically created RSVP LSPs, which means that automatically created SDPs need to be used for Layer 2 services. In turn, automatically created SDPs can only use LDP or BGP as a transport tunnel (not RSVP), therefore in order to use the automatically created RSVP mesh, LDP over RSVP must be used. These caveats need to be fully understood before considering deployment of automatically created RSVP-TE LSPs.

In This Chapter

This section describes advanced BGP anycast configurations.

Topics in this section include:

- [Applicability on page 388](#)
- [Summary on page 389](#)
- [Overview on page 390](#)
- [Configuration on page 395](#)
- [Conclusion on page 429](#)

Applicability

The configuration in this chapter is based on SR OS 9.0R1. The feature is supported on 7750 SR-c4/c12, SR-7 and SR-12 in chassis mode **d**, and 7450 ESS-7 and ESS-12 in mixed-mode.

Summary

Release 8.0.R4 and higher provide a resiliency mechanism to protect the data flow of Layer 2/ Layer 3 traffic in the event of a complete nodal failure of the PE or ABR routers. The intent is to allow traffic to continue to flow during the convergence process and as a result reduce the amount of traffic lost during this process. This is achieved by allowing two designated routers to back each other up such that both routers store the forwarding information learned from the alternate router in a secondary forwarding table, also referred to as context-specific label space. In the event that the primary router fails, the secondary router will continue to forward traffic to the ultimate destination using the forwarding information stored in the secondary forwarding table (context-specific label space). This will continue until the rest of the network can converge to use an alternate route through the network.

BGP anycast can be used in two different scenarios.

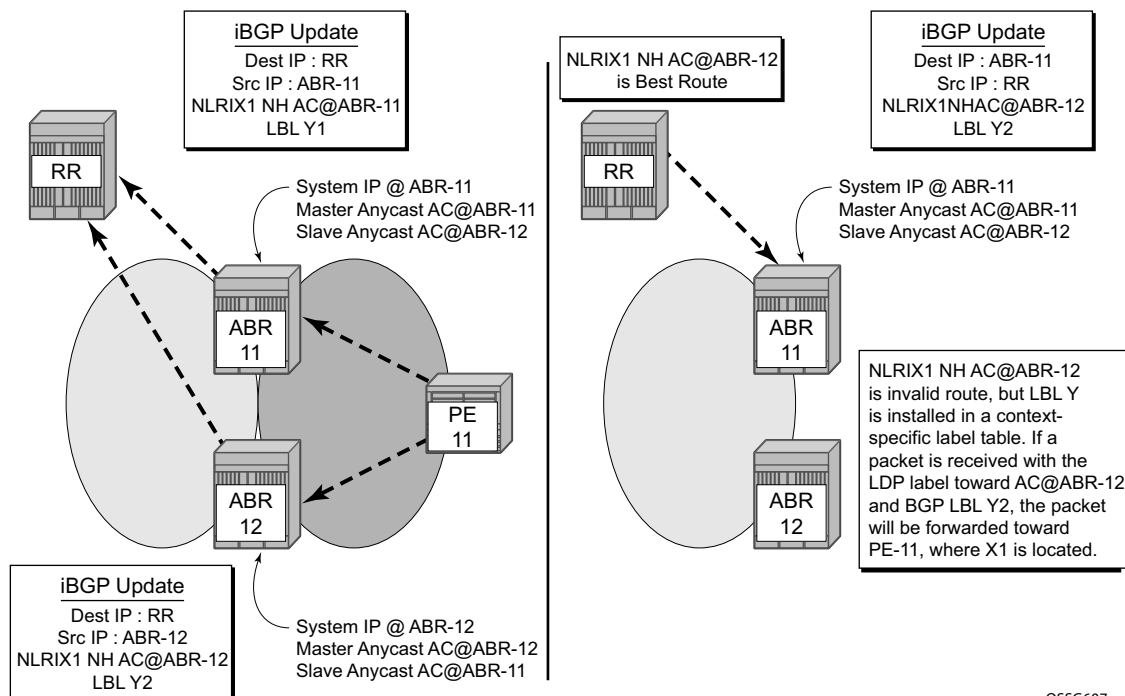
1. BGP to BGP label swapping in GRT — Where the LDP and BGP labels of an incoming IP packet will be swapped, and the packet being forwarded towards a destination in the related region/area.
2. BGP to VRF — Where an incoming packet, encapsulated with the redundant BGP service label will be forwarded towards the related VRF where an (egress) IP lookup will be performed so that the packet can be forwarded towards the correct CPE.

Overview

GRT

Assume that an access node (or similar) is connected to PE-11, represented by prefix X1/32, illustrated in [Figure 70](#)).

1. PE-11 will advertise a BGP labelled route towards its area border routers (ABRs) with its own system-IP as next hop (NH) (regular behavior).
2. Both redundant ABR (11 and 12), acting as route reflectors (RRs) for their own region, will advertise the route X1 towards the RR of the core, with their master anycast (AC) address as an NH.
3. The RR (of the core) will select the best route out of both, in this case originated from ABR-12 and reflects the route to all of the ABRs in the core.
4. The redundant ABR-11 will receive this BGP route and marks it as invalid since the NH will be a local interface, being the slave anycast interface.
5. ABR-11 will, however, install the BGP label in the context specific label space, so that incoming packets with the BGP label assigned by ABR-12 (step 2) and kept by ABR-11 (following upon the outer LDP label) can be forwarded towards destination X1.



O5SG607

Figure 70: BGP Anycast Operation in GRT

Note that [Figure 70](#) shows that X1 will be advertised by both (redundant) ABRs, each assigning their master anycast address as the NH. RR will select one of them as best and will reflect the route towards both ABRs. Only one ABR (ABR-11 in this case) will install the BGP anycast route towards the destination X1. This is fine since ABR-12 will be selected by the other ABR (connected to remote regions) in the core as NH for the route towards X1/32, hence ABR-12 will receive the traffic from other regions towards X1, and if there is a failure on ABR-12 (link/nodal) traffic will be diverted to ABR-11. ABR-11 can interpret the LDP and BGP labels, which are advertised by ILDP and the BGP label (in the context-specific label space), which results in forwarding traffic to X1.

IP-VPN

In case of IP-VPN routes, the scenario is different since unique route distinguisher (RD) will result in both IP-VPN routes being active at the RR, as illustrated [Figure 71](#).

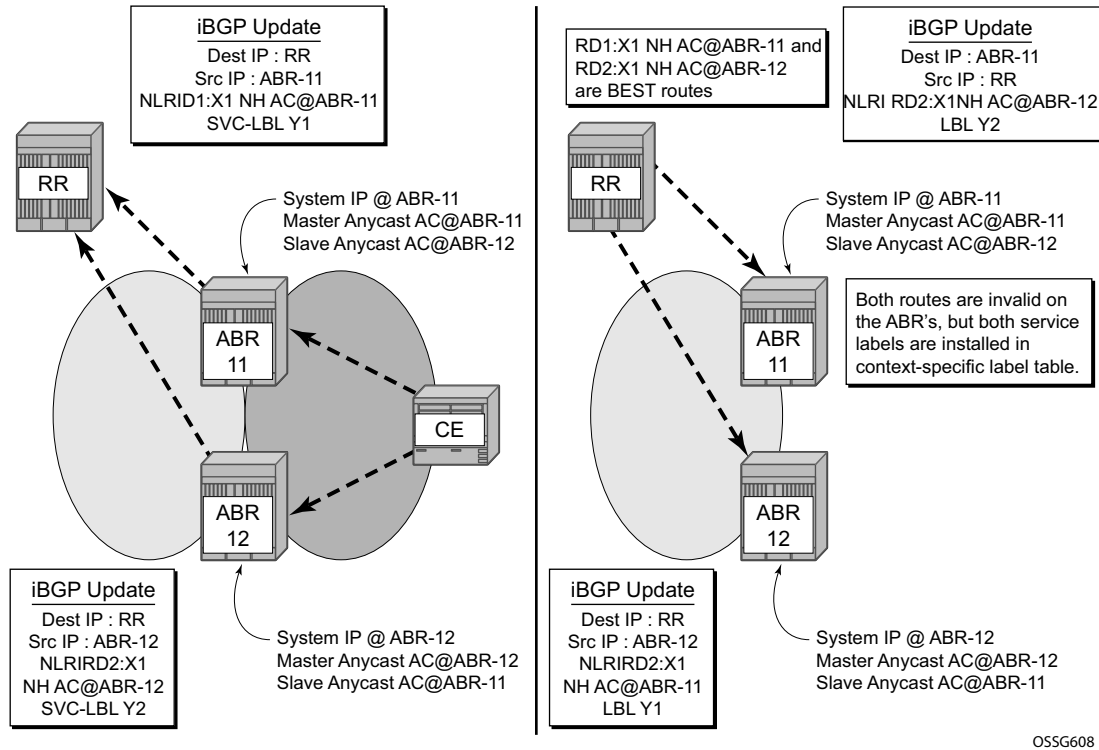


Figure 71: BGP Anycast Operation with IP-VPN

Context-Specific Label Space

A key-point of the BGP anycast feature is the presence of a context-specific label space on the Label Switch Router (LSR)/ABR. Where in normal MPLS scenarios, an LSR selects a free label from its own label-range, installs this in the Label Forwarding Information Base (LFIB) and signals this by a control protocol (BGP/LDP/RSVP), the LSR/ABR will now learn a label from a redundant LSR/ABR and installs this as an ingress label into a dedicated label space. This mechanism is different from the MPLS mechanisms used to date in the SR OS, which are based on downstream label allocation.

Data Path in GRT

Figure 72 illustrates the data path from source S to destination X1.

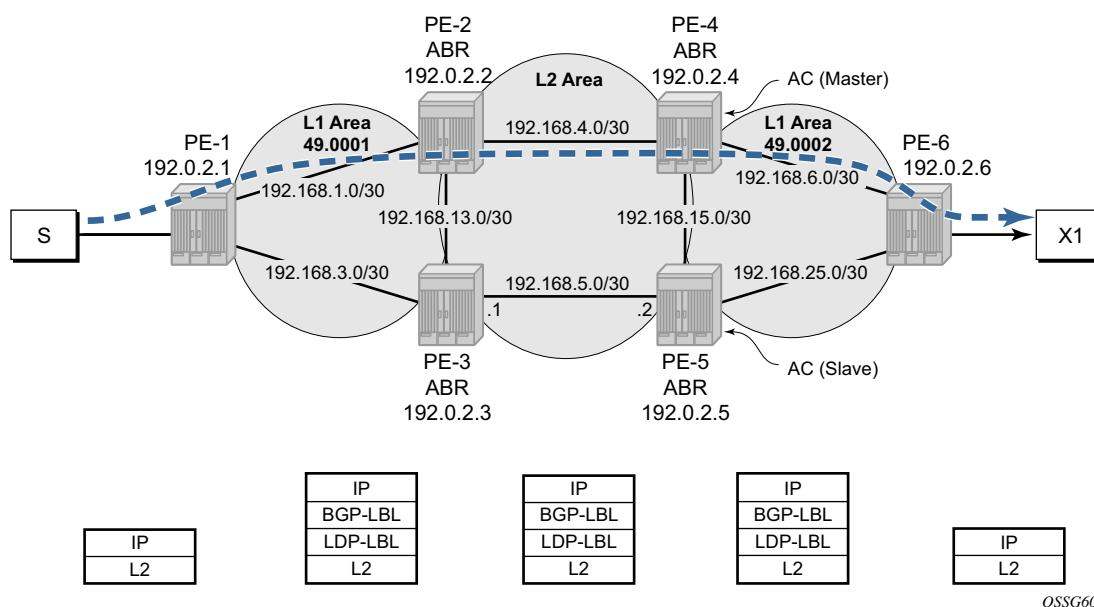


Figure 72: BGP Anycast Data Path (BGP to BGP Swapping)

An incoming IP packet on PE-1 will be encapsulated in a dual MPLS label packet, where;

1. BGP label will assure the forwarding towards the remote PE, being PE-6 in this case, PE-6 is part of the tunnel-table on PE-1, with BGP as MPLS control protocol.
2. LDP label will forward the packet to the anycast address of area 49.0002, where PE-4 is the master (in normal operations).

3. From PE-4 onwards, both the BGP and LDP labels are swapped, and forwarded out on the interface towards PE-6.

The data path for IP-VPN routes is described in the IP-VPN routes section.

BGP Control Plane

Additional attention is required for the BGP control plane, where the ABR needs to act as RR for its related region. This is mandatory since the ABR needs to perform a NHS (next hop self) action to the BGP routes advertising the system addresses of the PE (or AN) in the region.

Note that different RR hierarchies can be created based upon different address families.

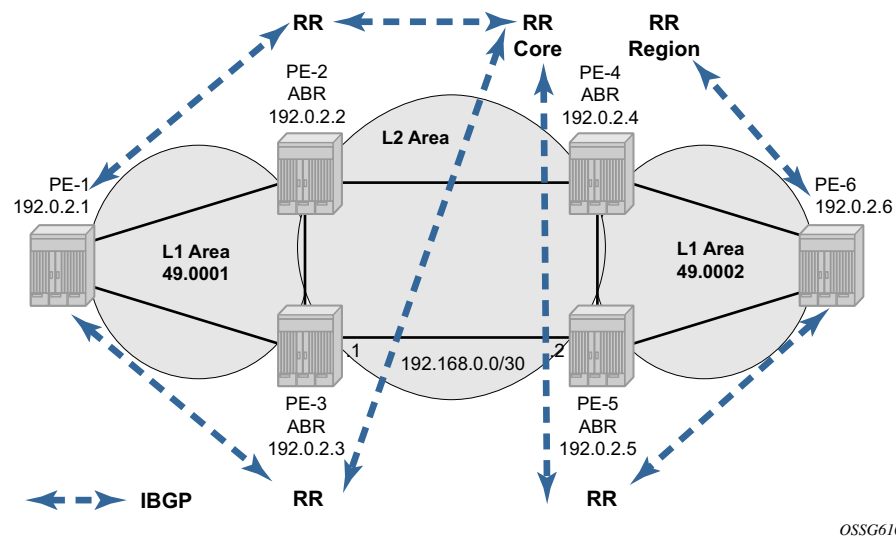


Figure 73: BGP Anycast BGP Topology

In this particular case, there is a cluster in the core area with PE-4 acting as RR. PE-4 is RR for 2 different clusters, 1.1.1.1 towards the core and 2.2.2.2 towards the regional PE-6. All other PEs (PE-2-3-5) in the core act as an RR for their related region.

Configuration

The following steps must be taken to configure the BGP anycast scenario.

1. Create the anycast interfaces.
2. IGP configuration in the MPLS domain.
3. Enable LDP to assure reach ability of the ABR.
4. Configure BGP to advertise the system addresses of PE-1 and PE-6 with a BGP label.
5. Configure a service to set-up end-to-end connectivity.

Anycast Interface

Each ABR will be configured with two anycast addresses, a master and slave address. The redundant ABR within the same area will be configured with the reverse, as seen in the following configuration;

On PE-2:

```
A:PE-2>config>router# info
-----
#-----
echo "IP Configuration"
#-----

      mh-primary-interface "masterAnycast"
        address 10.20.0.1/32
      exit
      mh-secondary-interface "slaveAnycast"
        address 10.30.0.1/32
```

On PE-3:

```
*A:PE-3>config>router# info
-----
#-----
echo "IP Configuration"
#-----

      mh-primary-interface "masterAnycast"
        address 10.30.0.1/32
      exit
      mh-secondary-interface "slaveAnycast"
        address 10.20.0.1/32
      exit
```

Both PE-2 and PE-3 will have 10.20.0.1 as one of their anycast addresses, but only PE-2 has it configured as the master anycast address. Conversely both PE-2 and PE-3 have 10.30.0.1/32 as

one of their anycast addresses, but only PE3 has it configured as the master anycast address. Both interfaces will be considered as a kind of virtual interface, with a virtual port-id;

```
*A:PE-3>config>router# show router interface "masterAnycast" detail | match Port
Port Id          : vport-7
*A:PE-3>config>router#
```

Both anycast addresses, master and slave will be visible in the GRT (global routing table) of the PE (ABR);

```
*A:PE-2>config>router# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix          Type      Proto    Age      Pref
Next Hop [Interface Name]      Metric
-----
10.20.0.1/32         Local    Local    01d14h16m  0
    masterAnycast          0
10.30.0.1/32         Local    Local    01d14h16m  0
    slaveAnycast           0
```

As similar configuration is performed on PE-4 and PE-5 with the anycast addresses being (Figure 74):

```
PE-4: masterAnycast = 10.40.0.1/32
      slaveAnycast  = 10.50.0.1/32
PE-5: masterAnycast = 10.50.0.1/32
      slaveAnycast  = 10.40.0.1/32
```

Therefore, if PE-4 fails, for example, PE-5 will take over the responsibility for PE-4's master anycast address (10.40.0.1); this will cause PE-2 and PE-3 to continue to send traffic which is directed towards the BGP routes with a next hop of 10.40.0.1 (PE-4's masterAnycast address) thinking its next hop is PE-4 when in reality it is PE-5. The failover initially only involves an IGP (ISIS) re-convergence, and hence an LDP re-convergence, instead of a BGP re-convergence which would take a longer time and therefore cause a larger outage (the IGP re-convergence may be reduced if ECMP or LDP FRR is used and an alternate label is already installed in the LFIB of PE-2/PE-3). BGP will subsequently re-converge such that the BGP next hop on PE-2 and PE-3 becomes 10.50.0.1 when they receive the withdraw of PE-4's 10.40.0.1/32 from the RR.

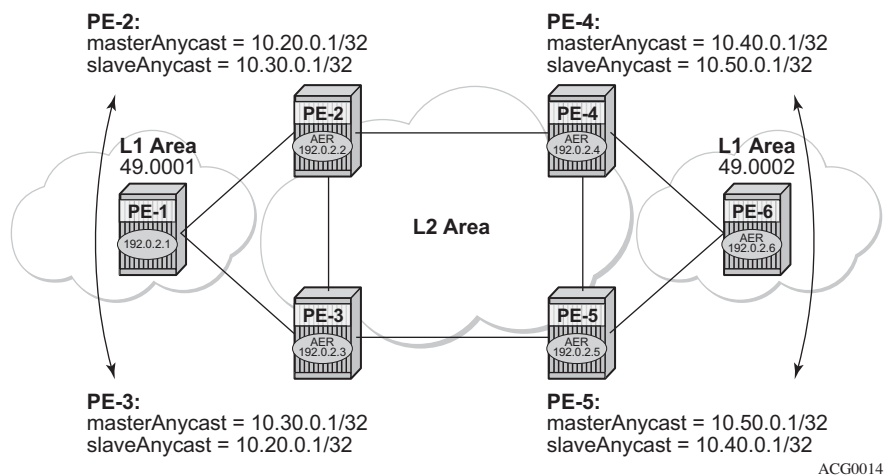


Figure 74: Anycast Address Configuration

IGP

Since both anycast addresses (master and slave) need to be reachable from the remote ABR and all remote PEs, they have to be advertised into the IGP (ISIS in this case). This will result in following ISIS configuration;

```
*A:PE-2>config>router>isis# info
-----
    area-id 49.0001
    export "remoteABR"
    traffic-engineering
    interface "system"
        passive
    exit
    interface "masterAnycast"
        level-capability level-2
    exit
    interface "slaveAnycast"
        level-capability level-2
    exit
    interface "int-PE-2-PE-3"
        level-capability level-2
        interface-type point-to-point
    exit
    interface "int-PE-2-PE-4"
        level-capability level-2
        interface-type point-to-point
    exit
    interface "int-PE-2-PE-1"
        level-capability level-1
        interface-type point-to-point
    exit
-----
*A:PE-2>config>router>isis#
```

Here, notice that the master and slave interface will be part of the Layer 2 area. Both interfaces will automatically be assigned with the correct ISIS metric, being 10 for the master and 30 for the slave (default values).

```
*A:PE-2>config>router>isis# show router isis interface
=====
ISIS Interfaces
=====
Interface                               Level CircID Oper State  L1/L2 Metric
-----
system                                  L1L2   1      Up      0/0
masterAnycast                          L2     2      Up      -/10
slaveAnycast                           L2     3      Up      -/30
```

The IGP (ISIS) metric can be configured to a different value if required.

The ISIS export policy is used to control redistribution of prefixes and essentially serves two purposes.

1. Ensure that anycast addresses and system IP addresses of other ABRs learned in Layer 2 LSPs through the core (from the perspective of PE-2, this would include PE-4 and PE-5) are re-advertised to PEs in the same area as Layer 1 LSPs.
2. Ensure that system IP addresses of PEs in the same region as the ABR learned through L1 LSPs are not advertised into the core as Layer 2 LSPs. The reason for this is that it is mandatory that a remote ABR has a best route to a remote PE (in a different area) through BGP and not by an IGP route in the FIB, otherwise it would not advertise the labeled BGP route toward PEs in its own region.

```
*A:PE-2>config>router>policy-options# info
```

```
-----
prefix-list "PE"
  prefix 192.0.2.1/32 exact
exit
prefix-list "remoteABR"
  prefix 10.40.0.1/32 exact
  prefix 10.50.0.1/32 exact
  prefix 192.0.2.4/32 exact
  prefix 192.0.2.5/32 exact
exit
prefix-list "masterAnycast"
  prefix 10.20.0.1/32 exact
exit

policy-statement "remoteABR"
  entry 10
    from
      prefix-list "remoteABR"
    exit
    action accept
    exit
  exit
  entry 20
    description "reject system-ip of own region"
    from
      prefix-list "PE"
    exit
    action reject
  exit
exit
```

LDP

All physical interfaces need to be part of ILDP so that the BGP label can be encapsulated in an MPLS packet with an LDP label. The configuration on PE-2 looks like the following.

```
*A:PE-2>config>router>ldp# info
-----
      export "exp-anycast"
      interface-parameters
        interface "int-PE-2-PE-3"
        exit
        interface "int-PE-2-PE-4"
        exit
        interface "int-PE-2-PE-1"
        exit
      exit
      targeted-session
      exit
-----
* A:PE-2>config>router>ldp# show router policy "exp-anycast"
  entry 10
    description "advertise master anycast address"
    from
      prefix-list "masterAnycast"
    exit
    to
      protocol ldp
    exit
    action accept
    exit
  exit
  entry 20
    description "advertise slave anycast address as well"
    from
      prefix-list "slaveAnycast"
    exit
    to
      protocol ldp
    exit
    action accept
    exit
  exit
A:PE-2>config>router>ldp#
```

By default, only the system address will be advertised by ILDP; hence the need for the export policy. This policy will advertise both the master and slave anycast address. The slave address advertisement is needed in both ABRs so that PE-2 can take over the role from the master (PE-3) in case of link or node failures (and vice versa).

To verify the LDP bindings, check if PE-2 can reach the remote anycast addresses through LDP, but not its own master and slave anycast address.

```

A:PE-2>config>router>ldp# show router ldp bindings
=====
LDP LSR ID: 192.0.2.2
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
        S - Status Signaled Up, D - Status Signaled Down
        E - Epipe Service, V - VPLS Service, M - Mirror Service
        A - Apipe Service, F - Fpipe Service, I - IES Service, R - VPRN service
        P - Ipipe Service, WP - Label Withdraw Pending, C - Cpipe Service
        TLV - (Type, Length: Value)
=====
LDP Prefix Bindings
=====
Prefix                Peer                IngLbl    EgrLbl  EgrIntf/  EgrNextHop
                        Peer                LspId
-----
10.20.0.1/32          192.0.2.1          262140U   --      --        --
10.20.0.1/32          192.0.2.4          262140U   --      --        --
10.30.0.1/32          192.0.2.1          262139U   --      --        --
10.30.0.1/32          192.0.2.4          262139U   --      --        --
10.40.0.1/32          192.0.2.1          262143U   262143   --        --
10.40.0.1/32          192.0.2.3          262143U   262135   --        --
10.40.0.1/32          192.0.2.4          --        262139   1/1/3:0   192.168.4.2
10.50.0.1/32          192.0.2.1          262130U   262138   --        --
10.50.0.1/32          192.0.2.3          262130N   262142   1/1/2:0   192.168.13.2
10.50.0.1/32          192.0.2.4          --        262143   --        --
192.0.2.1/32          192.0.2.1          --        262142   1/1/1:0   192.168.2.1
192.0.2.1/32          192.0.2.3          262134U   262132   --        --

```

PE-2 can reach the remote anycast addresses 10.40.0.1 and 10.50.0.1. In this case both addresses are reachable by a different interface. Note that it can be via the same interface as well, due to the ECMP nature in the (Layer 2 area) square topology, where 10.50.0.1 can be reached via PE-4 as well as PE-2. There is no outgoing label for 10.20.0.1 and 10.30.0.1 which is to be expected since they are local addresses on PE-2.

To verify the actual datapath, following OAM command can be used.

```

*A:PE-2# oam lsp-trace prefix 10.40.0.1/32
lsp-trace to 10.40.0.1/32: 0 hops min, 0 hops max, 104 byte packets
1 192.0.2.4 rtt=2.02ms rc=3(EgressRtr)
*A:PE-2# oam lsp-trace prefix 10.50.0.1/32
lsp-trace to 10.50.0.1/32: 0 hops min, 0 hops max, 104 byte packets
1 192.0.2.3 rtt=2.76ms rc=8(DSRtrMatchLabel)
2 192.0.2.5 rtt=3.57ms rc=3(EgressRtr)
*A:PE-2#

```

BGP Configuration

The configuration of BGP will be split up in following parts;

1. BGP configuration on the ABR
2. BGP configuration on the RR
3. BGP configuration on the PE

BGP on the ABR

The following BGP configuration is present on PE-2;

```
*A:PE-2# configure router bgp
*A:PE-2>config>router>bgp# info
-----
      min-route-advertisement 10
      group "region"
        description "all PE of the region will peer with this ABR"
        family ipv4 vpn-ipv4
        type internal
        cluster 2.2.2.2
        neighbor 192.0.2.1
          advertise-label ipv4
        exit
      exit
      group "fullmesh"
        description "PE-4 is RR"
        family ipv4 vpn-ipv4
        type internal
        export "exp-anycast-nhs"
        neighbor 192.0.2.4
          advertise-label ipv4
        exit
      exit
-----
*A:PE-2>config>router>bgp#
```

The two groups identify the PE in the local region (part of L1 area) and the RR in the core (part of Layer 2 area).

The export policy will set the next-hop of the IPv4 and IPv4-VPN routes equal to the master anycast address;

```
A:PE-2>config>router>bgp# show router policy "exp-anycast-nhs"
  entry 10
    description "set NH of PE to master anycast"
    from
      prefix-list "PE"
      family ipv4
    exit
```

```

    to
      protocol bgp
    exit
    action accept
      local-preference 150 (*)
      next-hop 10.20.0.1
    exit
  exit
entry 20
  description "set NH of IP-VPN routes to master anycast"
  from
    family vpn-ipv4
  exit
  to
    protocol bgp-vpn
  exit
  action accept
    local-preference 150 (*)
    next-hop 10.20.0.1
  exit
exit
A:PE-2>config>router>bgp#

```

(*) optional, not mandatory for functional testing.

For testing purpose, this policy is sufficient. In reality more restrictions will be needed to avoid obsolete advertisements of routes by the ABR, but this is out-of-scope for this document.

BGP on the RR

BGP on RR (PE-4) will look like the following:

```
A:PE-4>config>router>bgp# info
```

```
-----
vpn-apply-import
vpn-apply-export
min-route-advertisement 1
group "region"
    description "all PE of the region will peer with this ABR"
    family ipv4 vpn-ipv4
    type internal
    cluster 4.4.4.4
    neighbor 192.0.2.6
        advertise-label ipv4
    exit
exit
group "fullmesh"
    description "RR in cluster 1.1.1.1"
    family ipv4 vpn-ipv4
    type internal
    cluster 1.1.1.1
    export "exp-anycast-nhs"
    neighbor 192.0.2.2
        advertise-label ipv4
    exit
    neighbor 192.0.2.3
        advertise-label ipv4
    exit
    neighbor 192.0.2.5
        advertise-label ipv4
    exit
exit
-----
```

```
A:PE-4>config>router>bgp# show router policy "exp-anycast-nhs"
```

```
entry 10
    description "set NH of PE to master anycast"
    from
        prefix-list "PE"
        family ipv4
    exit
    to
        protocol bgp
    exit
    action accept
        local-preference 150
        next-hop 10.40.0.1
    exit
exit
```

```
A:PE-4# show router policy prefix-list "PE"
prefix 192.0.2.6/32 exact
prefix 192.168.0.6/32 exact
A:PE-4#
```


Where the anycast address is set as the BGP NH in all IPv4 prefixes for all PEs in the attached area/region. The BGP NH is not set for IP-VPN routes, as this would override the NH in advertised IP-VPN routes of PE-2 and PE-3 for which PE-4 acts as RRs and should therefore not modify the NH of the IP-VPN routes.

BGP on the PE

Each PE will have a peering towards each ABR of its own region. In this case PE-1 will peer with PE-2 and PE-3, which in turn act as RRs so there are no inter-PE peering sessions required within the region.

```
*A:PE-1# configure router bgp
*A:PE-1>config>router>bgp# info
-----
min-route-advertisement 10
group "abr"
  description "peering to ABR, which act as RR"
  family ipv4 vpn-ipv4
  type internal
  export "expbgp"
  neighbor 192.0.2.2
    advertise-label ipv4
  exit
  neighbor 192.0.2.3
    advertise-label ipv4
  exit
exit
-----
A:PE-1>config>router>bgp#
```

The referenced export policy triggers the advertisement of the PE-1 system IP address with a BGP-label.

```
A:PE-1# show router policy "expbgp"
  entry 10
    from
      prefix-list "PE"
    exit
  to
    protocol bgp
  exit
  action accept
  exit
exit

A:PE-1# show router policy prefix-list "PE"
prefix 192.0.2.1/32 exact
A:PE-1#
```

Data Path Verification

The best way to verify the data path is between both (remote) PEs.

Looking at PE-1, verify the following:

1. Was a BGP route received toward PE-6 with an NH equal to an anycast address?

```
A:PE-1# show router bgp routes 192.0.2.6/32
=====
BGP Router ID:192.0.2.1          AS:65536          Local AS:65536
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best
=====
BGP IPv4 Routes
=====
Flag  Network                      LocalPref  MED
      Nexthop                      VPNLabel
      As-Path
-----
u*>i  192.0.2.6/32                  150        10
      10.40.0.1                    -
      No As-Path
*i    192.0.2.6/32                  150        10
      10.50.0.1                    -
      No As-Path
-----
Routes : 2
=====
A:PE-1#
```

Note that both routes are in fact equal from a BGP point of view. If ECMP and (BGP) multipath would have been set to 2, both routes would be active in the RTM.

2. Can the anycast addresses be reached through LDP?

```
*A:PE-1# show router ldp bindings prefix 10.40.0.1/32
=====
LDP LSR ID: 192.0.2.1
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
      WP - Label Withdraw Pending
=====
LDP Prefix Bindings
=====
Prefix          Peer          IngLbl    EgrLbl  EgrIntf    EgrNextHop
-----
10.40.0.1/32    192.0.2.2      262143N   262143  1/1/1:0    192.168.2.2
10.40.0.1/32    192.0.2.3      262143U   262135  --         --
-----
No. of Prefix Bindings: 2
=====
*A:PE-1# oam lsp-trace prefix 10.40.0.1/32
lsp-trace to 10.40.0.1/32: 0 hops min, 0 hops max, 104 byte packets
```

```

1 192.0.2.2 rtt=2.18ms rc=8(DSRtrMatchLabel)
2 192.0.2.4 rtt=2.88ms rc=3(EgressRtr)
*A:PE-1#

```

There are two labels learned and one is active since the shortest path to PE-4 is via PE-2, as seen in the OAM command.

3. Verify that the ABR has learned the anycast advertisements of the neighboring master. This is needed to evaluate the incoming labelled packets correctly after failures in the Layer 2 area (link or node).

On PE-2, observe the presence of the anycast label in the context-specific label space, with a link to VPRN-id 1. This is described in [IP-VPN Routes on page 414](#).

```

A:PE-2# show router bgp anycast-label
=====
BGP Anycast-MH labels
=====
Secondary-MH-Addr      ABR-Lbl    Cfg-Time  VPRN-ID
PE-Addr               PE-Lbl      Rem-Time  Ref-Count
-----
10.30.0.1             262130      30        1
-                     -           -         3
=====
A:PE-2#

```

On the redundant ABR PE-3, an additional label-mapping is present because the route 192.168.0.1/32 (part of GRT) is received on PE-3 with the anycast NH of PE-2. PE-2 does not receive this route from PE-3 because the RR (PE-4) has selected the route 192.168.0.1/32 from PE-2 as the best one, which means that PE-2 will not receive the route to 192.168.0.1/32 with PE-3 anycast address as NH.

```

A:PE-3# show router bgp anycast-label
=====
BGP Anycast-MH labels
=====
Secondary-MH-Addr      ABR-Lbl    Cfg-Time  VPRN-ID
PE-Addr               PE-Lbl      Rem-Time  Ref-Count
-----
10.20.0.1             262131      30        -
192.0.2.1             262134      -         1

10.20.0.1             262133      30        1
-                     -           -         3
=====
A:PE-3#

```

This command can be used to check the mapping of BGP labels in the context-specific label space.

More details are provided [Deployment Options on page 409](#).

4. Verify data path towards PE-6, where some basic CLI commands can be used;

```

*A:PE-1# show router route-table 192.0.2.6/32
=====
Route Table (Router: Base)
=====
Dest Prefix                                Type    Proto    Age          Pref
  Next Hop[Interface Name]                Metric
-----
192.0.2.6/32                               Remote  BGP      00h30m50s    170
  10.40.0.1 (tunneled)                     0
-----
No. of Routes: 1
=====
*A:PE-1#
*A:PE-1# show router fib 1 192.0.2.6/32
=====
FIB Display
=====
Prefix                                Protocol
  NextHop
-----
192.0.2.6/32                           BGP
  10.40.0.1 (Transport:LDP)
-----
Total Entries : 1
=====
...
*A:PE-1# show router tunnel-table
=====
Tunnel Table (Router: Base)
=====
Destination      Owner Encap TunnelId  Pref    Nexthop      Metric
-----
192.0.2.6/32     bgp   MPLS   -         10      10.40.0.1    1000
=====
*A:PE-1# traceroute 192.0.2.6 no-dns
traceroute to 192.0.2.6, 30 hops max, 40 byte packets
  1  192.0.2.6    28.9 ms  4.53 ms  5.00 ms
*A:PE-1#

```

Deployment Options

There are two main deployment options with BGP anycast.

1. Providing an end-to-end (E2E) MPLS tunnel, between access nodes (ANs) connected to the PE. In this case, VLLs can be supported between AN over a BGP signalled SDP. This solves scaling issues and also provides resiliency.
2. Use the anycast address as the NH for IP-VPN routes, which improves NH tracking in large MPLS domains, where IGP convergence can not contribute to the NH convergence.

E2E MPLS Between Access Nodes

The AN will be simulated by loopback interfaces on PE-1 and PE-6. From a functional point of view this is equivalent to a connected AN that is not part of the IGP domain.

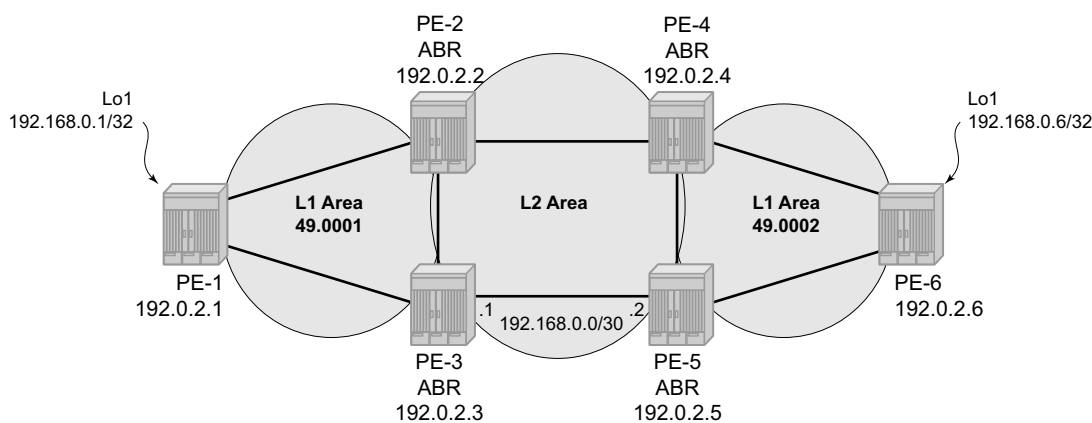


Figure 75: E2E MPLS Between Access Nodes

The loopback interface Lo1 is not part of the IGP (ISIS) but will be advertised by BGP towards the ABR with a BGP label. The configuration looks like the following.

```
*A:PE-1>config>router# info
-----
#-----
echo "IP Configuration"
#-----

interface "lo1"
  address 192.168.0.1/32
  description "simulate an AN (access node) "
```

```
        loopback
    exit
    interface "system"
        address 192.0.2.1/32
    exit
    autonomous-system 64496
```

BGP Configuration

The BGP configuration on the regional PE will look like the following:

```
A:PE-1# configure router bgp
A:PE-1>config>router>bgp# info
-----
    vpn-apply-import
    vpn-apply-export
    min-route-advertisement 1
    group "abr"
        description "peering to ABR, which act as RR"
        family ipv4 vpn-ipv4
        type internal
        export "expbgp"
        neighbor 192.0.2.2
            advertise-label ipv4
        exit
        neighbor 192.0.2.3
            advertise-label ipv4
        exit
    exit
-----

A:PE-1>config>router>bgp# show router policy "expbgp"
    entry 10
        description "advertise the PE system-ip through BGP"
        from
            prefix-list "PE"
        exit
        to
            protocol bgp
        exit
        action accept
            local-preference 200
        exit
    exit
    entry 20
        description "advertise connect AN through BGP"
        from
            prefix-list "AN"
        exit
        to
            protocol bgp
        exit
        action accept
        exit
    exit
```

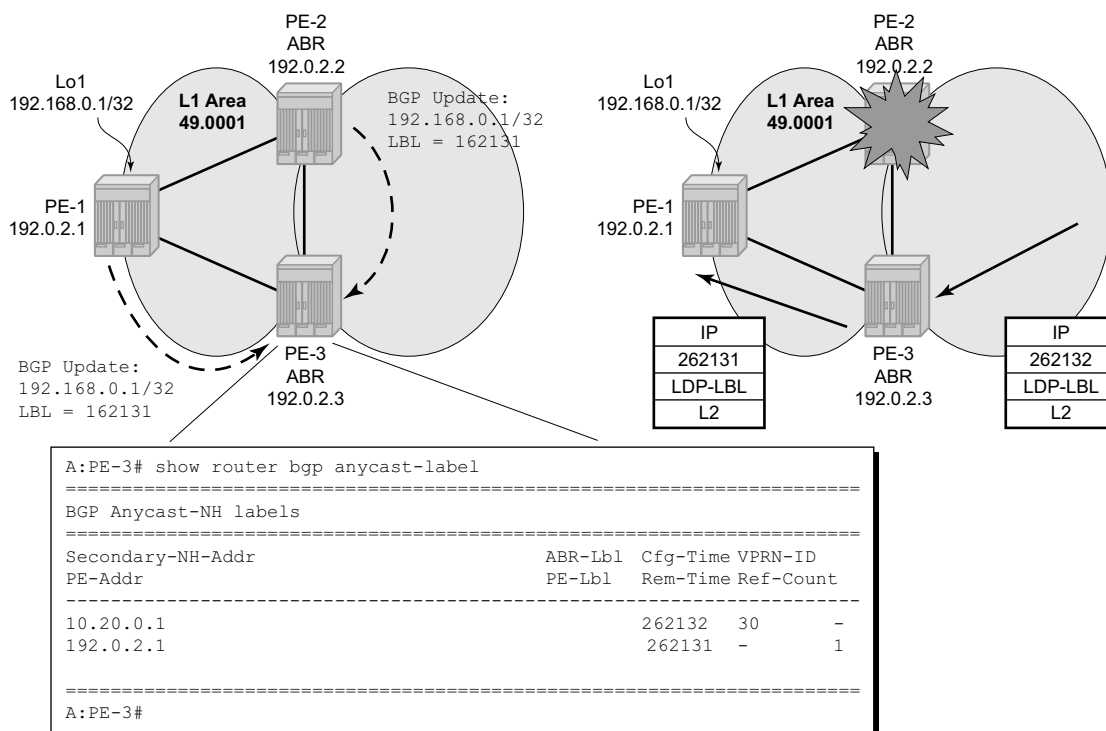
```

A:PE-1>config>router>bgp# show router policy prefix-list "AN"
prefix 192.168.0.1/32 exact
A:PE-1>config>router>bgp# show router policy prefix-list "PE"
prefix 192.0.2.1/32 exact
A:PE-1>config>router>bgp#

```

The prefix list should include all the addresses of connected ANs that need to be reachable through labelled BGP.

Note that the ABR will also learn the BGP route advertisement of its neighboring ABR and installs the BGP label accordingly, as shown in [Figure 76](#).



OSSG611

Figure 76: Data Path with Failing ABR

In case PE-2 fails or becomes unreachable, PE-3 can still recognize the BGP label of the packet and perform the correct BGP (and LDP) label swap so that the packet will not be dropped and can continue its way towards the destination.

Note that for every AN a new entry in the BGP anycast-label table will be added. Refer to the following output.

```
BGP anycast for A:PE-3# show router bgp anycast-label
=====
BGP Anycast-MH labels
=====
Secondary-MH-Addr          ABR-Lbl  Cfg-Time  VPRN-ID
PE-Addr                    PE-Lbl   Rem-Time  Ref-Count
-----
10.20.0.1                  262131   30        -
192.0.2.1                  262134   -         1

10.20.0.1                  262132   30        -
192.0.2.1                  262131   -         1
```

Important note: If an E2E MPLS transport tunnel is needed between PE-1 and PE-6, additional loopback interfaces need to be created on both PE since the system address cannot be used with the BGP anycast feature. The reason for this is that on the ABR, the best route towards the PE system-ip is an IGP route, not a (labeled) BGP route. BGP anycast can only map a BGP label against another BGP label, not towards an IGP (hence LDP) route¹. This is important to keep in mind, if BGP anycast is deployed in networks where BGP anycast is required for E2E MPLS tunnels between AN and remote PEs.

Figure 77 illustrates the SDP between PE-1 and PE-6.

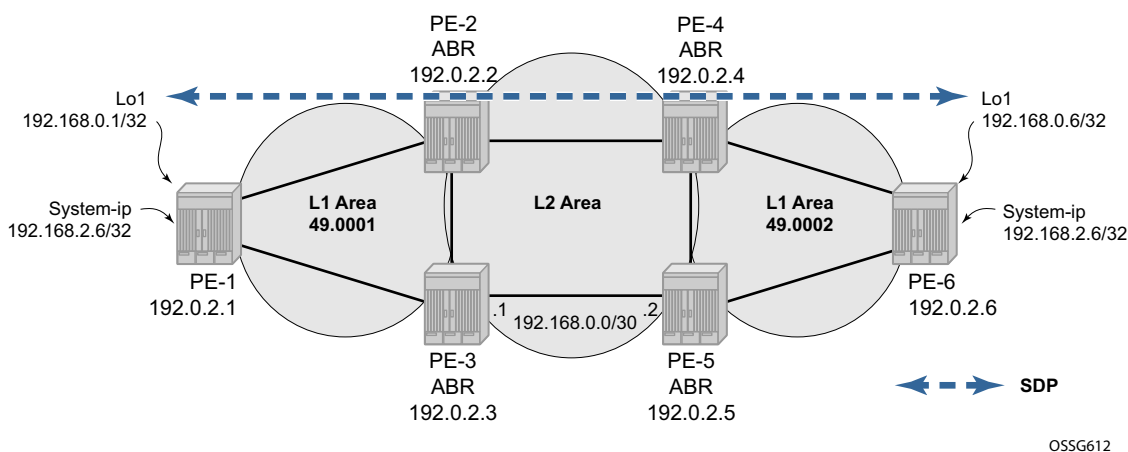


Figure 77: End-to-End Transport Tunnel Using Additional Loopback Interfaces

1. Although PE-1 will advertise the system IP as a BGP-labeled route, it cannot be used for the BGP anycast feature since the route to system IP of PE-1 in the GRT of PE-2 and PE-3 will always be an IGP route, which makes it impossible to use the BGP anycast feature since this requires an E2E BGP tunnel.

SDP Configuration

The SDP configuration consists of two parts, the service component and LDP component. In the service context, the SDP will be created with BGP as the tunneling protocol.

```
*A:PE-1# configure service sdp 100
*A:PE-1>config>service>sdp# info
-----
        far-end 192.168.0.6
        bgp-tunnel
        keep-alive
            shutdown
        exit
        no shutdown
-----
*A:PE-1>config>service>sdp#
```

Since the T-LDP session is set up between two loopback interfaces instead of the system IP of PE-1 and PE-6, a dedicated configuration is required at the LDP level.

```
*A:PE-1# show router interface "lo1"
=====
Interface Table (Router: Base)
=====
Interface-Name      Adm      Opr (v4/v6)  Mode      Port/SapId
IP-Address          PfxState
-----
lo1                  Up        Up/Down      Network   loopback
192.168.0.1/32      n/a
-----
Interfaces : 1
=====
*A:PE-1#
*A:PE-1# configure router ldp
*A:PE-1>config>router>ldp# info
-----
        interface-parameters
            interface "int-PE-1-PE-2"
            exit
            interface "int-PE-1-PE-3"
            exit
        exit
        targeted-session
            peer 192.168.0.6
                local-lsr-id "lo1"
            exit
        exit
-----
*A:PE-1>config>router>ldp#
```

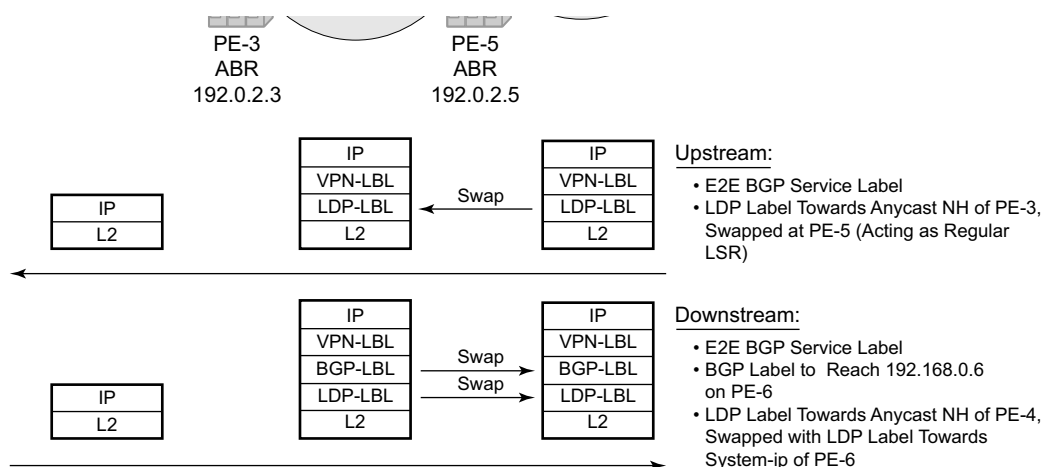
Note that where the destination address of the T-LDP session is set to the remote loopback address and the local address is set to the local loopback interface.

IP-VPN Routes

Important note: In the SR-OS 9.0R1 software release, BGP anycast for IP-VPN routes is only supported with a RR cluster in the core. A full mesh of iBGP peers or confederations are not supported in SROS 9.0R1.

The BGP anycast mechanism can also be used to advertise IP-VPN routes. The advertising PE sets the NH equal to the anycast master address instead of the regular system IP. The configuration is based upon PE-1 which will now act as a customer edge (CE) device. This can be achieved by creating a local IP-VPN on PE-1, without any MPLS connectivity, due to the use of a unique RT (route-target) and hybrid ports towards the ABR². As such, both PE-2 and PE-3 can advertise the same route with the AC address as a NH.

Figure 78 illustrates the logical IP-VPN topology.



OSSG614

Figure 78: IP-VPN with Anycast NH

- The unique RT and hybrid ports are required for this specific set-up where PE-1 is combined as a regular PE and as CE. The CE functionality is achieved by creating a local VPRN, and the unique RT will assure that the route is not imported at PE-2 and PE-3 by MP-BGP. The hybrid port is needed to combine a PE-CE interface and network interface (MPLS capable) on the same physical port.

Configure IP-VPN at the PE (PE-2 and PE-3)

The following output displays the VPRN configuration. The change of the next-hop to the BGP anycast address is done at the BGP group level in this case.

```
A:PE-2>config>service>vprn# info
-----
description "IP-VPN PE, with anycast NH set"
vrf-import "vrfImpPolBgpVpnRts"
vrf-export "adv_vpn"
router-id 192.0.2.2
autonomous-system 64498
route-distinguisher 1:2
auto-bind ldp
vrf-target target:1:1
interface "to_CE" create
  address 172.16.2.2/30
  sap 1/1/1:1 create
  exit
exit
bgp
  group "CE"
    type external
    export "exp_bgp_vpn_rts_to_ce"
    peer-as 64499
    neighbor 172.16.2.1
    exit
  exit
exit
no shutdown
-----

A:PE-2>config>service>vprn# show router policy "vrfImpPolBgpVpnRts"
description "Policy From bgpVpn To none"
entry 10
  description "Entry 10 - From Prot. bgpVpn To none"
  from
    protocol bgp-vpn
    community "vprn1"
  exit
  to
  exit
  action accept
  exit
exit

A:PE-2>config>service>vprn# show router policy "adv_vpn"
entry 10
  description "local routes"
  from
    protocol direct
  exit
  to
    protocol bgp-vpn
  exit
  action accept
```

Deployment Options

```
        community add "vprn1"
    exit
exit
entry 20
    description "PE-CE BGP routes"
    from
        protocol bgp
    exit
    to
        protocol bgp-vpn
    exit
    action accept
        community add "vprn1"
    exit
exit

A:PE-2# show router policy community "vprn1"
community "vprn1" members "target:1:1"
A:PE-2#
A:PE-2# show router policy "exp_bgp_vpn_rts_to_ce"
    description "Policy From bgpVpn To none"
    entry 10
        description "Entry 10 - From Prot. bgpVpn To none"
        from
            protocol bgp-vpn
        exit
        to
            exit
        action accept
        exit
    exit
A:PE-2#
```

Note that the route distinguisher of VPRN 1 at PE-3 is set to 1:3 instead of 1:2 as on PE-2.

As stated above, the NH will be set to the anycast address in the BGP group at the global level.

```
*A:PE-2# configure router bgp
*A:PE-2>config>router>bgp# info
-----
    vpn-apply-export
    min-route-advertisement 1
    group "region"
        description "all PE of the region will peer with this ABR"
        family ipv4 vpn-ipv4
        type internal
        cluster 1.1.1.1
        neighbor 192.0.2.1
            advertise-label ipv4
        exit
    exit
    group "fullmesh"
        description "PE-4 is RR"
        family ipv4 vpn-ipv4
        type internal
        export "exp-anycast-nhs"
        neighbor 192.0.2.4
            advertise-label ipv4
```

```

        exit
    exit
-----

*A:PE-2>config>router>bgp# show router policy "exp-anycast-nhs"
  entry 10
    description "set NH of PE to master anycast"
    from
      prefix-list "PE"
      family ipv4
    exit
    to
      protocol bgp
    exit
    action accept
      local-preference 150 (*)
      next-hop 10.20.0.1
    exit
  exit
entry 20
  description "set NH of IP-VPN routes to master anycast"
  from
    family vpn-ipv4
  exit
  to
    protocol bgp-vpn
  exit
  action accept
    local-preference 150 (*)
    next-hop 10.20.0.1
  exit
exit
*A:PE-2>config>router>bgp#
(*) optional

```

Configure IP-VPN at the Remote PE (PE-6)

On PE-6, IP-VPN routes need to be advertised with the address of lo1 as NH, instead of the regular system-ip. This is required to use a BGP tunnel (and on top of that BGP anycast) to reach PE-6 from PE-2 and PE-3.

```
A:PE-6>config>service>vpn# info
-----
vrf-import "vrfImpPolBgpVpnRts"
router-id 172.16.6.6
route-distinguisher 1:6
auto-bind mpls
vrf-target target:1:1
interface "lo1" create
    address 172.16.6.6/32
    loopback
exit
no shutdown
-----
A:PE-6>config>service>vpn#
```

Note that the NH of the IP-VPN routes will be set to 192.168.0.6 instead of the system-ip;

```
A:PE-6# configure router bgp
A:PE-6>config>router>bgp# info
-----
vpn-apply-import
vpn-apply-export
min-route-advertisement 1
outbound-route-filtering
exit
group "abr"
    description "peering to ABR, which act as RR"
    family ipv4 vpn-ipv4
    type internal
    export "expbgp"
    neighbor 192.0.2.4
        advertise-label ipv4
    exit
    neighbor 192.0.2.5
        advertise-label ipv4
    exit
exit
-----
A:PE-6>config>router>bgp# show router policy
policy          policy-edits
A:PE-6>config>router>bgp# show router policy "expbgp"
entry 10
    description "advertise the PE system-ip through BGP"
    from
        prefix-list "PE"
    exit
    to
        protocol bgp
    exit
    action accept
```

```

        local-preference 200
    exit
exit
entry 20
    description "advertise connect AN through BGP"
    from
        prefix-list "AN"
    exit
    to
        protocol bgp
    exit
    action accept
    exit
exit
entry 30
    description "set NH for vpn-routes to 192.168.0.6"
    from
        community "vprn1"
    exit
    action accept
        local-preference 333
        next-hop 192.168.0.6
    exit
exit
A:PE-6>config>router>bgp# exit all
A:PE-6# show router policy prefix-list "AN"
prefix 192.168.0.6/32 exact
A:PE-6# show router policy prefix-list "PE"
prefix 192.0.2.6/32 exact
A:PE-6#

```

The local preference is set to 333 for troubleshooting purposes, primarily to find the routes again more easily.

The NH 192.168.0.6 is a local loopback, and reachable from PE-2 and PE-3 by a BGP tunnel with the anycast address of PE-4 as NH.

On PE-6:

```

A:PE-6# show router interface
=====
Interface Table (Router: Base)
=====
Interface-Name      Adm      Opr (v4/v6)  Mode      Port/SapId
IP-Address          PfxState
-----
int-PE-6-PE-4      Up       Up/Down      Network  1/1/1:0
192.168.6.2/30      n/a
int-PE-6-PE-5      Up       Up/Down      Network  1/1/3:0
192.168.25.2/30     n/a
lo1                 Up       Up/Down      Network  loopback
192.168.0.6/32      n/a
system             Up       Up/Down      Network  system
192.0.2.6/32        n/a
-----
Interfaces : 4
=====

```

Deployment Options

A:PE-6#

On PE-2 (and PE-3):

A:PE-2# show router tunnel-table

Tunnel Table (Router: Base)

Destination	Owner	Encap	TunnelId	Pref	Nexthop	Metric
10.40.0.1/32	ldp	MPLS	-	9	192.168.4.2	20
10.50.0.1/32	ldp	MPLS	-	9	192.168.13.2	30
192.0.2.1/32	ldp	MPLS	-	9	192.168.2.1	10
192.0.2.3/32	ldp	MPLS	-	9	192.168.2.1	20
192.0.2.4/32	ldp	MPLS	-	9	192.168.4.2	10
192.0.2.5/32	ldp	MPLS	-	9	192.168.13.2	20
192.0.2.6/32	bgp	MPLS	-	10	10.40.0.1	1000
192.168.0.6/32	bgp	MPLS	-	10	10.40.0.1	1000

A:PE-2#

On PE-4 (and PE-5):

The BGP anycast labels are active.

A:PE-4# show router bgp anycast-label

BGP Anycast-MH labels

Secondary-MH-Addr	ABR-Lbl	Cfg-Time	VPRN-ID
PE-Addr	PE-Lbl	Rem-Time	Ref-Count
10.50.0.1	262132	30	-
192.0.2.6	262132	-	1

A:PE-4#

Configure IP-VPN at the CE

On PE-1, VPRN 1 is a local IP-VPN acting as CE. The IP-VPN is dual homed towards PE-2 and PE-3, including EBGP sessions that will advertise the directly connected links toward the PE.

```
A:PE-1>config>service>vprn# info
-----
description "local VPN, simulating CE"
router-id 172.168.1.1
autonomous-system 64499
route-distinguisher 1:1
interface "to_ABR_PE3" create
  address 172.16.3.1/30
  sap 1/1/3:1 create
  exit
exit
interface "to_ABR_PE2" create
  address 172.16.2.1/30
  sap 1/1/1:1 create
  exit
exit
interface "lo1" create
  address 172.168.1.1/32
  loopback
exit
bgp
  min-route-advertisement 1
  export "adv_direct"
  group "ABR"
    type external
    peer-as 64498
    neighbor 172.16.2.2
    exit
    neighbor 172.16.3.2
    exit
  exit
exit
no shutdown
-----

A:PE-1>config>service>vprn#

A:PE-1>config>service>vprn# show router policy "adv_direct"
entry 10
  description "advertise local interfaces to PE"
  from
    protocol direct
  exit
  to
    protocol bgp
  exit
  action accept
  exit
exit
A:PE-1>config>service>vprn#
```

To verify the BGP peering sessions;

A:PE-1# show router 1 bgp summary

```
=====
BGP Router ID:192.0.2.1          AS:64499          Local AS:64499
=====
BGP Admin State      : Up          BGP Oper State      : Up
Total Peer Groups    : 1          Total Peers          : 2
Total BGP Paths       : 3          Total Path Memory    : 384
Total IPv4 Remote Rts : 4          Total IPv4 Rem. Active Rts : 0
Total IPv6 Remote Rts : 0          Total IPv6 Rem. Active Rts : 0
Total Suppressed Rts  : 0          Total Hist. Rts      : 0
Total Decay Rts       : 0

=====
BGP Summary
=====
Neighbor
      AS PktRcvd InQ Up/Down  State|Rcv/Act/Sent (Addr Family)
      PktSent OutQ
-----
172.16.2.2
      64498  28695   0 05d23h53m 2/0/3 (IPv4)
      28671   0
172.16.3.2
      64498  28618   0 06d00h03m 2/0/3 (IPv4)
      28611   0
-----
A:PE-1#
```

The received routes are not inserted in the FIB since they are already known as local routes.

Verify Anycast Labels

On PE-2, first check the BGP service label that has been advertised by PE-3 as follows.

```
A:PE-2# show router bgp routes vpn-ipv4 1:3:172.168.1.1/32
=====
BGP Router ID:192.0.2.2      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best

=====
BGP VPN-IPv4 Routes
=====
Network      : 172.168.1.1/32
Nexthop      : 10.30.0.1
Route Dist.   : 1:3                      VPN Label      : 262130
From         : 192.0.2.4
Res. Nexthop  : n/a
Local Pref.   : 150                      Interface Name : slaveAnycast
Aggregator AS : None                      Aggregator     : None
Atomic Aggr.  : Not Atomic                MED            : None
Community     : target:1:1
Cluster       : 1.1.1.1
Originator Id : 192.0.2.3                  Peer Router Id  : 192.0.2.4
Flags         : Invalid Incomplete (*)
AS-Path       : 64499
VPRN Imported : None

-----
Routes : 1
```

(*) This means that the NH is resolved to its own local interface. Nevertheless, the service label will be inserted in the context-specific label space so that incoming packets with this service label can be linked to VPRN 1.

Notice the loopback route of the CE with label 262130, which is the BGP label that represents VPRN 1 at PE-3.

```
A:PE-2# show router bgp anycast-label
=====
BGP Anycast-MH labels
=====
Secondary-MH-Addr      ABR-Lbl    Cfg-Time  VPRN-ID
PE-Addr               PE-Lbl     Rem-Time  Ref-Count
-----
10.30.0.1              262130     30        1
-                      -          -         3
=====
A:PE-2#
```

PE-3 will forward the traffic received with BGP service label 262130 towards VPRN 1 where an egress IP lookup will be performed. In the VRF (FIB of VPRN-id 1) at PE-2, a route toward 172.168.1.1/32 with VPRN 1 at PE-1 as an NH is found.

Deployment Options

```
A:PE-2# show router 1 route-table
=====
Route Table (Service: 1)
=====
Dest Prefix          Type    Proto    Age          Pref
  Next Hop[Interface Name]          Metric
-----
172.16.2.0/30        Local   Local    06d00h35m    0
    to_CE              0
172.16.3.0/30        Remote  BGP      05h01m41s    170
    172.16.2.1          0
172.168.1.1/32       Remote  BGP      06d00h35m    170
    172.16.2.1          0
-----
No. of Routes: 3
=====
A:PE-2#
```

Verify Data Path Between VPRN on PE-6 and the CE

At PE-6, a ping and traceroute are performed towards the loopback interface of the CE (VPRN-id 1 at PE-1) with the following results.

```
A:PE-6# ping router 1 172.168.1.1 source 172.16.6.6
PING 172.168.1.1 56 data bytes
64 bytes from 172.168.1.1: icmp_seq=1 ttl=63 time=3.79ms.
^C
ping aborted by user

---- 172.168.1.1 PING Statistics ----
1 packet transmitted, 1 packet received, 0.00% packet loss
round-trip min = 3.79ms, avg = 3.79ms, max = 3.79ms, stddev = 0.000ms

A:PE-6# traceroute no-dns router 1 172.168.1.1 source 172.16.6.6
traceroute to 172.168.1.1 from 172.16.6.6, 30 hops max, 40 byte packets
 1  0.0.0.0  * * *
 2  172.168.1.1    4.62 ms  8.05 ms  4.28 ms
A:PE-6#
```

Upstream

From PE-6, the NH of the IP-VPN route will be the anycast address (in this case 10.30.0.1) which will be reachable through an LDP tunnel.

```
A:PE-6# show router 1 route-table
=====
Route Table (Service: 1)
=====
Dest Prefix                                Type    Proto   Age           Pref
  Next Hop[Interface Name]                Metric
-----
172.16.2.0/30                             Remote  BGP VPN  16h21m20s    170
  10.20.0.1 (tunneled)                     0
172.16.3.0/30                             Remote  BGP VPN  16h21m20s    170
  10.30.0.1 (tunneled)                     0
172.16.6.6/32                             Local   Local    10d21h52m     0
  lo1                                       0
172.168.1.1/32                            Remote  BGP VPN  16h21m20s    170
  10.30.0.1 (tunneled)                     0
-----
No. of Routes: 4
=====
A:PE-6# show router 1 fib 1
=====
FIB Display
=====
Prefix                                Protocol
  NextHop
-----
172.16.2.0/30                         BGP_VPN
  10.20.0.1 (VPRN Label:262133 Transport:LDP)
172.16.3.0/30                         BGP_VPN
  10.30.0.1 (VPRN Label:262130 Transport:LDP)
172.16.6.6/32                         LOCAL
```

Deployment Options

```
172.16.6.6 (lo1)
172.168.1.1/32
10.30.0.1 (VPRN Label:262130 Transport:LDP)
-----
Total Entries : 4
=====
A:PE-6#

A:PE-6# show router ldp bindings prefix 10.30.0.1/32
=====
LDP LSR ID: 192.0.2.6
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
      WP - Label Withdraw Pending
=====
LDP Prefix Bindings
=====
Prefix                Peer                IngLbl    EgrLbl  EgrIntf    EgrNextHop
-----
10.30.0.1/32          192.0.2.4          262135U   262139   --         --
10.30.0.1/32          192.0.2.5          262135N   262134   1/1/3:0    192.168.25.1
-----
No. of Prefix Bindings: 2
=====
A:PE-6#
```

Downstream

First, verify which PE is selected by the CE to reach the 172.16.6.6/32 (in VPN 1 at PE-6);

```
A:PE-1# show router 1 route-table
=====
Route Table (Service: 1)
=====
Dest Prefix                Type    Proto    Age        Pref
Next Hop[Interface Name]    Metric
-----
172.16.2.0/30              Local   Local    10d16h54m  0
to_ABR_PE2                  0
172.16.3.0/30              Local   Local    10d16h53m  0
to_ABR_PE3                  0
172.16.6.6/32              Remote  BGP      00h54m07s  170
172.16.2.2                  0
172.168.1.1/32             Local   Local    10d16h52m  0
lo1                         0
-----
No. of Routes: 4
=====
A:PE-1#
```

In this case, PE-2 is selected as the NH.

At PE-2, the path towards PE-6 is based upon a BGP tunnel with the BGP anycast feature.

```

A:PE-2# show router 1 route-table
=====
Route Table (Service: 1)
=====
Dest Prefix                                Type    Proto    Age          Pref
      Next Hop[Interface Name]                                Metric
-----
172.16.2.0/30                             Local   Local    06d16h45m    0
      to_CE                                     0
172.16.3.0/30                             Remote  BGP      21h12m08s   170
      172.16.2.1                                   0
172.16.6.6/32                             Remote  BGP VPN   00h08m18s   170
      192.168.0.6 (tunneled)                       0
172.168.1.1/32                             Remote  BGP      06d16h45m   170
      172.16.2.1                                   0
-----
No. of Routes: 4
=====
A:PE-2#

A:PE-2# show router 1 fib 1
=====
FIB Display
=====
Prefix                                Protocol
      NextHop
-----
172.16.2.0/30                         LOCAL
      172.16.2.0 (to_CE)
172.16.3.0/30                         BGP
      172.16.2.1 Indirect (to_CE)
172.16.6.6/32                         BGP_VPN
      192.168.0.6 (VPRN Label:262134 Transport:BGP)
172.168.1.1/32                         BGP
      172.16.2.1 Indirect (to_CE)
-----
Total Entries : 4
=====
A:PE-2#

A:PE-2# show router tunnel-table
=====
Tunnel Table (Router: Base)
=====
Destination      Owner  Encap  TunnelId  Pref    Nexthop      Metric
-----
10.40.0.1/32     ldp    MPLS   -         9       192.168.4.2   20
10.50.0.1/32     ldp    MPLS   -         9       192.168.13.2  30
192.0.2.1/32     ldp    MPLS   -         9       192.168.2.1   10
192.0.2.3/32     ldp    MPLS   -         9       192.168.2.1   20
192.0.2.4/32     ldp    MPLS   -         9       192.168.4.2   10
192.0.2.5/32     ldp    MPLS   -         9       192.168.13.2  20
192.0.2.6/32     bgp    MPLS   -         10      10.40.0.1     1000
192.168.0.6/32   bgp    MPLS   -         10      10.40.0.1     1000
=====
A:PE-2#

```

The end-to-end label stack for the ping will look like the following (in this particular case);

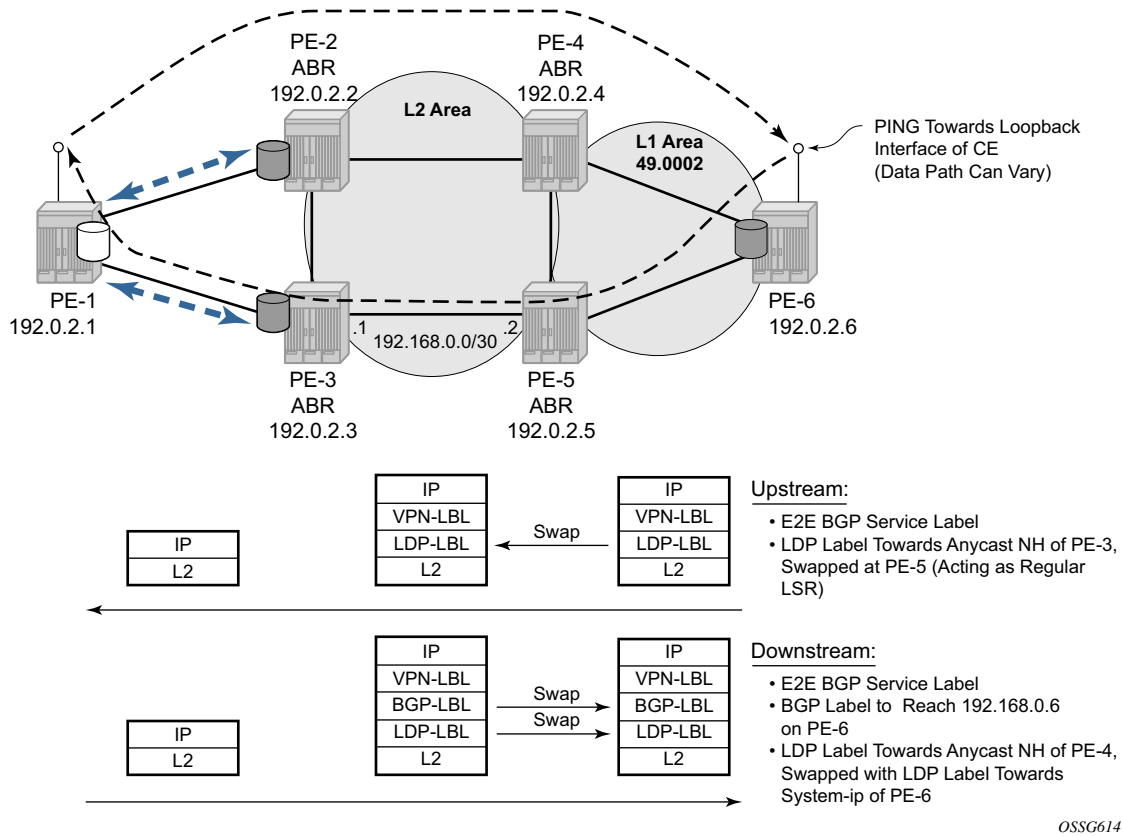


Figure 79: IP-VPN with Anycast NH, Data Path

Notice that the data path is not symmetric. By modifying BGP attributes, the path can be influenced so that the upstream and downstream directions follow the same path, but this is out-of-scope for this document.

Conclusion

BGP anycast provides an end-to-end MPLS reach ability in large MPLS domains, where a single IGP area cannot be deployed or even when a single IGP is not sufficient.

The feature provides a transport layer which can be used for any kind of service on the 7750 SR. On top of that, BGP anycast can also be used to advertise redundant IP-VPN routes into large MPLS domains. In this case, BGP anycast is not offering a redundant transport layer, but redundancy at the service layer.

The (context-specific label-switching) mechanism where a neighboring anycast ABR will install advertised labels from its anycast neighbor in the LFIB contributes to a fast convergence after nodal or link failures.

IGP Shortcuts

In This Chapter

This chapter provides information about IGP shortcuts.

Topics in this section include:

- [Applicability on page 432](#)
- [Overview on page 433](#)
- [Configuration on page 436](#)
- [Conclusion on page 491](#)

Applicability

This chapter is applicable to the 7950 XRS, the 7750 SR series and the 7450 platforms when the feature is not related to BGP and was tested on release 13.0.R2. There are no other pre-requisites for this configuration.

Overview

Interior Gateway Protocols (IGPs) are routing protocols that operate inside an AS (Autonomous System). An AS is a network domain that is managed under a single administration. Because the scope of operation of an IGP is usually within an AS, IGPs are also called intra-AS protocols. The purpose of an IGP is to provide reachability information to destination nodes that are inside the domain. IGPs can be one or more of a variety of protocols, including routing protocols such as RIP version 1 or 2, OSPF, and IS-IS.

IGPs such as OSPF and IS-IS are link-state protocols that use an Shortest Path First (SPF) algorithm to compute the shortest path tree to all nodes in a network. The results of such computations can be represented by the destination node, next-hop address, and output interface, where the output interface is a physical interface. Optionally, MPLS (Multiprotocol Label Switching) LSPs (Label Switched Paths) can be included in the SPF algorithm on the node performing the calculations, as LSPs behave as logical interfaces directly connected to remote nodes in the network. Because the SPF algorithm treats the LSPs in the same way as a physical interface (being a potential output interface), the computation results could be to select a destination node together with an output LSP, using the LSP as a shortcut through the network to the destination node.

Figure 80 shows a normal SPF tree sourced by PE-1 (Provider Edge-1).

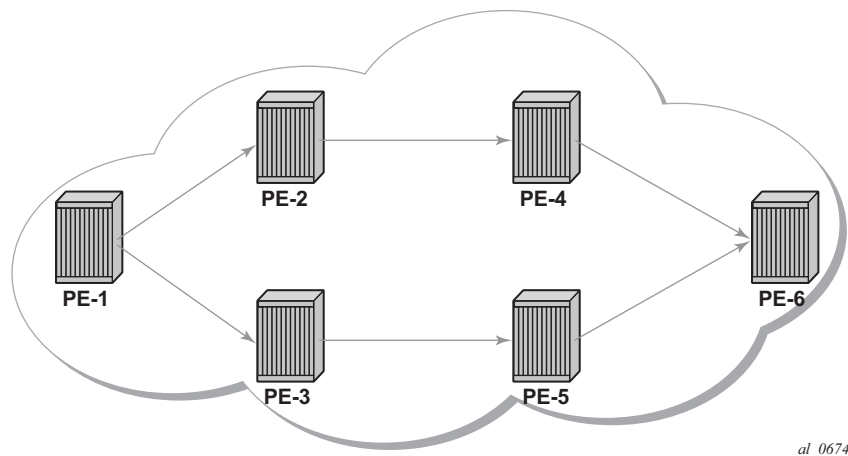


Figure 80: Normal SPF Tree Sourced by PE-1

If there is an LSP that connects PE-1 to PE-5, and IGP shortcuts are configured on PE-1, the SPF tree will be as shown in [Figure 81](#).

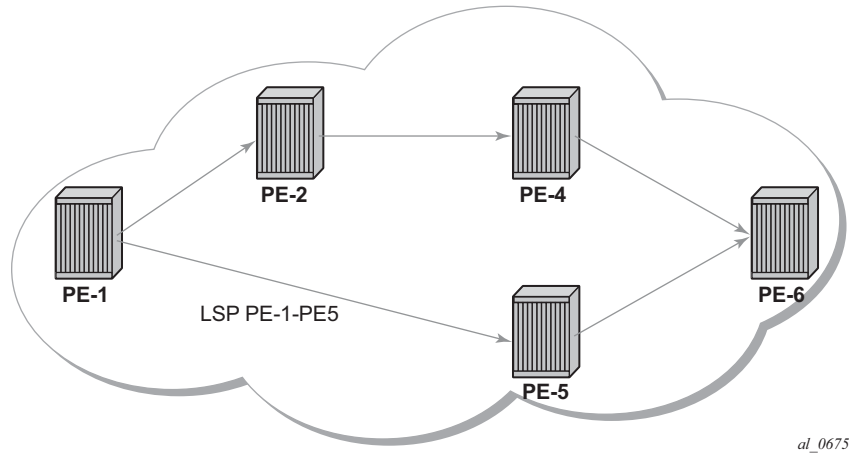


Figure 81: SPF Tree Sourced by PE-1 Using LSP Shortcuts

IGP shortcuts are enabled on a per router basis; SPF computations are independent and irrelevant to other routers, so there is no need to enable shortcuts on every single router.

The network topology used in this example is displayed in [Figure 82](#). The setup consists of six 7750 service routers. There is a single AS and a single IGP area. The following configuration tasks should be completed first:

- IS-IS or OSPF on all interfaces within the AS (configuration has been done using IS-IS but using OSPF shows exactly the same behavior).
- Label Distribution Protocol (LDP) and Resource Reservation Protocol (RSVP) on all interfaces within the AS.

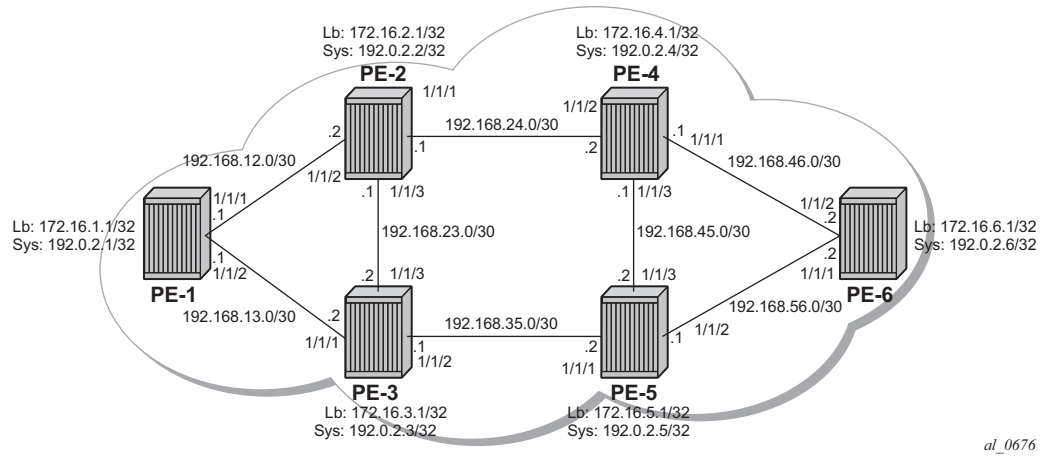


Figure 82: Tested Network Topology

Note: In all figures, **Lb** stands for Loopback and **Sys** stands for the system IP addresses.

Configuration

The first step is to configure the IGP (IS-IS) on all nodes, where IS-IS redistributes route reachability to all routers. To facilitate IS-IS configuration, all routers are L2-L1 capable within the same IS-IS area-id so there is only a single topology area in the network (all routers share the same topology). Traffic engineering is enabled on the IGP as it is a requirement for RSVP. The metric is using the default values: since no reference-bandwidth command is used, the default metric of 10 is applicable on all interfaces. The configuration for PE-2 is displayed below.

```
*A:PE-2# configure router
    interface "int-PE-2-PE-1"
        address 192.168.12.2/30
        port 1/1/2
    exit
    interface "int-PE-2-PE-3"
        address 192.168.23.1/30
        port 1/1/3
    exit
    interface "int-PE-2-PE-4"
        address 192.168.24.1/30
        port 1/1/1
    exit
    interface "system"
        address 192.0.2.2/32
    exit

*A:PE-2# configure router
    isis
        area-id 49.0001
        traffic-engineering
        interface "system"
            passive
        exit
        interface "int-PE-2-PE-1"
            interface-type point-to-point
        exit
        interface "int-PE-2-PE-4"
            interface-type point-to-point
        exit
        interface "int-PE-2-PE-3"
            interface-type point-to-point
        exit
```

The configuration for the other nodes is very similar. The IP addresses can be derived from [Figure 82](#).

The GRT (Global Route Table) for PE-2 is displayed below.

```
*A:PE-2# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type  Proto  Age      Pref
  Next Hop[Interface Name]                        Metric
-----
192.0.2.1/32                                     Remote ISIS   00h00m46s  15
      192.168.12.1                               10
192.0.2.2/32                                     Local  Local   00h02m00s   0
      system                                       0
192.0.2.3/32                                     Remote ISIS   00h00m38s  15
      192.168.23.2                               10
192.0.2.4/32                                     Remote ISIS   00h00m23s  15
      192.168.24.2                               10
192.0.2.5/32                                     Remote ISIS   00h00m18s  15
      192.168.23.2                               20
192.0.2.6/32                                     Remote ISIS   00h00m08s  15
      192.168.24.2                               20
192.168.12.0/30                                  Local  Local   00h02m00s   0
      int-PE-2-PE-1                             0
192.168.13.0/30                                  Remote ISIS   00h00m46s  15
      192.168.12.1                               20
192.168.23.0/30                                  Local  Local   00h02m00s   0
      int-PE-2-PE-3                             0
192.168.24.0/30                                  Local  Local   00h02m00s   0
      int-PE-2-PE-4                             0
192.168.35.0/30                                  Remote ISIS   00h00m38s  15
      192.168.23.2                               20
192.168.45.0/30                                  Remote ISIS   00h00m23s  15
      192.168.24.2                               20
192.168.46.0/30                                  Remote ISIS   00h00m23s  15
      192.168.24.2                               20
192.168.56.0/30                                  Remote ISIS   00h00m17s  15
      192.168.23.2                               30
-----
No. of Routes: 14
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====
*A:PE-2#
```

LDP and RSVP Shortcuts

Interface Label Distribution Protocol (iLDP) is enabled on all interfaces (except system interfaces, which is not allowed) in all routers. The configuration on all nodes is similar and the IP addresses are derived from [Figure 82](#). Below is the configuration of PE-4.

```
*A:PE-4# configure router ldp
      interface-parameters
        interface "int-PE-4-PE-2"
        exit
        interface "int-PE-4-PE-5"
        exit
        interface "int-PE-4-PE-6"
        exit
      exit
      targeted-session
      exit
```

With iLDP enabled, PE-4 establishes iLDP sessions with its directly connected neighbors, as shown below.

```
*A:PE-4# show router ldp session
=====
LDP IPv4 Sessions
=====
Peer LDP Id          Adj Type  State          Msg Sent  Msg Recv  Up Time
-----
192.0.2.2:0          Link      Established    1205      1204      0d 00:53:02
192.0.2.5:0          Link      Established    1198      1197      0d 00:52:55
192.0.2.6:0          Link      Established    181       183       0d 00:07:43
-----
No. of IPv4 Sessions: 3
=====
LDP IPv6 Sessions
=====
Peer LDP Id
Adj Type          State          Msg Sent  Msg Recv  Up Time
-----
No Matching Entries Found
=====
*A:PE-4#
```

The following tunnel table shows that there is a Label Switched Path (LSP) to every other router. The reason is that the LDP label distribution mode is DU (Downstream Unsolicited) by default, originating label bindings for system addresses only (which are used by iLDP as transport address by default). The command also shows LSPs' preference (where the preference is 9 for LDP) and metric (metric is inherited from the IGP, each hop counts as a metric of 10), as shown below. The metric to destinations PE-1 and PE-3 is 20 because there are two hops in between (PE-4 is two hops away from PE-1 and PE-3).

```
*A:PE-4# show router tunnel-table
=====
IPv4 Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId  Pref    Nexthop      Metric
-----
192.0.2.1/32     ldp        MPLS   65538     9        192.168.24.1  20
192.0.2.2/32     ldp        MPLS   65537     9        192.168.24.1  10
192.0.2.3/32     ldp        MPLS   65539     9        192.168.24.1  20
192.0.2.5/32     ldp        MPLS   65540     9        192.168.45.2  10
192.0.2.6/32     ldp        MPLS   65545     9        192.168.46.2  10
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
*A:PE-4#
```

In order to configure RSVP shortcuts, RSVP must be enabled on all interfaces where traffic engineering is required. For simplicity, RSVP is configured on all interfaces of the network, including system interfaces. The configuration for PE-6 is displayed below. When RSVP is in **no shutdown**, it is automatically configured on the interfaces where MPLS is configured.

```
*A:PE-6# configure router mpls no shutdown
*A:PE-6# configure router rsvp no shutdown

*A:PE-6# configure router
      mpls
        interface "system"
        exit
        interface "int-PE-6-PE-4"
        exit
        interface "int-PE-6-PE-5"
        exit
```

The configuration of the other nodes is similar. The IP addresses can be derived from [Figure 82](#). Because there are no RSVP LSPs configured yet, the tunnel-table has no RSVP LSPs and only contains LDP LSPs.

LDP Static Route (IP Tunneled in LDP Tunnel)

Using LDP LSP shortcuts for static route resolution enables forwarding of IPv4 packets over LDP LSPs instead of using a regular IP next-hop. In other words, the traffic to the resolved static routes is forwarded making use of an MPLS LDP LSP rather than plain IP.

The configuration defines a static route pointing to the destination PE (remote loopback, which is an indirect next hop in the example), and explicitly indicates that it should use LDP rather than IGP. Taking PE-1 and PE-6 as an example, two loopback interfaces are configured (172.16.X.1/32), where X = PE number, and a static-route is defined according to the explanation above. The following shows the configuration on PE-1.

```
*A:PE-1# configure router
      interface "loopback"
        address 172.16.1.1/32
        loopback
      exit
      static-route 172.16.6.1/32 indirect 192.0.2.6
      static-route-entry 172.16.6.1/32
        indirect 192.0.2.6
        tunnel-next-hop
          resolution-filter
            ldp
          exit
          disallow-igp
          resolution filter
        exit
      exit
    exit
```

Looking at the GRT or FIB (Forwarding Database), there are two new entries corresponding to the two configured loopbacks. One entry is associated with protocol local (local loopback on the PE), and the other entry is protocol static, where the next hop is reached using a LDP LSP.

```
*A:PE-1# show router fib 1
=====
FIB Display
=====
Prefix [Flags]                                Protocol
NextHop
-----
172.16.1.1/32                                LOCAL
    172.16.1.1 (loopback)
172.16.6.1/32                                STATIC
    192.0.2.6 (Transport:LDP)
---snipped---
```

The next output shows that a ping sourced by PE-1's loopback interface is able to reach PE-6's loopback, and traceroute demonstrates that the traffic is following the LDP LSP. The ping and traceroute commands cannot follow the IGP because the static-route command states that the IGP

is disallowed when no LDP LSP towards PE-6 is available (also, the loopback interfaces are not enabled on IS-IS).

```
*A:PE-1# ping 172.16.6.1 source 172.16.1.1
PING 172.16.6.1 56 data bytes
64 bytes from 172.16.6.1: icmp_seq=1 ttl=64 time=2.03ms.
64 bytes from 172.16.6.1: icmp_seq=2 ttl=64 time=2.16ms.
64 bytes from 172.16.6.1: icmp_seq=3 ttl=64 time=2.01ms.
64 bytes from 172.16.6.1: icmp_seq=4 ttl=64 time=2.78ms.
64 bytes from 172.16.6.1: icmp_seq=5 ttl=64 time=3.18ms.

---- 172.16.6.1 PING Statistics ----
5 packets transmitted, 5 packets received, 0.00% packet loss
round-trip min = 2.01ms, avg = 2.43ms, max = 3.18ms, stddev = 0.466ms

*A:PE-1# traceroute 172.16.6.1 source 172.16.1.1
traceroute to 172.16.6.1 from 172.16.1.1, 30 hops max, 40 byte packets
 1  0.0.0.0  * * *
 2  0.0.0.0  * * *
 3  172.16.6.1 (172.16.6.1)    1.69 ms  3.24 ms  2.46 ms
*A:PE-1#
```

With the traceroute command, there are three hops from PE-1 to PE-6. There is no information regarding IP for the first two hops because the traffic is encapsulated in an MPLS LDP. The reason why the hops are displayed even when there is an MPLS LSP tunnel is because by default, the SR router propagates (copies) the TTL (Time to Live) from the IP header in the MPLS header. This is known as uniform mode.

However, a service provider might not want to show how many MPLS hops (nodes) there are in their network if a **traceroute** command is executed from outside their network. To prevent internal hops being shown, no **propagate** commands are needed in the LDP configuration, as shown below. This is known as pipe mode.

```
*A:PE-1# configure router ldp
      no shortcut-local-ttl-propagate
      no shortcut-transit-ttl-propagate
exit
```

Once TTL propagation is disabled, the hops are not displayed any longer when running the **traceroute** command.

```
*A:PE-1# traceroute 172.16.6.1 source 172.16.1.1
traceroute to 172.16.6.1 from 172.16.1.1, 30 hops max, 40 byte packets
 1  172.16.6.1 (172.16.6.1)    2.36 ms  2.24 ms  2.25 ms
*A:PE-1#
```

RSVP Static Route (IP Tunneled in RSVP Tunnel)

Using RSVP LSP shortcuts for static route resolution enables forwarding of IPv4 packets over RSVP LSPs instead of using a regular IP next-hop. In other words, the traffic to the resolved static routes is forwarded making use of an MPLS RSVP LSP rather than plain IP.

The configuration defines a static route pointing to a destination PE (remote loopback, which is an indirect next hop in the example), and explicitly indicates that it should use RSVP rather than IGP. Taking PE-6 and PE-1 as an example, two loopback interfaces are configured (172.16.X.1/32), where X = PE number, and a static-route is defined according to the explanation above. The following shows the configuration on PE-6.

```
*A:PE-6# configure router
      interface "loopback"
        address 172.16.6.1/32
        loopback
        no shutdown
      exit
      static-route 172.16.1.1/32 indirect 192.0.2.1
      static-route-entry 172.16.1.1/32
        indirect 192.0.2.1
          tunnel-next-hop
            resolution-filter
              rsvp-te
              exit
            exit
            disallow-igp
            resolution filter
          exit
        exit
      exit
    exit
```

Also, an RSVP LSP needs to be configured with PE-1's system interface as the destination:

```
*A:PE-6# configure router mpls
      path "loose"
        no shutdown
      exit
      lsp "LSP-PE-6-PE-1"
        to 192.0.2.1
        primary "loose"
        exit
        no shutdown
      exit
```

Reviewing the LSP tunnel table, observe that there is an RSVP LSP created:

```
*A:PE-6# show router tunnel-table
=====
IPv4 Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId  Pref    Nexthop      Metric
-----
192.0.2.1/32     rsvp       MPLS    1          7        192.168.46.1  30
192.0.2.1/32     ldp        MPLS    65553      9        192.168.46.1  30
192.0.2.2/32     ldp        MPLS    65552      9        192.168.46.1  20
192.0.2.3/32     ldp        MPLS    65540      9        192.168.56.1  20
192.0.2.4/32     ldp        MPLS    65551      9        192.168.46.1  10
192.0.2.5/32     ldp        MPLS    65541      9        192.168.56.1  10
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
*A:PE-6#
```

Note that the default RSVP preference is 7 (preferred over that of LDP, which is 9) and the metric reflects that this LSP spans 3 hops (for a dynamic LSP not using CSPF, the metric is inherited from IGP). See [RSVP Shortcut for IGP Route Resolution on page 449](#) for more details about the metric applied in LSPs.

The RSVP LSP is used to resolve the indirect next hop (PE-1 system address) in the static route (the LSP used is identified with the Tunnel ID, in this case 1), hence the GRT is modified with the following entry:

```
*A:PE-6# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type    Proto    Age      Pref
Next Hop[Interface Name]                          Metric
-----
172.16.1.1/32                                     Remote  Static   00h00m21s  5
    192.0.2.1 (tunneled:RSVP:1)                    1
---snipped---
-----
No. of Routes: 16
```

As in the LDP shortcut with static route example, between PE-6 and PE-1, TTL propagation is disabled.

```
*A:PE-6# configure router mpls
      no shortcut-local-ttl-propagate
      no shortcut-transit-ttl-propagate
```

RSVP Static Route (IP Tunneled in RSVP Tunnel)

The output is the following when running a traceroute:

```
*A:PE-6# traceroute 172.16.1.1 source 172.16.6.1
traceroute to 172.16.1.1 from 172.16.6.1, 30 hops max, 40 byte packets
 1  172.16.1.1 (172.16.1.1)      2.22 ms  2.04 ms  2.18 ms
*A:PE-6#
```

Note that the two static routes that have been defined to use the LDP and RSVP shortcuts follow the static routes default values and have a preference of 5 and a metric of 1.

LDP Shortcut for IGP Route Resolution

Using LDP shortcuts for IGP route resolution enables forwarding of packets to IGP learned routes over an LDP LSP. The default is to disable the LDP shortcut across all interfaces in the node.

When LDP shortcuts are enabled, LDP populates the RTM (Route Table Manager) with next-hop entries corresponding to all prefixes for which it activated an LDP Forwarding Equivalence Class (FEC). For a given prefix, two route entries are populated in RTM. One corresponds to the LDP shortcut next-hop and has an owner of LDP. The other one is the regular IP next-hop. The LDP shortcut next-hop always takes preference over the regular IP next-hop for forwarding user packets and specific control packets over a given outgoing interface to the route next-hop.

Once LDP has activated a FEC for a given prefix and programmed RTM, it also programs the ingress Tunnel Table in the line card with the LDP tunnel information.

When an IPv4 packet is received on an ingress network interface, a subscriber IES (Internet Enhanced Service) interface, or a regular IES interface, the lookup of the packet by the ingress line card results in the packet being sent labeled with the label stack corresponding to the NHLFE (Next Hop Label Forwarding Entry) of the LDP LSP when the preferred RTM entry corresponds to an LDP shortcut. If the preferred RTM entry corresponds to an IP next-hop, the IPv4 packet is forwarded unlabeled. The activation of the FEC by LDP is done by performing an exact match with an IGP route prefix in RTM but it can also be done by performing a longest prefix-match with an IGP route in RTM if the aggregate-prefix-match option is enabled globally in LDP.

Handling of Control Packets

All control plane packets will not see the LDP shortcut route entry in RTM with the exception of the following control packets which will be forwarded over an LDP shortcut when enabled:

- A locally generated or in transit ICMP ping and UDP traceroute of an IGP route. The transit message appears as a user packet to the ingress LER node.
- A locally generated response to a received ICMP ping or UDP traceroute message.

All other control plane packets that require an RTM lookup and have knowledge of which destination is reachable over the LDP shortcut will continue to be forwarded over the IP next-hop route in RTM.

Handling of Multicast Packets

LDP shortcuts apply to unicast FEC types and are used for forwarding IP unicast packets in the data path. IP multicast packets forwarded over an mLDP P2MP LSP (Multicast Label Distribution Protocol Point-to-Multi-point LSP) make use of a multicast FEC and thus cannot make use of the LDP unicast shortcut.

ECMP Considerations

When Equal Cost Multi-Path (ECMP) is enabled and multiple equal-cost next-hops exist for the IGP route, the ingress line card will spray the packets for this route based on the hashing routine supported for IPv4 packets. When the preferred RTM entry corresponds to an LDP shortcut route, spraying is performed across the multiple next-hops for the LDP FEC. The FEC next-hops can either be direct link LDP neighbors or T-LDP (Targeted LDP) neighbors reachable over RSVP LSPs in the case of LDP-over-RSVP but not both. This is as per ECMP for LDP in the existing implementation. When the preferred RTM entry corresponds to a regular IP route, spraying will be performed across regular IP next-hops for the prefix. Spraying across regular IP next-hops and LDP-shortcut next-hops concurrently is not supported.

Configuring IGP LDP shortcuts is straightforward, and only applies to the node where there is interest to provision the LDP shortcut. In this example, only PE-1 is provisioned with LDP shortcuts, as shown below.

```
*A:PE-1#configure router ldp-shortcut
```

Now, all tunnel LSPs that resolve an IGP next hop will replace the IP next hops, as depicted in the following output:

```
*A:PE-1# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type  Proto  Age           Pref
Next Hop[Interface Name]                        Metric
-----
192.0.2.1/32                                     Local  Local  01h31m15s    0
system
192.0.2.2/32                                     Remote LDP    00h04m15s    9
192.168.12.2 (tunneled)
192.0.2.3/32                                     Remote LDP    00h04m15s    9
192.168.13.2 (tunneled)
192.0.2.4/32                                     Remote LDP    00h04m15s    9
192.168.12.2 (tunneled)
192.0.2.5/32                                     Remote LDP    00h04m15s    9
192.168.13.2 (tunneled)
192.0.2.6/32                                     Remote LDP    00h04m15s    9
192.168.12.2 (tunneled)
192.168.12.0/30                                 Local  Local  01h31m15s    0
```

```

        int-PE-1-PE-2
192.168.13.0/30                Local    Local    01h31m15s  0
        int-PE-1-PE-3
192.168.23.0/30                Remote  ISIS     01h29m45s  15
        192.168.12.2
192.168.24.0/30                Remote  ISIS     01h29m45s  15
        192.168.12.2
192.168.35.0/30                Remote  ISIS     01h29m37s  15
        192.168.13.2
192.168.45.0/30                Remote  ISIS     01h29m22s  15
        192.168.12.2
192.168.46.0/30                Remote  ISIS     01h29m22s  15
        192.168.12.2
192.168.56.0/30                Remote  ISIS     01h29m16s  15
        192.168.13.2
-----
No. of Routes: 14
Flags: n = Number of times nexthop is repeated
       B = BGP backup route available
       L = LFA nexthop available
       S = Sticky ECMP requested
=====
*A:PE-1#
*A:PE-1# show router fib 1
=====
FIB Display
=====
Prefix [Flags]                                Protocol
NextHop
-----
192.0.2.1/32                                LOCAL
    192.0.2.1 (system)
192.0.2.2/32                                LDP
    192.0.2.2 (Transport:LDP)
192.0.2.3/32                                LDP
    192.0.2.3 (Transport:LDP)
192.0.2.4/32                                LDP
    192.0.2.4 (Transport:LDP)
192.0.2.5/32                                LDP
    192.0.2.5 (Transport:LDP)
192.0.2.6/32                                LDP
    192.0.2.6 (Transport:LDP)
192.168.12.0/30                            LOCAL
    192.168.12.0 (int-PE-1-PE-2)
192.168.13.0/30                            LOCAL
    192.168.13.0 (int-PE-1-PE-3)
192.168.23.0/30                            ISIS
    192.168.12.2 (int-PE-1-PE-2)
192.168.24.0/30                            ISIS
    192.168.12.2 (int-PE-1-PE-2)
192.168.35.0/30                            ISIS
    192.168.13.2 (int-PE-1-PE-3)
192.168.45.0/30                            ISIS
    192.168.12.2 (int-PE-1-PE-2)
192.168.46.0/30                            ISIS
    192.168.12.2 (int-PE-1-PE-2)
192.168.56.0/30                            ISIS
    192.168.13.2 (int-PE-1-PE-3)
-----

```

LDP Shortcut for IGP Route Resolution

Total Entries : 14

Applying LDP IGP shortcuts only on PE-1 implies that IP traffic from PE-1 to any of the system addresses of the rest of nodes will use the LDP shortcut, however, the traffic replied from any PE back to PE-1 will be native IP since IGP shortcuts have not been provisioned in the other nodes.

RSVP Shortcut for IGP Route Resolution

Using RSVP LSP shortcuts when resolving IGP routes enables forwarding of packets to IGP learned routes over an RSVP LSP. The use of RSVP shortcut for resolving IGP routes is enabled at the IS-IS (or OSPF) routing protocol level or at the LSP level, and instructs IS-IS and OSPF to include RSVP LSPs originating on this node and terminating on the system address (router-id) of a remote node and considers them as direct links. RSVP LSPs with a destination address corresponding to an interface address or any other loopback interface address of a remote node are automatically not considered by IS-IS or OSPF.

By default, **rsvp-shortcut** is disabled in all IGP instances.

RSVP LSPs are included in the IGP SPF computation with the following characteristics:

- RSVP LSP is modeled as a point-to-point link IP interface and its metric is used in the computation of the shortest path of IGP routes
- Next-hop and interface include the NHLFE of the shortcut LSP when the IGP path cost using the RSVP LSP is the best.
- Shortcuts are not used when the destination RSVP LSP is in a different IGP area. In addition, IGP adjacencies across an RSVP LSP are not supported.

The next output shows the configuration commands:

```
configure router isis rsvp-shortcut
```

Note that the configuration can be done at the IGP level or per LSP level. When rsvp-shortcut is enabled at the IGP instance level, all RSVP LSPs originating on this node are eligible by default. The user can, however, exclude a specific RSVP LSP from being used as a shortcut for resolving IGP routes by entering the command

```
*A:PE-6# configure router mpls lsp "LSP-PE-6-PE-1" no igp-shortcut
```

As RSVP shortcuts can coexist with LDP shortcuts or IP next hops, SPF computation and path selection follows the procedures in RFC 3906:

- SPF picks the RSVP shortcut next-hop if there is an RSVP LSP directly to that address regardless of the path cost compared to the IGP next-hop.
- SPF picks the RSVP shortcut next-hop or the IGP next-hop based on path lowest cost if there is an IGP path to the prefix that does not go via the tail-end of the LSP.
- If the IGP next-hop is picked, then it can be an LDP shortcut next-hop or a regular IP next-hop. The LDP shortcut next-hop always has preference over the regular IP next-hop.

Handling of Control Packets

All control plane packets requiring an RTM lookup and whose destination is reachable over the RSVP shortcut are forwarded over the shortcut. This is because RTM keeps a single route entry for each prefix except if there is ECMP over different outgoing interfaces. Interface bound control packets are not impacted by the RSVP shortcut since RSVP LSPs with a destination address different than the router-id are not included by IGP in its SPF calculation.

Important note: RSVP shortcuts for IGP shortcut resolution should only be used with CSPF LSPs and/or with fully explicit path non CSPF LSP. RSVP hop-by-hop Path messages will try to use the shortcut and consequently LSPs without CSPF enabled, or that use a loose/empty hop path, will not come up. However, LSPs with CSPF enabled or using a strict hop path will come up. This is because in the former case the RTM lookup to get the next hop results in using the shortcut and so the path messages are sent directly to the destination of the LSP, where they are dropped. With CSPF enabled, the next-hop (and the entire path) is provided by CSPF and the path messages are sent unlabeled to the directly connected neighbor which corresponds to the next-hop of the destination of the LSP. Similar processing occurs if a strict hop path is used, as is the case in the example below.

Handling of Multicast Packets

IP multicast packets cannot be forwarded over an RSVP shortcut, they can only be forwarded over an RSVP P2MP LSP. However, as RSVP shortcut routes appear in RTM and are seen by all applications when they are the best route. When the reverse path forwarding (RPF) check for the source of the multicast packet matches an RSVP shortcut route, the check will pass if both the RSVP shortcut and the multicast-import options are enabled in the IGP, as shown below, as the RTM is populated with next hops only and not with tunnels (RPFs will fail for source prefixes resolved to a tunnel NH).

```
*A:PE-6# configure router isis multicast-import
- multicast-import [both]
- multicast-import [ipv4]
- multicast-import [ipv6]
- no multicast-import [both]
- no multicast-import [ipv4]
- no multicast-import [ipv6]

<ipv4>           : keyword
<ipv6>           : keyword
<both>          : keyword

*A:PE-6#
```

The unicast RTM can still make use of the tunnel next-hop for the same prefix. SPF keeps track of both the direct first hop and the tunneled first hop of a node which is added to the Dijkstra tree.

ECMP Considerations

When ECMP is enabled and multiple equal-cost paths exist for the route over a set of tunnel next-hops based on the hashing routine supported for IPv4 packets, there are two possibilities:

- Destination is tunnel-endpoint: the system selects the tunnel with lowest tunnel ID (IP next-hop is never used).
- Destination is different from the tunnel endpoint: it selects tunnel endpoints when the LSP metric is not greater than the IGP cost and it prefers tunnel endpoint over IP next-hop.

Note that ECMP is not performed across both the IP and tunnel next-hops.

RSVP Shortcuts Configuration

Configuring RSVP LSP shortcuts is straightforward, and only applies to the node where there is interest to provision the RSVP shortcut. Two LSPs, from PE-6 to PE-1 and from PE-1 to PE-6, with strict hops, are provisioned according to [Figure 83](#).

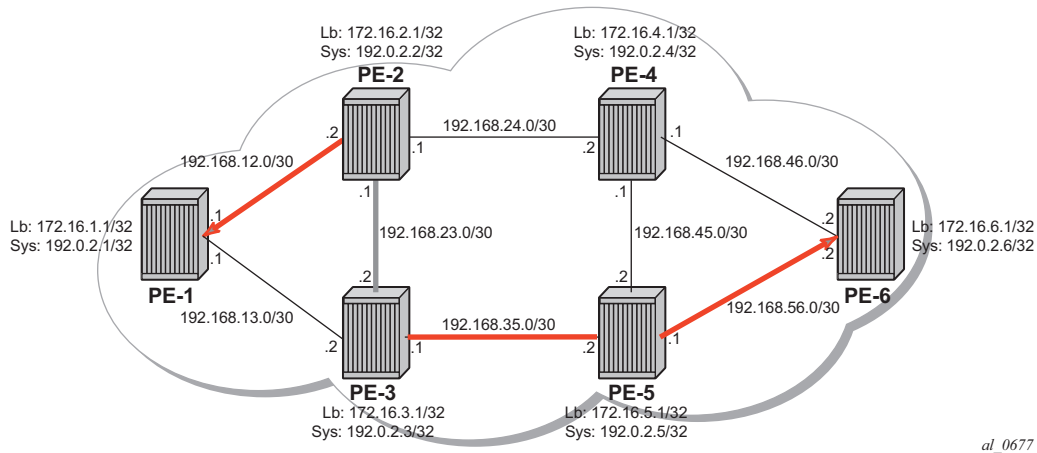


Figure 83: LSPs Between PE-1 and PE-6

The configuration on PE-1 and PE-6 is similar (replacing the IP addresses), so only the configuration for PE-6 is shown:

```
configure router isis rsvp-shortcut

configure router mpls
  path "path-to-PE-1"
    hop 10 192.0.2.5 strict
    hop 20 192.0.2.3 strict
    hop 30 192.0.2.2 strict
    hop 40 192.0.2.1 strict
    no shutdown
  exit
  lsp "LSP-PE-6-PE-1-strict"
    to 192.0.2.1
    primary "path-to-PE-1"
  exit
  no shutdown
exit
```


The GRT output shows the change in the next hop, using an RSVP shortcut:

```
*A:PE-6# show router route-table

=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                Type   Proto   Age           Pref
Next Hop[Interface Name]          Metric
-----
192.0.2.1/32                      Remote ISIS   00h00m34s  15
      192.0.2.1 (tunneled:RSVP:2)  16777215
192.0.2.2/32                      Remote ISIS   00h00m34s  15
      192.168.46.1                  20
192.0.2.3/32                      Remote ISIS   00h00m34s  15
      192.168.56.1                  20
192.0.2.4/32                      Remote ISIS   00h00m34s  15
      192.168.46.1                  10
192.0.2.5/32                      Remote ISIS   00h00m34s  15
      192.168.56.1                  10
192.0.2.6/32                      Local  Local    01h45m01s   0
      system                        0
192.168.12.0/30                   Remote ISIS   00h00m34s  15
      192.168.46.1                  30
192.168.13.0/30                   Remote ISIS   00h00m34s  15
      192.168.56.1                  30
192.168.23.0/30                   Remote ISIS   00h00m34s  15
      192.168.46.1                  30
192.168.24.0/30                   Remote ISIS   00h00m34s  15
      192.168.46.1                  20
192.168.35.0/30                   Remote ISIS   00h00m34s  15
      192.168.56.1                  20
192.168.45.0/30                   Remote ISIS   00h00m34s  15
      192.168.46.1                  20
192.168.46.0/30                   Local  Local    01h45m02s   0
      int-PE-6-PE-4                  0
192.168.56.0/30                   Local  Local    01h45m01s   0
      int-PE-6-PE-5                  0
-----
No. of Routes: 14
Flags: n = Number of times nexthop is repeated
       B = BGP backup route available
       L = LFA nexthop available
       S = Sticky ECMP requested
=====
*A:PE-6#
```

The RSVP LSP in the output has a metric of 16777215, the LSP administrative metric matches the maximum value allowed for an IS-IS link using the wide-metric (24-bit value with a range of [0 — 16777215]). The following metric rules apply:

- A dynamic strict path non-CSPF LSP has the maximum metric (16777215).

- A dynamic CSPF LSP has a metric equal to the cumulative IGP cost.
 - If the user enabled the use of the TE metric on this LSP (configure router mpls lsp cspf use-te-metric), then the metric for the LSP is the maximum (16777215).
 - If the user enabled the use of the TE metric on this LSP (configure router mpls lsp cspf use-te-metric) and provisioned a specific metric on the lsp (configure router mpls lsp metric <metric>:<0..16777215>), then the metric for the LSP is the one provisioned. Note that when configuring the metric of an LSP, the parameter “use-te-metric” is not required.
- A static LSP has a maximum metric (16777215).
- Manual and dynamic bypass LSPs have the maximum metric (16777215).

Note: The RSVP shortcuts section detailed the importance of the LSP metric when using CSPF LSPs or when importing RSVP tunnel links into the IGP. The LSP metric can be inherited from the IGP, or can be manually modified by configuring a specific LSP metric or relative-metric offset. As IP and LDP FECs resolve to RSVP LSPs when the metric is equal or lower compared to the regular routing metric, configuring a specific static LSP metric (lower than the IGP metric) or relative-metric offset is strongly recommended when using RSVP shortcuts, so that the GRT and LDP FEC resolution will always prefer to use RSVP LSP shortcuts when the CSPF path computation is not using the shortest path.

For the example above, first rule applies.

Advertising RSVP LSP Tunnel Links in the IGP: Forwarding Adjacency Feature

If configured, an RSVP LSP can also be advertised into the IGP similar to regular links so that other routers in the network can include it into their SPF computations. The forwarding adjacency feature can be enabled independently from the RSVP shortcut feature in CLI. If both are configured for a given IGP instance, the forwarding adjacency takes precedence. An RSVP LSP must exist in the reverse direction in order for the advertised link to pass the bi-directional link check and be usable by other routers in the network. However, this is not required for the node which originates the LSP. The LSP is advertised as an unnumbered point-to-point link and the link LSP/LSA has no Traffic Engineering opaque sub-TLVs as per RFC 3906.

Reusing the RSVP IGP shortcuts set up previously (PE-1 and PE-6 RSVP IGP shortcut example according to [Figure 83](#)), the outcome is a route linked with an RSVP LSP as next hop, as seen below:

```
*A:PE-6# show router route-table 192.0.2.1/32
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type    Proto    Age          Pref
      Next Hop[Interface Name]                      Metric
-----
192.0.2.1/32                                     Remote  ISIS     00h02m37s    15
      192.0.2.1 (tunneled:RSVP:2)                      16777215
-----
No. of Routes: 1
```

The route tunneled through RSVP has a metric of 16777215, so it is not used by PE-6 GRT to reach any other routes since the metric is very high. After enabling the forwarding adjacency feature (tunnel links) to use shortcuts in the configuration, PE-1 and PE-6 have a direct connection through the RSVP LSP (as a virtual link). This configuration command must be executed in both routers.

```
configure router isis advertise-tunnel-link
```

Once the shortcut is advertised by IS-IS, the route will disappear from the RTM as the metric of the shortcut is greater than the IGP cost.

```
*A:PE-6# show router route-table 192.0.2.1/32
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type    Proto    Age          Pref
      Next Hop[Interface Name]                      Metric
-----
192.0.2.1/32                                     Remote  ISIS     00h00m04s    15
      192.168.46.1                      30
-----
No. of Routes: 1
```

RSVP Shortcut for IGP Route Resolution

If the LSP is reconfigured to use a metric equal to or smaller than the IGP cost, the router PE-6 will use the RSVP shortcut again. In the example, the LSP is reconfigured with a metric of 30:

```
*A:PE-6# configure router mpls lsp "LSP-PE-6-PE-1-strict" metric 30
```

Now the shortcut shows up as the preferred next hop to reach PE-1 from PE-6.

```
*A:PE-6# show router route-table 192.0.2.1/32
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                Type   Proto   Age           Pref
  Next Hop[Interface Name]                Metric
-----
192.0.2.1/32                      Remote ISIS   00h00m06s    15
    192.0.2.1 (tunneled:RSVP:2)                30
-----
No. of Routes: 1
```

As explained earlier, this could be combined together with ECMP, so if ECMP is configured to 2, the system shows the two equal cost paths.

```
*A:PE-6# configure router ecmp 2

*A:PE-6# show router route-table 192.0.2.1/32
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                Type   Proto   Age           Pref
  Next Hop[Interface Name]                Metric
-----
192.0.2.1/32                      Remote ISIS   00h00m05s    15
    192.0.2.1 (tunneled:RSVP:2)                30
192.0.2.1/32                      Remote ISIS   00h00m05s    15
    192.168.46.1                             30
-----
No. of Routes: 2
```

Checking GRT on PE-4, it displays the route to reach PE-1 (192.0.2.1/32) with a metric of 20 via PE-2 as next-hop. Although now PE-6 is announcing the RSVP LSP-PE-6-PE-1 to the other routers, the LSP shortcut is not used by PE-4 because the metric to reach PE-6 (10) plus the metric of the LSP shortcut from PE-6 to PE-1 (metric 30) is greater than 20.

```
*A:PE-4# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                Type   Proto   Age           Pref
  Next Hop[Interface Name]                Metric
-----
192.0.2.1/32                      Remote ISIS   01h57m14s    15
    192.168.24.1                     20
192.0.2.2/32                      Remote ISIS   01h57m14s    15
```

```

192.168.24.1
192.0.2.3/32 Remote ISIS 01h57m14s 15
192.168.24.1
192.0.2.4/32 Local Local 01h58m26s 0
system
192.0.2.5/32 Remote ISIS 01h57m08s 15
192.168.45.2
192.0.2.6/32 Remote ISIS 01h57m01s 15
192.168.46.2
192.168.12.0/30 Remote ISIS 01h57m14s 15
192.168.24.1
192.168.13.0/30 Remote ISIS 01h57m14s 15
192.168.24.1
192.168.23.0/30 Remote ISIS 01h57m14s 15
192.168.24.1
192.168.24.0/30 Local Local 01h58m26s 0
int-PE-4-PE-2
192.168.35.0/30 Remote ISIS 01h57m08s 15
192.168.45.2
192.168.45.0/30 Local Local 01h58m26s 0
int-PE-4-PE-5
192.168.46.0/30 Local Local 01h58m26s 0
int-PE-4-PE-6
192.168.56.0/30 Remote ISIS 01h57m08s 15
192.168.45.2

```

No. of Routes: 14

If the metric of the LSP LSP-PE-6-PE-1 is modified to a value between 1 and 9, there is a better metric (less than 20) so that PE-4 will change the next hop via PE-6. First the metric of the LSP is modified to 9:

```
*A:PE-6# configure router mpls lsp "LSP-PE-6-PE-1-strict" metric 9
```

And checking PE-4's GRT the next hop to reach PE-1 has changed, from next-hop PE-2 to next-hop PE-6 (hence, using the LSP shortcut), and the metric is 19 (10 to reach PE-6 plus metric 9 of the LSP PE-6-PE-1 shortcut):

```

*A:PE-4# show router route-table 192.0.2.1/32
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                Type   Proto   Age      Pref
Next Hop[Interface Name]          Metric
-----
192.0.2.1/32                      Remote ISIS 00h00m07s 15
192.168.46.2                      19
-----
No. of Routes: 1

```

Because the metric of the LSP shortcut was modified to a value of 9, when displaying the GRT of PE-6 it is noted that the next hops of several routes have changed and are also using the shortcut LSP PE-6-PE-1 because the metric is better than the regular IS-IS metric. It is important to

emphasize that IGP shortcuts will not be used to resolve prefixes downstream of the LSP endpoint when the LSP metric is higher than the underlying IGP cumulative metric.

```
*A:PE-6# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type   Proto   Age           Pref
Next Hop[Interface Name]                          Metric
-----
192.0.2.1/32                                       Remote ISIS   00h01m15s    15
192.0.2.1 (tunneled:RSVP:2)                       9
192.0.2.2/32                                       Remote ISIS   00h01m15s    15
192.0.2.2 (tunneled:RSVP:2)                       19
192.0.2.3/32                                       Remote ISIS   00h01m15s    15
192.0.2.3 (tunneled:RSVP:2)                       19
192.0.2.4/32                                       Remote ISIS   00h11m25s    15
192.168.46.1                                       10
192.0.2.5/32                                       Remote ISIS   00h11m25s    15
192.168.56.1                                       10
192.0.2.6/32                                       Local  Local    02h01m04s    0
system
192.168.12.0/30                                    Remote ISIS   00h01m15s    15
192.0.2.1 (tunneled:RSVP:2)                       19
192.168.13.0/30                                    Remote ISIS   00h01m15s    15
192.0.2.1 (tunneled:RSVP:2)                       19
192.168.23.0/30                                    Remote ISIS   00h01m15s    15
192.0.2.1 (tunneled:RSVP:2)                       29
192.168.24.0/30                                    Remote ISIS   00h11m25s    15
192.168.46.1                                       20
192.168.35.0/30                                    Remote ISIS   00h11m25s    15
192.168.56.1                                       20
192.168.45.0/30                                    Remote ISIS   00h10m34s    15
192.168.46.1                                       20
192.168.46.0/30                                    Local  Local    02h01m04s    0
int-PE-6-PE-4                                     0
192.168.56.0/30                                    Local  Local    02h01m04s    0
int-PE-6-PE-5                                     0
-----
No. of Routes: 14
```

Note that there are also cases where an LDP FEC can resolve to an RSVP LSP, if the user enables the LDP-over-RSVP feature or IGP shortcut feature when prefer-tunnel-in-tunnel is enabled in LDP and the endpoint of the RSVP LSP matches the FEC prefix. For those cases, the metric to the prefix is the sum of the RSVP LSP metric + remaining IGP path cost.

[Table 2](#) provides a summary of the outcome when configuring the forwarding adjacency, LDPoRSVP and RSVP shortcut options at both the IGP instance level and at the LSP level.

Table 2: RSVP LSP Role As Outcome of LSP Level and IGP Level Configuration Options

	IGP Instance Level Configurations					
LSP Level Configuration	advertise-tunnel-link enabled/ rsvp-shortcut enabled/ ldp-over-rsvp enabled	advertise-tunnel-link enabled/ rsvp-shortcut enabled/ ldp-over-rsvp disabled	advertise-tunnel-link enabled/ rsvp-shortcut disabled/ ldp-over-rsvp disabled	advertise-tunnel-link disabled/ rsvp-shortcut disabled/ ldp-over-rsvp disabled	advertise-tunnel-link disabled/ rsvp-shortcut enabled/ ldp-over-rsvp enabled	advertise-tunnel-link disabled/ rsvp-shortcut disabled/ ldp-over-rsvp enabled
igp-shortcut enabled/ldp-over-rsvp enabled	Forwarding Adjacency	Forwarding Adjacency	Forwarding Adjacency	None	IGP Shortcut	LDP-over-RSVP
igp-shortcut enabled/ldp-over-rsvp disabled	Forwarding Adjacency	Forwarding Adjacency	Forwarding Adjacency	None	IGP Shortcut	None
igp-shortcut disabled/ldp-over-rsvp enabled	None	None	None	None	None	LDP-over-RSVP
igp-shortcut disabled/ldp-over-rsvp disabled	None	None	None	None	None	None

LSP Relative Metric

It is possible to use relative metrics for IGP shortcuts as per RFC 3906, *Calculating Interior Gateway Protocol (IGP) Routes Over Traffic Engineering Tunnels*, with the following command:

```
*A:PE-6# configure router mpls lsp "LSP-PE-6-PE-1-loose" igp-shortcut relative-metric
- igp-shortcut [lfa-protect | lfa-only] [relative-metric [offset]]
- no igp-shortcut

<lfa-protect>      : keyword
<lfa-only>         : keyword
<relative-metric>  : keyword
<offset>           : [-10..10]
```

When this feature is enabled, IGP applies the shortest IGP cost between the endpoints of the LSP, plus the value of a configured offset when computing the cost of the prefix that is resolved to the LSP.

The offset value is optional and can have a value between -10 and 10, and defaults to zero (0). An offset value of zero (0) is used when the relative-metric option is enabled without specifying the offset parameter value. The minimum net cost for the prefix is capped to the value of one (1) after applying the offset:

Prefix cost = max (1, IGP Cost + relative metric offset)

The **relative-metric** option is ignored when advertise-tunnel-link is enabled in IS-IS or OSPF, in that case, the IGP advertises the LSP as a P2P unnumbered link using the LSP operational metric.

The **relative-metric** option is mutually exclusive with the **lfa-protect** (LFA:Loop-Free Alternate) or the **lfa-only** options. An LSP with **relative-metric** option enabled cannot be included in the LFA SPF and vice-versa when the **rsvp-shortcut** option is enabled in the IGP (see [LDP/IP FRR LFA for IGP Shortcut Using IS-IS/OSPF on page 461](#) for more information).

The offset can be used to enforce the preference of the shortcut path over the other paths for the prefix. Using an example, a new CSPF LSP with empty path and relative metric of -10 is created between PE-6 and PE-1. While the operational or absolute metric is 30 (IGP cost and populated in the Tunnel Table Manager, TTM), the metric that the RTM shows is 20 after applying the offset:

```
*A:PE-6# configure router mpls
      lsp "LSP-PE-6-PE-1-loose"
        to 192.0.2.1
        cspf
        igp-shortcut relative-metric -10
        primary "loose"
        exit
        no shutdown
      exit

*A:PE-6# show router tunnel-table 192.0.2.1
=====
```



```

IPv4 Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId  Pref    Nexthop      Metric
-----
192.0.2.1/32     rsvp      MPLS   2          7      192.168.56.1 16777215
192.0.2.1/32     rsvp      MPLS   3          7      192.168.56.1 30
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
*A:PE-6#
*A:PE-6# show router route-table 192.0.2.1
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type    Proto    Age          Pref
      Next Hop[Interface Name]                      Metric
-----
192.0.2.1/32                                         Remote  ISIS     00h00m07s    15
      192.0.2.1 (tunneled:RSVP:3)                    20
-----
No. of Routes: 1

```

LDP/IP FRR LFA for IGP Shortcut Using IS-IS/OSPF

MPLS LDP/IP FRR LFA for IGP shortcuts adds the use of RSVP-LSP-based IGP shortcuts as a Loop-Free Alternate (LFA) backup to expand the coverage of the IP Fast-Reroute (FRR) capability and the LDP FRR capability for IS-IS and OSPF prefixes. For a detailed description about IP and LDP FRR, refer to [MPLS LDP FRR using ISIS as IGP on page 563](#).

When an RSVP LSP is used as a shortcut by IS-IS or OSPF, it is included by the SPF as a P2P link and it can also be optionally advertised into the rest of the network by the IGP.

Two LSP-level configuration options are provided:

- The `lfa-protect` option includes the RSVP LSP in both the main SPF and the LFA SPFs. If the prefix primary Next-Hop (NH) is tunneled, no LFA NH is computed. The protection in this case is provided by RSVP FRR. If the prefix primary NH is direct, then an LFA NH is computed. A direct LFA NH is preferred over a tunneled LFA NH. Within each LFA NH type, node-protection is preferred over link-protection. The configuration command is:

```
configure router mpls lsp <lsp-name> igp-shortcut lfa-protect
```

- The `lfa-only` option includes the LSP in the LFA SPFs only so that the introduction of IGP shortcuts does not impact the main SPF decision. The prefix primary NH is always direct and the prefix LFA NH is computed. A direct LFA NH is preferred over a tunneled LFA NH. Within each LFA NH type, node-protection is preferred over link-protection. The configuration command is:

```
configure router mpls lsp <lsp-name> igp-shortcut lfa-only
```

LDP/IP FRR is a local decision so it can be enabled per node and there are no interoperability issues with other nodes. In the topology, PE-2 is provisioned with IS-IS LFA (OSPF configuration for the rest of this section is similar):

```
A:PE-2# configure router isis loopfree-alternate
```

The second item to configure is whether LDP or IP FRR is provisioned. To configure ip-fast-reroute, the command is:

```
A:PE-2# configure router ip-fast-reroute
```

Note: IP FRR feature for IS-IS/OSPF prefixes is supported on 7950 XRS, on 7750 SR-7/12/12e in chassis mode D, on the 7450 ESS-6/6v/7/12 in chassis mode D with or without mixed-mode, and 7750 SR-c4/c12.

To configure LDP FRR (no chassis dependency), this command is used:

```
A:PE-2# configure router ldp fast-reroute
```

Note: Although not shown, it is recommended to enable IGP-LDP synchronization per interface to avoid possible traffic blackholes.

Once LFA is enabled in all routers of the topology, looking at PE-2 (the configuration is done only on this node), the following command shows the LFA coverage where 4/5 nodes and 7/10 IPv4 prefixes are protected (IPv6 is not configured). Note that the output shows L1 and L2 because this node is provisioned as an L1-L2 IS-IS router. PE-2, PE-3, PE-4 and PE-5 share the same results, whereas only PE-1 and PE-6 have a 100% of coverage.

```
*A:PE-2# show router isis lfa-coverage
=====
Router Base ISIS Instance 0 LFA Coverage
=====
Topology          Level  Node          IPv4          IPv6
-----
IPv4 Unicast      L1     4/5 (80%)     7/10 (70%)    0/0 (0%)
IPv6 Unicast      L1     0/0 (0%)      0/0 (0%)      0/0 (0%)
IPv4 Multicast    L1     0/0 (0%)      0/0 (0%)      0/0 (0%)
IPv6 Multicast    L1     0/0 (0%)      0/0 (0%)      0/0 (0%)
IPv4 Unicast      L2     4/5 (80%)     7/10 (70%)    0/0 (0%)
IPv6 Unicast      L2     0/0 (0%)      0/0 (0%)      0/0 (0%)
IPv4 Multicast    L2     0/0 (0%)      0/0 (0%)      0/0 (0%)
IPv6 Multicast    L2     0/0 (0%)      0/0 (0%)      0/0 (0%)
=====

*A:PE-2#
*A:PE-1# show router isis lfa-coverage
=====
Router Base ISIS Instance 0 LFA Coverage
=====
Topology          Level  Node          IPv4          IPv6
-----
```

```

IPV4 Unicast    L1      5/5 (100%)    11/11 (100%)    0/0 (0%)
IPV6 Unicast    L1      0/0 (0%)      0/0 (0%)      0/0 (0%)
IPV4 Multicast  L1      0/0 (0%)      0/0 (0%)      0/0 (0%)
IPV6 Multicast  L1      0/0 (0%)      0/0 (0%)      0/0 (0%)
IPV4 Unicast    L2      5/5 (100%)    11/11 (100%)    0/0 (0%)
IPV6 Unicast    L2      0/0 (0%)      0/0 (0%)      0/0 (0%)
IPV4 Multicast  L2      0/0 (0%)      0/0 (0%)      0/0 (0%)
IPV6 Multicast  L2      0/0 (0%)      0/0 (0%)      0/0 (0%)
=====
*A:PE-1#

```

Taking a deeper look into the IS-IS LFA on PE-2, it can be seen that the node which is not protected is PE-4 (system address 192.0.2.4, since it is the one missing):

```

*A:PE-2# show router route-table alternative | match LFA pre-lines 2
192.0.2.1/32                                Remote  ISIS      02h22m46s  15
    192.168.12.1                            10
    192.168.23.2 (LFA)                      20
192.0.2.3/32                                Remote  ISIS      02h22m38s  15
    192.168.23.2                            10
    192.168.12.1 (LFA)                      20
192.0.2.5/32                                Remote  ISIS      02h22m18s  15
    192.168.23.2                            20
    192.168.24.2 (LFA)                      20
192.0.2.6/32                                Remote  ISIS      02h22m08s  15
    192.168.24.2                            20
    192.168.23.2 (LFA)                      30
192.168.13.0/30                             Remote  ISIS      02h22m46s  15
    192.168.12.1                            20
    192.168.23.2 (LFA)                      30
192.168.35.0/30                             Remote  ISIS      02h22m38s  15
    192.168.23.2                            20
    192.168.12.1 (LFA)                      30
192.168.56.0/30                             Remote  ISIS      02h22m17s  15
    192.168.23.2                            30
    192.168.24.2 (LFA)                      30
Flags: n = Number of times nexthop is repeated
        Backup = BGP backup route
        LFA = Loop-Free Alternate nexthop
*A:PE-2#

```

LFA is improved by taking advantage of RSVP shortcuts when it is properly provisioned. The reason why PE-4 cannot be protected with an LFA path is because the direct NH is using the direct link between PE-2 and PE-4 (the shortest IGP) and the intended LFA path through PE-3 is not valid (when LFA tries to find an alternate path via PE-3, the IGP cost from PE-3 to PE-4 is the same going via PE-5 then the path back via PE-2, invalidating that LFA calculation as there is a loop). This is normal as PE-2, PE-3, PE-4 and PE-5 are forming a ring. LFA coverage is increased by adding a link between PE-2 and PE-5, which can be done using a physical link or a virtual link with an RSVP shortcut. From the two possible options (lfa-only and lfa-protect), a new LSP “LSP-PE-2-PE-5” is configured with igp-shortcut lfa-only.

```

*A:PE-2# configure router mpls
        path "path-to-PE-5"

```

RSVP Shortcut for IGP Route Resolution

```
hop 10 192.0.2.3 strict
hop 20 192.0.2.5 strict
no shutdown
exit
lsp "LSP-PE-2-PE-5"
to 192.0.2.5
  igp-shortcut lfa-only
  primary "path-to-PE-5"
exit
no shutdown
exit
```

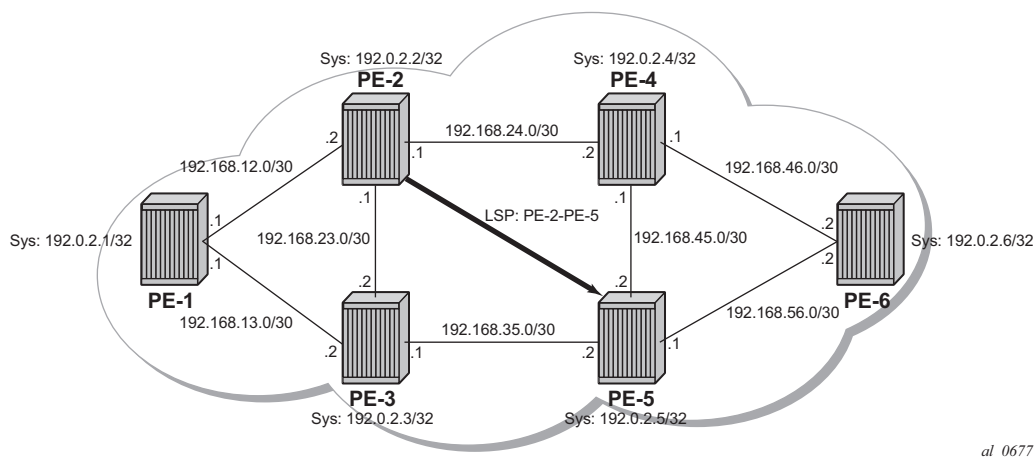


Figure 84: RSVP Shortcuts LFA Use Case Example

Now the coverage is 100% as shown by the output:

```
*A:PE-2# show router isis lfa-coverage
=====
Router Base ISIS Instance 0 LFA Coverage
=====
Topology      Level  Node      IPv4      IPv6
-----
IPV4 Unicast  L1     5/5 (100%) 10/10 (100%) 0/0 (0%)
IPV6 Unicast  L1     0/0 (0%)  0/0 (0%)  0/0 (0%)
IPV4 Multicast L1     0/0 (0%)  0/0 (0%)  0/0 (0%)
IPV6 Multicast L1     0/0 (0%)  0/0 (0%)  0/0 (0%)
IPV4 Unicast  L2     5/5 (100%) 10/10 (100%) 0/0 (0%)
IPV6 Unicast  L2     0/0 (0%)  0/0 (0%)  0/0 (0%)
IPV4 Multicast L2     0/0 (0%)  0/0 (0%)  0/0 (0%)
IPV6 Multicast L2     0/0 (0%)  0/0 (0%)  0/0 (0%)
=====
*A:PE-2#
```

The GRT details the prefix information after the new LFA calculation using the lfa-only option (the shortcut is used by LFA SPF). Note that the metric from PE-2 to PE-4 is the maximum plus

the IGP cost (16777215 + 10) and that the shortcut is also used to protect the rest of the previously unprotected prefixes:

```
*A:PE-2# show router route-table alternative | match LFA pre-lines 2
192.0.2.1/32 Remote ISIS 02h27m59s 15
    192.168.12.1 10
    192.168.23.2 (LFA) 20
192.0.2.3/32 Remote ISIS 02h27m51s 15
    192.168.23.2 10
    192.168.12.1 (LFA) 20
192.0.2.4/32 Remote ISIS 02h27m36s 15
    192.168.24.2 10
    192.0.2.5 (LFA) (tunneled:RSVP:1) 16777225
192.0.2.5/32 Remote ISIS 02h27m31s 15
    192.168.23.2 20
    192.168.24.2 (LFA) 20
192.0.2.6/32 Remote ISIS 02h27m21s 15
    192.168.24.2 20
    192.168.23.2 (LFA) 30
192.168.13.0/30 Remote ISIS 02h27m59s 15
    192.168.12.1 20
    192.168.23.2 (LFA) 30
192.168.35.0/30 Remote ISIS 02h27m51s 15
    192.168.23.2 20
    192.168.12.1 (LFA) 30
192.168.45.0/30 Remote ISIS 02h27m36s 15
    192.168.24.2 20
    192.0.2.5 (LFA) (tunneled:RSVP:1) 16777235
192.168.46.0/30 Remote ISIS 02h27m36s 15
    192.168.24.2 20
    192.0.2.5 (LFA) (tunneled:RSVP:1) 16777235
192.168.56.0/30 Remote ISIS 02h27m30s 15
    192.168.23.2 30
    192.168.24.2 (LFA) 30
Flags: n = Number of times nexthop is repeated
Backup = BGP backup route
LFA = Loop-Free Alternate nexthop
*A:PE-2#
```

The tunnel table shows the RSVP LSP used as a shortcut and its operational metric.

```
*A:PE-2# show router tunnel-table 192.0.2.5
=====
IPv4 Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId Pref  Nexthop      Metric
-----
192.0.2.5/32     rsvp      MPLS    1       7      192.168.23.2 16777215
192.0.2.5/32     ldp       MPLS    65540   9      192.168.23.2 20
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
*A:PE-2#
```

RSVP Shortcut for IGP Route Resolution

Now, if the LSP “LSP-PE-2-PE-5” is provisioned with lfa-protect instead of lfa-only, the result is that the LSP "LSP-PE-2-PE-5" is used by normal SPF to define the primary NH and it is not used by LFA SPF anymore.

```
A:PE-2# configure router mpls lsp "LSP-PE-2-PE-5" igp-shortcut lfa-protect
```

The coverage when lfa-protect is used also shows a 100% for nodes. The 112% coverage for prefixes shown below is not correct and should be 100% since the IGP shortcut used as a primary next-hop to reach PE-5 from PE-2 is providing protection for prefixes reachable via PE-5. The display issue will be fixed in a later release and will be referenced by [201872] in the release notes.

```
*A:PE-2# show router isis lfa-coverage
=====
Router Base ISIS Instance 0 LFA Coverage
=====
Topology          Level   Node          IPv4           IPv6
-----
IPv4 Unicast      L1     4/4 (100%)    9/8 (112%)     0/0 (0%)
IPv6 Unicast      L1     0/0 (0%)     0/0 (0%)       0/0 (0%)
IPv4 Multicast    L1     0/0 (0%)     0/0 (0%)       0/0 (0%)
IPv6 Multicast    L1     0/0 (0%)     0/0 (0%)       0/0 (0%)
IPv4 Unicast      L2     4/4 (100%)    9/8 (112%)     0/0 (0%)
IPv6 Unicast      L2     0/0 (0%)     0/0 (0%)       0/0 (0%)
IPv4 Multicast    L2     0/0 (0%)     0/0 (0%)       0/0 (0%)
IPv6 Multicast    L2     0/0 (0%)     0/0 (0%)       0/0 (0%)
=====
*A:PE-2#
```

In this case the GRT looks as follows, the main difference being that now PE-5 (192.0.2.5) has a direct shortcut from PE-2:

```
*A:PE-2# show router route-table alternative
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type  Proto  Age           Pref
      Next Hop[Interface Name]                    Metric
      Alt-NextHop                                   Alt-
                                                    Metric
-----
192.0.2.1/32                                         Remote  ISIS    02h32m17s    15
      192.168.12.1                                  10
      192.168.23.2 (LFA)                             20
192.0.2.2/32                                         Local   Local   02h33m31s     0
      system                                           0
192.0.2.3/32                                         Remote  ISIS    02h32m09s    15
      192.168.23.2                                  10
      192.168.12.1 (LFA)                             20
192.0.2.4/32                                         Remote  ISIS    02h31m54s    15
      192.168.24.2                                  10
      192.0.2.5 (LFA) (tunneled:RSVP:1)              16777225
192.0.2.5/32                                         Remote  ISIS    00h01m56s    15
```

192.0.2.5 (tunneled:RSVP:1)			16777215
192.0.2.6/32	Remote	ISIS	02h31m39s 15
192.168.24.2			20
192.168.23.2 (LFA)			30
192.168.12.0/30	Local	Local	02h33m31s 0
int-PE-2-PE-1			0
192.168.13.0/30	Remote	ISIS	02h32m17s 15
192.168.12.1			20
192.168.23.2 (LFA)			30
192.168.23.0/30	Local	Local	02h33m31s 0
int-PE-2-PE-3			0
192.168.24.0/30	Local	Local	02h33m31s 0
int-PE-2-PE-4			0
192.168.35.0/30	Remote	ISIS	02h32m09s 15
192.168.23.2			20
192.168.12.1 (LFA)			30
192.168.45.0/30	Remote	ISIS	02h31m54s 15
192.168.24.2			20
192.0.2.5 (LFA) (tunneled:RSVP:1)			16777235
192.168.46.0/30	Remote	ISIS	02h31m54s 15
192.168.24.2			20
192.0.2.5 (LFA) (tunneled:RSVP:1)			16777235
192.168.56.0/30	Remote	ISIS	00h01m56s 15
192.168.24.2			30
192.168.23.2 (LFA)			40

No. of Routes: 14

Rules Determining the Installation of Shortcuts into RTM

Although it was already mentioned in the RSVP-TE LSP shortcut for IGP route resolution section, the rules determining how shortcuts are installed into RTM are (sorted by higher priority):

1. RSVP shortcut.
2. LDP shortcut.
3. IGP route with regular IP next-hop.

The implementation is compliant with RFC3906.

To check the rules, the network configuration is iLDP in all interfaces with LDP shortcuts enabled, there is also an RSVP LSP from PE-6 to PE-3 available but RSVP shortcuts are disabled. The topology is shown in [Figure 85](#).

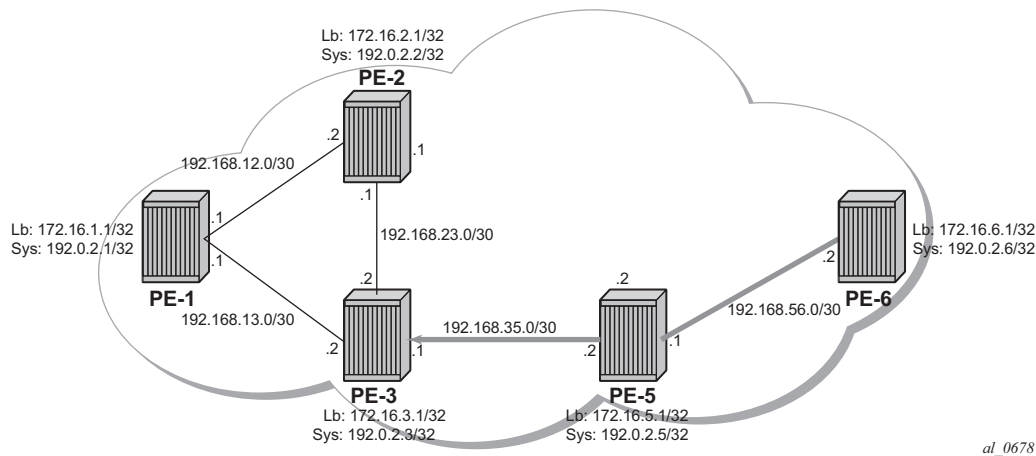


Figure 85: Network Topology to Verify Installation of Shortcuts into RTM

The following RSVP LSP is needed between PE-6 and PE-3.

```
configure router ldp-shortcut
configure router isis no rsvp-shortcut

configure router mpls
  path "loose"
  no shutdown
exit
lsp "LSP-PE-6-PE-3"
  to 192.0.2.3
  cspf
  primary "loose"
  exit
  no shutdown
exit
```

Displaying relevant info in PE-6, the routes are:

```
*A:PE-6# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type  Proto  Age           Pref
Next Hop[Interface Name]                          Metric
-----
192.0.2.1/32                                       Remote LDP    00h00m03s    9
192.168.56.1 (tunneled)                          30
192.0.2.2/32                                       Remote LDP    00h00m03s    9
```



```

192.168.56.1 (tunneled) 30
192.0.2.3/32 Remote LDP 00h00m03s 9
192.168.56.1 (tunneled) 20
192.0.2.5/32 Remote LDP 00h20m48s 9
192.168.56.1 (tunneled) 10
192.0.2.6/32 Local Local 02h57m46s 0
system 0
192.168.12.0/30 Remote ISIS 00h00m03s 15
192.168.56.1 40
192.168.13.0/30 Remote ISIS 00h00m03s 15
192.168.56.1 30
192.168.23.0/30 Remote ISIS 00h00m03s 15
192.168.56.1 30
192.168.35.0/30 Remote ISIS 00h00m03s 15
192.168.56.1 20
192.168.56.0/30 Local Local 02h57m46s 0
int-PE-6-PE-5 0
-----
No. of Routes: 10

```

The Tunnel Table shows the LSPs available for the shortcuts, and hence these are used in the GRT for LDP (but not for RSVP):

```

*A:PE-6# show router tunnel-table
=====
IPv4 Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId Pref  Nexthop      Metric
-----
192.0.2.1/32     ldp       MPLS  65564    9    192.168.56.1  30
192.0.2.2/32     ldp       MPLS  65565    9    192.168.56.1  30
192.0.2.3/32     rsvp      MPLS   4        7    192.168.56.1  20
192.0.2.3/32     ldp       MPLS  65566    9    192.168.56.1  20
192.0.2.5/32     ldp       MPLS  65541    9    192.168.56.1  10
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
*A:PE-6#

```

So far, LDP shortcuts are preferred over the IGP next-hops for the system addresses (router-id). After enabling RSVP shortcuts under the IS-IS context, the changes in the GRT are:

```

*A:PE-6# configure router isis rsvp-shortcut

*A:PE-6# show router route-table next-hop-type tunneled
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]      Type  Proto  Age      Pref
Next Hop[Interface Name] Metric
-----
192.0.2.1/32           Remote ISIS  00h00m08s  15
192.0.2.3 (tunneled:RSVP:4) 30
192.0.2.2/32           Remote ISIS  00h00m08s  15

```

RSVP Shortcut for IGP Route Resolution

```

192.0.2.3 (tunneled:RSVP:4)                               30
192.0.2.3/32 Remote ISIS 00h00m08s 15
192.0.2.3 (tunneled:RSVP:4)                               20
192.0.2.5/32 Remote LDP 00h23m42s 9
192.168.56.1 (tunneled)                                   10
192.168.12.0/30 Remote ISIS 00h00m08s 15
192.0.2.3 (tunneled:RSVP:4)                               40
192.168.13.0/30 Remote ISIS 00h00m08s 15
192.0.2.3 (tunneled:RSVP:4)                               30
192.168.23.0/30 Remote ISIS 00h00m08s 15
192.0.2.3 (tunneled:RSVP:4)                               30
-----
No. of Routes: 7

```

The GRT shows that PE-6 is using an LDP shortcut to reach PE-5, but PE-6 is using the RSVP shortcut to reach not only PE-3's system address, but also PE-1 and PE-2 routes (including all interfaces) which were behind the RSVP LSP shortcut.

In summary, the behavior is:

- When resolving a prefix, SPF picks the RSVP shortcut next-hop if there is an RSVP LSP directly to that address regardless of the IGP path cost compared to the IGP next-hop. When multiple RSVP LSPs to that address exist and all have the same lowest metric, if ECMP is enabled on the system, the LSP with lowest tunnel ID is chosen. In this example, if LSP "LSP-PE-6-PE-3" is provisioned with a metric of 100 (IGP metric is 20), the GRT shows that the PE-3 system address is reachable via the LSP.

```

*A:PE-6# show router route-table 192.0.2.3
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                     Type  Proto  Age      Pref
Next Hop[Interface Name]                               Metric
-----
192.0.2.3/32 Remote ISIS 00h00m12s 15
192.0.2.3 (tunneled:RSVP:4)                               100
-----
No. of Routes: 1

```

- SPF also picks the RSVP LSP shortcut if both the LSP path and the IGP path to the prefix are via the tail-end of the LSP. This is regardless of the path cost compared to the IGP next-hop. When paths over multiple RSVP shortcuts have the same lowest cost, if ECMP is enabled on the system, the LSP with lowest tunnel ID is chosen. In this example, 192.168.13.0 and 192.168.23.0 are using the shortcut but 192.168.12.0 is not.

```

*A:PE-6# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                     Type  Proto  Age      Pref
Next Hop[Interface Name]                               Metric
-----
---snip---

```

192.168.12.0/30	Remote	ISIS	00h00m46s	15
192.168.56.1			40	
192.168.13.0/30	Remote	ISIS	00h00m46s	15
192.0.2.3 (tunneled:RSVP:4)			110	
192.168.23.0/30	Remote	ISIS	00h00m46s	15
192.0.2.3 (tunneled:RSVP:4)			110	

---snip---

No. of Routes: 10

LDP/RSVP LSP Shortcut for BGP NH Resolution

Using LDP/RSVP LSP shortcuts for resolving BGP next-hops allows IPv4 packet forwarding to routes resolved via a BGP next-hop using an LDP/RSVP LSP instead of using a regular IP next-hop. In the network topology of [Figure 82](#), both PE-3 and PE-6 have a single peer configured, initially without any shortcuts enabled under the BGP context. Also, one static route is configured in PE-3 and PE-6 and that is redistributed into BGP. The relevant configuration on PE-3 is the following:

```
*A:PE-3# configure router
      interface "static-route"
        address 172.16.33.1/30
        port 1/1/4:33
      exit
      autonomous-system 65536

      static-route 10.10.10.0/24 next-hop 172.16.33.2

      policy-options
        begin
        policy-statement "static-routes"
          description "export static-routes for I-BGP"
          entry 10
            from
              protocol static
            exit
            to
              protocol bgp
            exit
            action accept
              next-hop-self
            exit
          exit
        exit
      exit
      commit
    exit

    bgp
      export "static-routes"
      group "ibgp"
        type internal
        neighbor 192.0.2.6
      exit
    exit
  exit
```

Checking the static route received on PE-6 via BGP, the next-hop is the PE-3 system address:

```
*A:PE-6# show router bgp routes 10.10.10.0/24 detail
=====
BGP Router ID:192.0.2.6      AS:65536      Local AS:65536
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP IPv4 Routes
=====
Original Attributes

Network       : 10.10.10.0/24
Nexthop       : 192.0.2.3
Path Id       : None
From          : 192.0.2.3
Res. Nexthop  : 192.168.56.1
Local Pref.   : 100
Aggregator AS : None
Atomic Aggr.  : Not Atomic
AIGP Metric   : None
Connector     : None
Community     : No Community Members
Cluster       : No Cluster Members
Originator Id : None
Fwd Class     : None
Flags         : Used Valid Best Incomplete
Route Source  : Internal
AS-Path       : No As-Path
Route Tag     : 0
Neighbor-AS   : N/A
Orig Validation: NotFound
Source Class  : 0
Add Paths Send : Default
Last Modified : 00h00m10s
Interface Name : int-PE-6-PE-5
Aggregator     : None
MED            : None
Peer Router Id : 192.0.2.3
Priority       : None
Dest Class     : 0

Modified Attributes

Network       : 10.10.10.0/24
Nexthop       : 192.0.2.3
Path Id       : None
From          : 192.0.2.3
Res. Nexthop  : 192.168.56.1
---snipped---
-----
Routes : 1
```

LDP/RSVP LSP Shortcut for BGP NH Resolution

The BGP peering configuration possibilities are LDP, RSVP, or BGP. In case both LDP and RSVP are included in the filter, RSVP is preferred. Disabling the IGP is also allowed (meaning that unless there is a shortcut, the BGP peering will not fall back to IGP):

```
*A:PE-6# configure router bgp next-hop-resolution shortcut-tunnel family ipv4 resolution
- resolution {any|filter|disabled}

*A:PE-6# configure router bgp next-hop-resolution shortcut-tunnel family ipv4 resolution-
filter
- resolution-filter

[no] bgp          - Use BGP tunnelling for next hop resolution
[no] ldp          - Use LDP tunnelling for next hop resolution
[no] rsvp         - Use RSVP tunnelling for next hop resolution

*A:PE-6#
```

When enabling LDP shortcuts on PE-6, the output changes showing the detail of the received BGP route indicating that the next hop is resolved using LDP:

```
*A:PE-6# configure router bgp
      next-hop-resolution
      shortcut-tunnel
      family ipv4
      resolution-filter
      ldp
      exit
      resolution filter
      exit
    exit
  exit

*A:PE-6# show router bgp routes 10.10.10.0/24 detail
=====
BGP Router ID:192.0.2.6      AS:65536      Local AS:65536
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP IPv4 Routes
=====
Original Attributes

Network      : 10.10.10.0/24
Nextthop     : 192.0.2.3
Path Id      : None
From         : 192.0.2.3
Res. Nextthop : 192.168.56.1 (LDP)
Local Pref.  : 100
Aggregator AS : None
Atomic Aggr. : Not Atomic
AIGP Metric  : None
Interface Name : int-PE-6-PE-5
Aggregator    : None
MED           : None
```

```

Connector      : None
Community      : No Community Members
Cluster        : No Cluster Members
Originator Id  : None                      Peer Router Id : 192.0.2.3
Fwd Class      : None                      Priority       : None
Flags          : Used Valid Best Incomplete
Route Source   : Internal
AS-Path        : No As-Path
Route Tag      : 0
Neighbor-AS    : N/A
Orig Validation: NotFound
Source Class   : 0                        Dest Class     : 0
Add Paths Send : Default
Last Modified  : 00h20m53s

```

Modified Attributes

```

Network      : 10.10.10.0/24
Nexthop      : 192.0.2.3
Path Id      : None
From         : 192.0.2.3
Res. Nexthop : 192.168.56.1 (LDP)
---snipped---

```

```

-----
Routes : 1

```

The GRT output command also shows that the route is reachable using LDP (indicated as tunneled):

```

*A:PE-6# show router route-table next-hop-type tunneled
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type   Proto   Age           Pref
      Next Hop[Interface Name]                      Metric
-----
10.10.10.0/24                                     Remote BGP     00h20m30s  170
      192.0.2.3 (tunneled)                          0
-----
No. of Routes: 1

```

It can be seen that the previously created LSP LSP-PE-6-PE-3 is up and running:

```

*A:PE-6# show router mpls lsp "LSP-PE-6-PE-3" path detail
=====
MPLS LSP LSP-PE-6-PE-3 Path (Detail)
=====
Legend :
  @ - Detour Available          # - Detour In Use
  b - Bandwidth Protected      n - Node Protected
  s - Soft Preemption
  S - Strict                    L - Loose
  A - ABR
=====
-----

```

LDP/RSVP LSP Shortcut for BGP NH Resolution

LSP LSP-PE-6-PE-3 Path loose

```

-----
LSP Name      : LSP-PE-6-PE-3                      Path LSP ID : 8196
From          : 192.0.2.6                          To          : 192.0.2.3
Adm State     : Up                                  Oper State  : Up
Path Name     : loose                              Path Type   : Primary
Path Admin    : Up                                  Path Oper   : Up
OutInterface  : 1/1/1                              Out Label   : 262139
Path Up Time  : 0d 00:01:18                        Path Dn Time: 0d 00:00:00
Retry Limit   : 0                                  Retry Timer  : 30 sec
RetryAttempt  : 0                                  NextRetryIn : 0 sec

Adspec        : Disabled                          Oper Adspec  : Disabled
CSPF          : Enabled                          Oper CSPF    : Enabled
Least Fill    : Disabled                        Oper LeastF* : Disabled
FRR           : Disabled                        Oper FRR     : Disabled
Prop Adm Grp  : Disabled                        Oper PropAG  : Disabled
Inter-area    : False

Neg MTU       : 1564                              Oper MTU     : 1564
Bandwidth     : No Reservation                    Oper Bw      : 0 Mbps
Hop Limit     : 255                              Oper HopLim* : 255
Record Route  : Record                          Oper RecRou* : Record
Record Label  : Record                          Oper RecLab* : Record
SetupPriori*  : 7                               Oper SetupP* : 7
Hold Priori*  : 0                               Oper HoldPr*  : 0
Class Type    : 0                               Oper CT       : 0
Backup CT     : None

MainCT Retry  : n/a
Rem           :
MainCT Retry  : 0
Limit        :
Include Grps  :
None
Exclude Grps :
None

Adaptive      : Enabled                          Oper Metric  : 20
Preference    : n/a
Path Trans    : 3                               CSPF Queries: 2
Failure Code  : noError                         Failure Node : n/a
ExplicitHops  :
    No Hops Specified
Actual Hops   :
    192.168.56.2 (192.0.2.6)
    -> 192.168.56.1 (192.0.2.5)
    -> 192.168.35.1 (192.0.2.3)
ComputedHops :
    192.168.56.2(S)
    -> 192.168.56.1(S)
    -> 192.168.35.1(S)
ResigEligib*  : False
LastResignal  : n/a                            CSPF Metric  : 20
=====
* indicates that the corresponding row element may have been truncated.
*A:PE-6#

```


After adding the **resolution-filter rsvp** to the shortcut-tunnel configuration in the bgp context, the output shows that the BGP peer is reachable using an RSVP LSP (switched from LDP to RSVP since RSVP is preferred):

```
*A:PE-6# configure router bgp
      next-hop-resolution
      shortcut-tunnel
        family ipv4
          resolution-filter
            ldp
            rsvp
          exit
        resolution filter
      exit
    exit
  exit
```

```
*A:PE-6# show router bgp routes ipv4 10.10.10.0/24 detail
=====
BGP Router ID:192.0.2.6      AS:65536      Local AS:65536
=====
Legend -
Status codes  : u - used, s - suppressed, h - history, d - decayed, * - valid
                l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP IPv4 Routes
=====
Original Attributes

Network       : 10.10.10.0/24
Nexthop       : 192.0.2.3
Path Id       : None
From          : 192.0.2.3
Res. Nexthop  : 192.168.56.1 (RSVP LSP: 4)
Local Pref.   : 100                               Interface Name : int-PE-6-PE-5
Aggregator AS : None                               Aggregator    : None
Atomic Aggr.  : Not Atomic                         MED           : None
AIGP Metric   : None
Connector     : None
Community     : No Community Members
Cluster       : No Cluster Members
Originator Id : None                               Peer Router Id : 192.0.2.3
Fwd Class     : None                               Priority       : None
Flags         : Used Valid Best Incomplete
Route Source  : Internal
AS-Path       : No As-Path
Route Tag     : 0
Neighbor-AS   : N/A
Orig Validation: NotFound
Source Class  : 0                                   Dest Class     : 0
Add Paths Send : Default
Last Modified  : 01h37m12s

Modified Attributes
```

LDP/RSVP LSP Shortcut for BGP NH Resolution

```
Network      : 10.10.10.0/24
Nexthop      : 192.0.2.3
Path Id      : None
From         : 192.0.2.3
Res. Nexthop : 192.168.56.1 (RSVP LSP: 4)
```

---snipped---

Routes : 1

The GRT output command also shows that the route is reachable using RSVP (indicated as tunneled:RSVP:4):

```
*A:PE-6# show router route-table next-hop-type tunneled
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type    Proto    Age          Pref
      Next Hop[Interface Name]                      Metric
-----
10.10.10.0/24                                     Remote  BGP       00h16m09s    170
      192.0.2.3 (tunneled:RSVP:4)                      0
-----
No. of Routes: 1
```

If the RSVP LSP is **shutdown**, the system reverts back to the LDP LSP:

```
*A:PE-6# configure router mpls lsp "LSP-PE-6-PE-3" shutdown

*A:PE-6# show router bgp routes 10.10.10.0/24 detail
=====
BGP Router ID:192.0.2.6      AS:65536      Local AS:65536
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP IPv4 Routes
=====
Original Attributes

Network      : 10.10.10.0/24
Nexthop      : 192.0.2.3
Path Id      : None
From         : 192.0.2.3
Res. Nexthop : 192.168.56.1 (LDP)
Local Pref.  : 100
Aggregator AS : None
Atomic Aggr. : Not Atomic
AIGP Metric  : None
Connector    : None
Community    : No Community Members
Cluster      : No Cluster Members

Interface Name : int-PE-6-PE-5
Aggregator     : None
MED            : None
```

```

Originator Id   : None                      Peer Router Id : 192.0.2.3
Fwd Class       : None                      Priority       : None
Flags           : Used Valid Best Incomplete
Route Source    : Internal
AS-Path         : No As-Path
Route Tag       : 0
Neighbor-AS     : N/A
Orig Validation : NotFound
Source Class    : 0                        Dest Class     : 0
Add Paths Send  : Default
Last Modified   : 01h49m57s

```

Modified Attributes

```

Network        : 10.10.10.0/24
Nexthop        : 192.0.2.3
Path Id        : None
From           : 192.0.2.3
Res. Nexthop   : 192.168.56.1 (LDP)

```

```
---snip---
```

```
-----
Routes : 1

```

When the shortcut-tunnel with resolution-filter rsvp is enabled at the BGP level, all RSVP LSPs originating on this node are eligible to be used by default as long as the destination address of the LSP corresponds to that of the BGP next-hop for that prefix. It is also possible to exclude a specific RSVP LSP from BGP next-hop resolution, similar to the exclusion of a specific RSVP LSP being used as a shortcut for resolving IGP routes. In this example, if the RSVP LSP "LSP-PE-6-PE-3" is excluded to be eligible for BGP next-hop resolution, it reverts back to LDP.

```

*A:PE-6# configure router mpls lsp "LSP-PE-6-PE-3"
      no bgp-shortcut
      no shutdown
      exit

```

```
*A:PE-6# show router route-table 10.10.10.0
```

```
=====
Route Table (Router: Base)
=====
```

Dest Prefix[Flags]	Type	Proto	Age	Pref
Next Hop[Interface Name]			Metric	
10.10.10.0/24	Remote	BGP	00h04m56s	170
192.0.2.3 (tunneled)			0	

```
-----
No. of Routes: 1

```

If the configuration is using **disallow-igp**, and neither LDP nor RSVP LSPs are available, the remote route received via BGP is removed from the GRT although the BGP peer session remains up. A field in the detailed show BGP route output indicates that the next hop is "Unresolved":

LDP/RSVP LSP Shortcut for BGP NH Resolution

```
*A:PE-6# configure router bgp
      next-hop-resolution
      shortcut-tunnel
      family ipv4
      resolution-filter
      ldp
      rsvp
      exit
      disallow-igp
      resolution filter
      exit
    exit
  exit
exit

*A:PE-6# configure router ldp shutdown

*A:PE-6# show router bgp routes 10.10.10.0/24 detail
=====
BGP Router ID:192.0.2.6      AS:65536      Local AS:65536
=====
Legend -
Status codes  : u - used, s - suppressed, h - history, d - decayed, * - valid
                l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP IPv4 Routes
=====
Original Attributes

Network       : 10.10.10.0/24
Nexthop       : 192.0.2.3
Path Id       : None
From          : 192.0.2.3
Res. Nexthop  : Unresolved
Local Pref.   : 100
Aggregator AS : None
Atomic Aggr.  : Not Atomic
AIGP Metric   : None
Connector     : None
Community     : No Community Members
Cluster       : No Cluster Members
Originator Id : None
Fwd Class     : None
Flags         : Invalid Incomplete Nexthop-Unresolved
Route Source  : Internal
AS-Path       : No As-Path
Route Tag     : 0
Neighbor-AS   : N/A
Orig Validation: NotFound
Source Class  : 0
Add Paths Send : Default
Last Modified : 03h20m08s

Interface Name : int-PE-6-PE-5
Aggregator     : None
MED            : None

Peer Router Id : 192.0.2.3
Priority        : None

Dest Class     : 0

Modified Attributes

Network       : 10.10.10.0/24
Nexthop       : 192.0.2.3
Path Id       : None
```

```

From          : 192.0.2.3
Res. Nexthop   : Unresolved
Local Pref.    : 100
Aggregator AS : None
Atomic Aggr.   : Not Atomic
AIGP Metric    : None
Connector      : None
Community      : No Community Members
Cluster        : No Cluster Members
Originator Id  : None
Fwd Class      : None
Flags          : Invalid Incomplete
Route Source   : Internal
AS-Path        : No As-Path
Route Tag      : 0
Neighbor-AS    : N/A
Orig Validation: NotFound
Source Class   : 0
Add Paths Send : Default
Last Modified  : 03h20m08s
Interface Name : int-PE-6-PE-5
Aggregator     : None
MED            : None
Peer Router Id : 192.0.2.3
Priority        : None
NextHop-Unresolved
Dest Class     : 0

```

```

-----
Routes : 1

```

As the route is unresolved, it does not appear in the GRT:

```
*A:PE-6# show router route-table
```

```

=====
Route Table (Router: Base)
=====

```

Dest Prefix[Flags] Next Hop[Interface Name]	Type	Proto	Age Metric	Pref
20.20.20.0/24	Remote	Static	03h26m25s	5
172.16.66.2			1	
172.16.6.1/32	Local	Local	00h00m11s	0
loopback			0	
172.16.66.0/30	Local	Local	03h26m25s	0
static-route			0	
192.0.2.1/32	Remote	ISIS	03h27m22s	15
192.168.46.1			30	
192.0.2.2/32	Remote	ISIS	03h27m23s	15
192.168.46.1			20	
192.0.2.3/32	Remote	ISIS	03h27m23s	15
192.168.56.1			20	
192.0.2.4/32	Remote	ISIS	03h27m23s	15
192.168.46.1			10	
192.0.2.5/32	Remote	ISIS	03h27m23s	15
192.168.56.1			10	
192.0.2.6/32	Local	Local	01d00h03m	0
system			0	
192.168.12.0/30	Remote	ISIS	03h27m23s	15
192.168.46.1			30	
192.168.13.0/30	Remote	ISIS	03h27m23s	15
192.168.56.1			30	
192.168.23.0/30	Remote	ISIS	03h27m22s	15
192.168.46.1			30	
192.168.24.0/30	Remote	ISIS	03h27m23s	15

LDP/RSVP LSP Shortcut for BGP NH Resolution

192.168.46.1			20	
192.168.35.0/30	Remote	ISIS	03h27m23s	15
192.168.56.1			20	
192.168.45.0/30	Remote	ISIS	03h27m22s	15
192.168.46.1			20	
192.168.46.0/30	Local	Local	03h27m24s	0
int-PE-6-PE-4			0	
192.168.56.0/30	Local	Local	01d00h03m	0
int-PE-6-PE-5			0	

No. of Routes: 17

MPLS/GRE Shortcut for BGP NH Resolution within a VRF

Using RSVP/LDP or GRE shortcuts for resolving BGP next-hops within a Virtual Private Routed Network (VPRN), also known as auto-bind-tunnel, allows a VPRN service to automatically resolve the BGP next-hop for VPRN routes to an MPLS LSP or a GRE tunnel. Three possible mechanisms to provide transport tunnels for forwarding traffic between PE routers within an RFC 4364, *BGP/MPLS IP Virtual Private Networks (VPNs)*, network exist:

- RSVP-TE protocol to create tunnel LSPs between PE routers.
- LDP protocol to create tunnel LSPs between PE routers.
- GRE tunnels between PE routers.

These transport tunneling mechanisms provide the flexibility to use dynamically created LSPs where the service tunnels are automatically bound (the “auto-bind-tunnel” feature), and the ability to provide certain VPN services with their own transport tunnels by explicitly binding SDPs if desired. All services using the auto-bind-tunnel feature use the same set of LSPs, which does not allow for alternate tunneling mechanisms (like GRE) or the ability to craft sets of LSPs with bandwidth reservations for specific customers, as is available with explicit SDPs for the service.

The auto-bind-tunnel configuration is as follows:

```
*A:PE-2# configure service vprn 1 auto-bind-tunnel resolution
- resolution {disabled|any|filter}

<disabled|any|filt*> : disabled|any|filter

*A:PE-2# configure service vprn 1 auto-bind-tunnel resolution-filter
- resolution-filter

[no] gre          - Enable/disable setting GRE type for auto-bind-tunnel
[no] ldp          - Enable/disable setting LDP type for auto-bind-tunnel
[no] rsvp         - Enable/disable setting RSVP-TE type for auto-bind-tunnel
[no] sr-isis      - Enable/disable setting SR-ISIS type for auto-bind-tunnel
[no] sr-ospf      - Enable/disable setting SR-OSPF type for auto-bind-tunnel
```

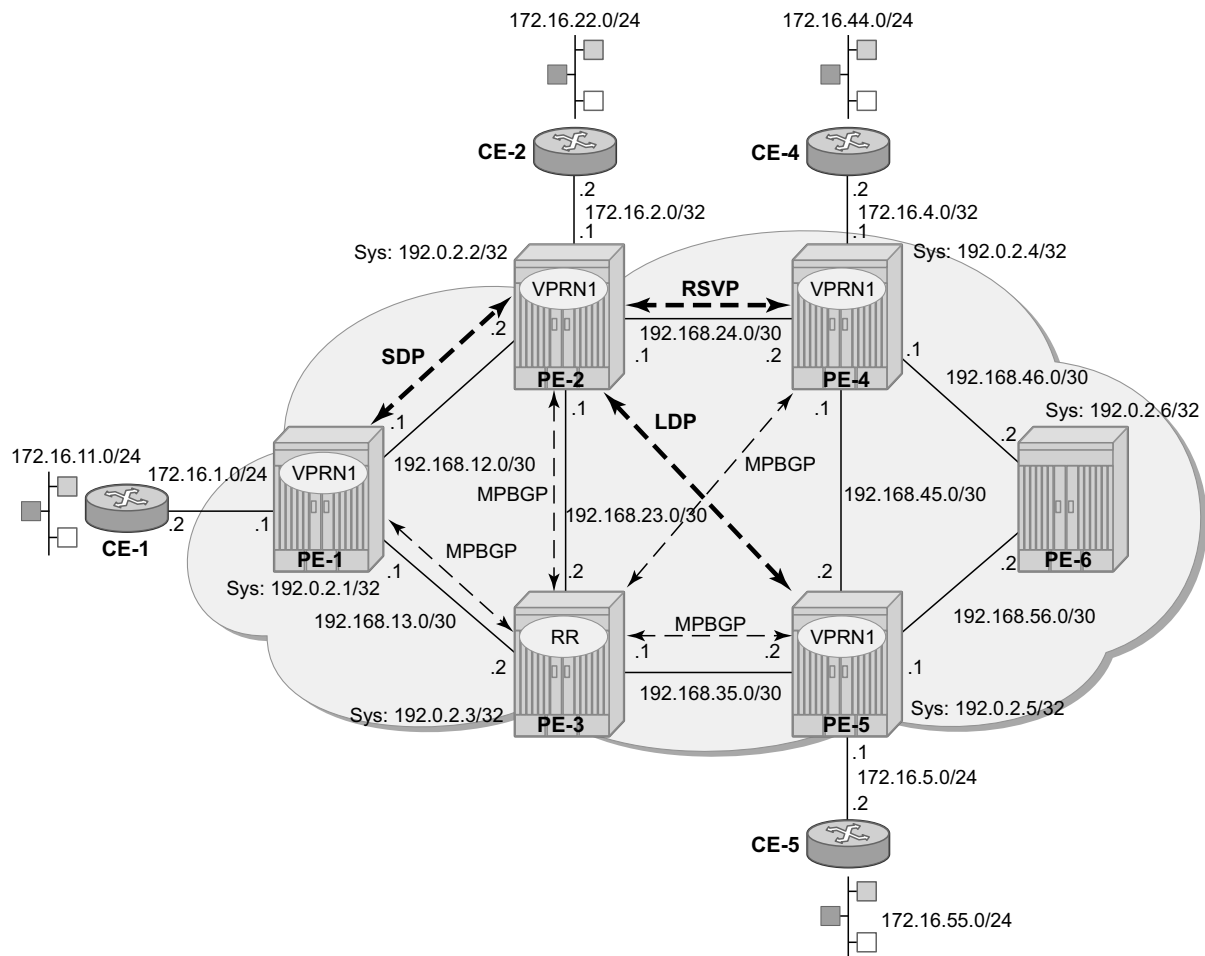
Parameter descriptions:

- **ldp** — Specifies LDP based LSPs should be used to resolve the BGP next-hop for VPRN routes in an associated VPRN instance.
- **gre** — Specifies GRE based tunnels to be used to resolve the BGP next-hop for VPRN routes in an associated VPRN instance. GRE is out of the scope regarding shortcuts, refer to SR OS documentation for further details.
- **rsvp** — Specifies RSVP-TE LSPs should be used to resolve the BGP next-hop for VPRN routes in an associated VPRN instance.
- **ldp rsvp** — Chooses an existing RSVP-TE LSP if available, otherwise use LDP.

- **gre ldp rsvp** — When there are multiple tunnels to the BGP next-hop address, the tunnel with the lowest tunnel-table preference value is selected (first RSVP, then LDP, then GRE).
- **sr-isis** — For segment routing with IS-IS (beyond the scope of this document)
- **sr-ospf** — For segment routing with OSPF (beyond the scope of this document)

In all cases, if an explicit spoke-sdp is specified in the VPRN, it is always preferred over automatically selected tunnels (even if the SDP is down, the route becomes inactive; there is no fallback to the automatic selection).

The network is configured according to the topology shown in Figure 6. Four PEs (PE-1, PE-2, PE-4 and PE-5) are connected forming a meshed IP-VPN (named VPRN 1), using a route reflector on PE-3 for MP-BGP peering. All PEs have LDP tunnels enabled so at a minimum all can establish LDP shortcut tunnels to the others. In order to have not only LDP but also RSVP-TE LSPs and static SDPs (using an RSVP LSP) in the network, a mix of tunneling methods is configured. For the sake of simplicity, a closer view on PE-2 only, provides all details about the shortcuts created by auto-bind-tunnel. PE-2 has a static SDP (RSVP based) with PE-1, an RSVP LSP with PE-4, and an LDP LSP with PE-5. Every PE has a CE connected, so each PE has an interface connected to the CE as well as a static route to a CE LAN (although redistribution routing policies are needed, they are not shown for simplicity).



OSSG627

Figure 86: Shortcuts Within a VRF Topology Network

On PE-2, the following output shows the configuration of VPRN1:

```
*A:PE-2# configure service
sdp 1 mpls create
far-end 192.0.2.1
lsp "LSP-PE-2-PE-1"
no shutdown
exit
vprn 1 customer 1 create
vrf-import "VPN1-import"
vrf-export "VPN1-export"
route-distinguisher 65002:1
auto-bind-tunnel
resolution-filter
gre
ldp
rsvp
```

```

    exit
    resolution filter
  exit
  interface "to-CE-2" create
    address 172.16.2.1/24
    sap 1/1/4:1 create
    exit
  exit
  static-route 172.16.22.0/24 next-hop 172.16.2.2
  spoke-sdp 1 create
  exit
  no shutdown
exit

```

As previously mentioned, regarding IP-VPN meshed connectivity, the configuration shows that there is a static SDP 1 (pointing to PE-1), and the rest of the configuration is just **auto-bind-tunnel**. On PE-2, the connectivity towards the other PEs in the network can be verified by checking VPRN 1:

```

*A:PE-2# show router 1 route-table
=====
Route Table (Service: 1)
=====
Dest Prefix[Flags]                                Type   Proto   Age           Pref
  Next Hop[Interface Name]                        Metric
-----
172.16.1.0/24                                     Remote BGP VPN   00h12m26s    170
    192.0.2.1 (tunneled)                          0
172.16.2.0/24                                     Local  Local    00h20m27s     0
    to-CE-2                                          0
172.16.4.0/24                                     Remote BGP VPN   00h00m27s    170
    192.0.2.4 (tunneled:RSVP:3)                    0
172.16.5.0/24                                     Remote BGP VPN   00h12m26s    170
    192.0.2.5 (tunneled)                          0
172.16.11.0/24                                    Remote BGP VPN   00h12m26s    170
    192.0.2.1 (tunneled)                          0
172.16.22.0/24                                    Remote Static    00h20m27s     5
    172.16.2.2                                      1
172.16.44.0/24                                    Remote BGP VPN   00h00m27s    170
    192.0.2.4 (tunneled:RSVP:3)                    0
172.16.55.0/24                                    Remote BGP VPN   00h12m26s    170
    192.0.2.5 (tunneled)                          0
-----
No. of Routes: 8

```

As can be seen, there are eight routes since every PE has two routes (one direct PE-CE interface and one static route), so six routes are received from other PEs via MP-BGP. The VPRN 1 routing table can be understood by looking at the tunnel table (active LSPs for remote system-ids):

```

*A:PE-2# show router tunnel-table
=====
IPv4 Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId  Pref    Nexthop      Metric
-----

```

192.0.2.1/32	sdp	MPLS	1	5	192.0.2.1	0
192.0.2.1/32	rsvp	MPLS	2	7	192.168.12.1	10
192.0.2.1/32	ldp	MPLS	65537	9	192.168.12.1	10
192.0.2.3/32	ldp	MPLS	65538	9	192.168.23.2	10
192.0.2.4/32	rsvp	MPLS	3	7	192.168.24.2	10
192.0.2.4/32	rsvp	MPLS	4	7	192.168.24.2	16777215
192.0.2.4/32	ldp	MPLS	65549	9	192.168.24.2	10
192.0.2.5/32	ldp	MPLS	65546	9	192.168.23.2	20
192.0.2.6/32	ldp	MPLS	65551	9	192.168.24.2	20

The tunnel-table shows one entry per LSP per remote PE. The following tunnel selection rules apply:

- SDP has the lowest (best) preference, followed by RSVP and then by LDP.
- If the preference is the same, the lowest metric is selected (ECMP is possible with LDP).

PE-2 has three possibilities to reach PE-1 (192.0.2.1): an SDP Tunnel ID 1 with preference 5, an RSVP Tunnel ID 1 with preference 7, and an LDP LSP with preference 9. As SDP Tunnel ID 1 has the lowest preference, it is the chosen option. PE-2 has three possibilities to reach PE-4 (192.0.2.4): an RSVP Tunnel ID 3 with preference 7 and metric 10, an RSVP Tunnel ID 4 with preference 7 and metric 16777215, and an LDP LSP with preference 9; hence RSVP Tunnel ID 3 is selected. PE-2 only has one option to reach PE-5 and PE-6 (192.0.2.5 and .6) using an LDP LSP.

As the VPRN 1 output does not show the details of the tunneling, displaying the FIB on router VPRN 1 provides more detailed information:

```
*A:PE-2# show router 1 fib 1
=====
FIB Display
=====
Prefix [Flags]                                     Protocol
NextHop
-----
172.16.1.0/24                                     BGP_VPN
    192.0.2.1 (VPRN Label:262137 Transport:SDP:1)
172.16.2.0/24                                     LOCAL
    172.16.2.0 (to-CE-2)
172.16.4.0/24                                     BGP_VPN
    192.0.2.4 (VPRN Label:262136 Transport:RSVP LSP:3)
172.16.5.0/24                                     BGP_VPN
    192.0.2.5 (VPRN Label:262136 Transport:LDP)
172.16.11.0/24                                    BGP_VPN
    192.0.2.1 (VPRN Label:262137 Transport:SDP:1)
172.16.22.0/24                                    STATIC
    172.16.2.2 (to-CE-2)
172.16.44.0/24                                    BGP_VPN
    192.0.2.4 (VPRN Label:262136 Transport:RSVP LSP:3)
172.16.55.0/24                                    BGP_VPN
    192.0.2.5 (VPRN Label:262136 Transport:LDP)
-----
Total Entries : 8
=====
```

The FIB shows the chosen transport tunnel, specifying SDP ID, RSVP Tunnel ID, and LDP, as well as service label information linked to the routes.

As static SDP tunnels are preferred over dynamic tunnels (RSVP or LDP auto-bind), when the static SDP 1 is shutdown with or the LSP goes down (there is no fallback to dynamic tunneling), the associated routes are removed:

```
*A:PE-2# configure service sdp 1 shutdown

*A:PE-2# show router 1 fib 1
=====
FIB Display
=====
Prefix [Flags]                                Protocol
NextHop
-----
172.16.2.0/24                                LOCAL
    172.16.2.0 (to-CE-2)
172.16.4.0/24                                BGP_VPN
    192.0.2.4 (VPRN Label:262136 Transport:RSVP LSP:3)
172.16.5.0/24                                BGP_VPN
    192.0.2.5 (VPRN Label:262136 Transport:LDP)
172.16.22.0/24                               STATIC
    172.16.2.2 (to-CE-2)
172.16.44.0/24                               BGP_VPN
    192.0.2.4 (VPRN Label:262136 Transport:RSVP LSP:3)
172.16.55.0/24                               BGP_VPN
    192.0.2.5 (VPRN Label:262136 Transport:LDP)
-----
Total Entries : 6
```

To avoid this fallback issue, the configuration is modified and the manual spoke-sdps are removed in the configuration on PE-1 and PE-2, the rest of the configuration remains the same. Now the connectivity between PE-1 and PE-2 is using an RSVP LSP, as shown in the PE-1 output below (RSVP LSP which was used by SDP 1 has disappeared):

```
*A:PE-1# configure service vprn 1 no spoke-sdp 1
*A:PE-2# configure service vprn 1 no spoke-sdp 1

*A:PE-1# show router 1 route-table
=====
Route Table (Service: 1)
=====
Dest Prefix[Flags]                                Type    Proto    Age          Pref
Next Hop[Interface Name]                                Metric
-----
172.16.1.0/24                                Local   Local    00h33m21s    0
    to-CE-1
172.16.2.0/24                                Remote  BGP VPN   00h00m24s    170
    192.0.2.2 (tunneled:RSVP:2)
172.16.4.0/24                                Remote  BGP VPN   00h13m29s    170
    192.0.2.4 (tunneled)
172.16.5.0/24                                Remote  BGP VPN   00h25m09s    170
    192.0.2.5 (tunneled)
172.16.11.0/24                               Remote  Static    00h33m21s    5
```

```

172.16.1.2
172.16.22.0/24 Remote BGP VPN 00h00m24s 170
192.0.2.2 (tunneled:RSVP:2) 0
172.16.44.0/24 Remote BGP VPN 00h13m29s 170
192.0.2.4 (tunneled) 0
172.16.55.0/24 Remote BGP VPN 00h25m09s 170
192.0.2.5 (tunneled) 0
-----
No. of Routes: 8

```

If RSVP is disabled, the connectivity falls back to LDP as the output shows:

```

*A:PE-1# configure router mpls shutdown

*A:PE-1# show router 1 fib 1
=====
FIB Display
=====
Prefix [Flags]                                Protocol
NextHop
-----
172.16.1.0/24                                LOCAL
172.16.1.0 (to-CE-1)
172.16.2.0/24                                BGP_VPN
192.0.2.2 (VPRN Label:262136 Transport:LDP)
172.16.4.0/24                                BGP_VPN
192.0.2.4 (VPRN Label:262136 Transport:LDP)
172.16.5.0/24                                BGP_VPN
192.0.2.5 (VPRN Label:262136 Transport:LDP)
172.16.11.0/24                               STATIC
172.16.1.2 (to-CE-1)
172.16.22.0/24                               BGP_VPN
192.0.2.2 (VPRN Label:262136 Transport:LDP)
172.16.44.0/24                               BGP_VPN
192.0.2.4 (VPRN Label:262136 Transport:LDP)
172.16.55.0/24                               BGP_VPN
192.0.2.5 (VPRN Label:262136 Transport:LDP)
-----
Total Entries : 8
-----

```

If LDP is disabled, the connectivity falls back to GRE as the output shows:

```

*A:PE-1# configure router ldp shutdown

*A:PE-1# show router 1 fib 1
=====
FIB Display
=====
Prefix [Flags]                                Protocol
NextHop
-----
172.16.1.0/24                                LOCAL
172.16.1.0 (to-CE-1)
172.16.2.0/24                                BGP_VPN
192.0.2.2 (VPRN Label:262136 Transport:GRE)

```

MPLS/GRE Shortcut for BGP NH Resolution within a VRF

172.16.4.0/24	BGP_VPN
192.0.2.4 (VPRN Label:262136 Transport:GRE)	
172.16.5.0/24	BGP_VPN
192.0.2.5 (VPRN Label:262136 Transport:GRE)	
172.16.11.0/24	STATIC
172.16.1.2 (to-CE-1)	
172.16.22.0/24	BGP_VPN
192.0.2.2 (VPRN Label:262136 Transport:GRE)	
172.16.44.0/24	BGP_VPN
192.0.2.4 (VPRN Label:262136 Transport:GRE)	
172.16.55.0/24	BGP_VPN
192.0.2.5 (VPRN Label:262136 Transport:GRE)	

Total Entries : 8

Conclusion

IGP shortcuts provide a variety of shortcuts in IP, MPLS and IP-VPN scenarios to customers who want to use new options for building routing topologies. Because IGP shortcuts are enabled on a per router basis, SPF computations are independent and irrelevant to other routers, so there is no need to enable shortcuts globally. This network example shows the configuration of IGP shortcuts together with the associated show outputs which can be used for verification and troubleshooting.

Inter-Area TE Point-to-Point LSPs

In This Chapter

This section describes inter-area TE point-to-point LSP configurations.

Topics in this section include:

- [Applicability on page 494](#)
- [Summary on page 495](#)
- [Overview on page 497](#)
- [Configuration on page 499](#)
- [Conclusion on page 520](#)

Applicability

Inter-Area Traffic Engineering (TE) point-to-point (P2P) LSPs are supported on all 7x50 platforms. This feature is supported on all IOM/IMM types. The configuration was tested on release 13.0.R1.

Summary

MPLS TE is implemented on a wide scale in current ISP networks to steer traffic across the backbone to facilitate efficient use of available bandwidth between the routers and to guarantee fast convergence in case a link or node fails.

Previously, the MPLS TE designs allowed for TE LSPs that are confined to only a single IGP area/level. This is due to the fact that the head-end has information in the TE database of only the local area (OSPF) or level (ISIS).

Inter-Area TE LSP Based On Explicit Route Expansion

To be able to support Inter Area MPLS traffic engineering, the design needs to be extended. Inter-Area TE LSP based on Explicit route Object (ERO) expansion allows for the head-end to calculate the ERO path within its own area/level and keep the remaining ABRs of other areas/levels as loose hops in the ERO path. On receiving a PATH message with a loose hop ERO, based on local configuration each ABR does a partial Constrained Shortest Path First (CSPF) calculation to the next ABR or full CSPF to reach the final destination.

Automatic selection of ABRs is supported, in this way the head-end node can work with an empty primary path. When the **to** field of an LSP definition is in a different area/level than the head-end node, CSPF will automatically compute the segment to the exit ABR router which advertised the prefix and which is currently the best path for resolving the prefix in Route Table Manager (RTM).

ABR Protection

Link and Node protection within the respective areas are supported through the TE capabilities of the IGP and RSVP in each area. To support ABR node protection, a bypass is required from the Point of Local Repair (PLR; node prior to ABR) to the Merge Point (MP; next-hop node to ABR). Two methods are possible: dynamic ABR protection and static ABR protection. Static ABR protection uses Manual Bypass Tunnels (MBTs), statically configured by the operator between PLR and MP.

For dynamic ABR protection, node ID propagation and signaling of an Exclude Route (XRO) object in RSVP PATH messages must both be supported.

Since the description of the RRO Node ID sub-object in RFC 4561 (*Definition of a Record Route Object (RRO) Node-Id Sub-Object*) is not clear about the format of the included node-address (S), interface-address (I) and label (L), the system is programmed to understand multiple formats: IL, SL, ISL, SIL, SLI, ILSL and SLIL. The system uses the SLIL (node-address, label, interface-address, label) format to include the node-ID itself.

XRO object inclusion (RFC 4874, *Exclude Routes - Extension to Resource ReserVation Protocol-Traffic Engineering*) in bypass RSVP PATH messages is required to exclude the protected ABR from the bypass path. The XRO object is filled in with ABRs system IP address.

Overview

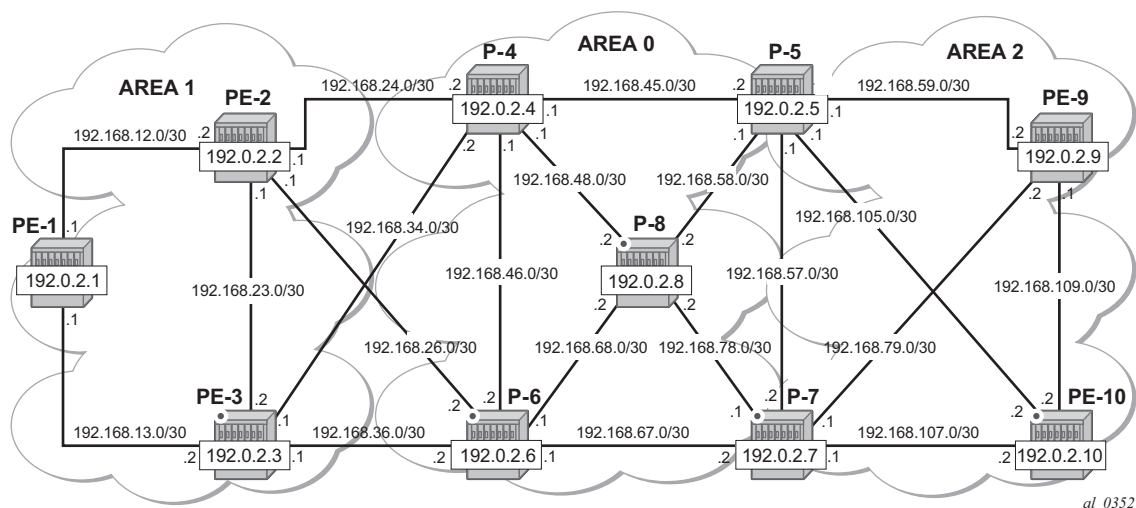


Figure 87: Inter-Area TE LSP Setup

The setup in this section contains 10 nodes in three areas. [Figure 87](#) shows the physical topology of the setup.

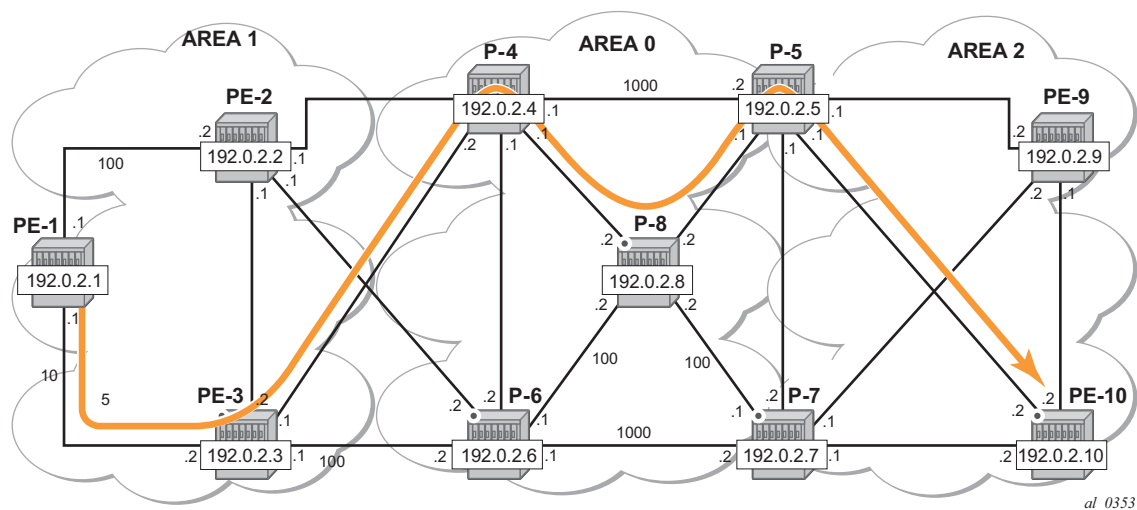


Figure 88: Inter-Area TE LSP Path

[Figure 88](#) shows the LSP path intended to be setup through the network. An empty MPLS path is used. At the head-end node (PE-1), the destination address (PE-10) is learned via ABR node P-4 and ABR node P-5.

Configuration

The assumption is made that following base configuration has been implemented on the PEs:

- Cards, MDAs and ports configured
- Interfaces configured
- IGP areas configured and converged
- Traffic Engineering configured for the IGP
- MPLS and RSVP configured on all links in the network

OSPF or ISIS can be configured as the IGP; OSPF is used here.

```
A:PE-1# show router ospf opaque-database
=====
OSPF Opaque Link State Database (Type : All)
=====
```

Type	Id	Link State Id	Adv Rtr Id	Age	Sequence	Cksum
Area	0.0.0.1	1.0.0.1	192.0.2.1	1503	0x80000002	0x9234
Area	0.0.0.1	1.0.0.3	192.0.2.1	1470	0x80000001	0x9b45
Area	0.0.0.1	1.0.0.4	192.0.2.1	1465	0x80000001	0xe9f2
Area	0.0.0.1	1.0.0.1	192.0.2.2	1498	0x80000002	0x962e
Area	0.0.0.1	1.0.0.3	192.0.2.2	1470	0x80000001	0xdce8
Area	0.0.0.1	1.0.0.4	192.0.2.2	1470	0x80000001	0x833b
Area	0.0.0.1	1.0.0.5	192.0.2.2	1471	0x80000001	0x637b
Area	0.0.0.1	1.0.0.6	192.0.2.2	1471	0x80000001	0x665f
Area	0.0.0.1	1.0.0.1	192.0.2.3	1504	0x80000002	0x9a28
Area	0.0.0.1	1.0.0.3	192.0.2.3	1472	0x80000001	0x6d43
Area	0.0.0.1	1.0.0.4	192.0.2.3	1481	0x80000001	0x1495
Area	0.0.0.1	1.0.0.5	192.0.2.3	1470	0x80000001	0x9f3c
Area	0.0.0.1	1.0.0.6	192.0.2.3	1472	0x80000001	0x4283
Area	0.0.0.1	1.0.0.1	192.0.2.4	1482	0x80000002	0x9e22
Area	0.0.0.1	1.0.0.6	192.0.2.4	1471	0x80000001	0x7e44
Area	0.0.0.1	1.0.0.7	192.0.2.4	1473	0x80000001	0x218b
Area	0.0.0.1	1.0.0.1	192.0.2.6	1584	0x80000002	0xa616
Area	0.0.0.1	1.0.0.6	192.0.2.6	1467	0x80000001	0xf6c5
Area	0.0.0.1	1.0.0.7	192.0.2.6	1482	0x80000001	0x990d

```
-----
No. of Opaque LSAs: 19
=====
A:PE-1#
```

The output above shows the opaque database of PE-1. The information is only about routers that are part of area 0.0.0.1. PE-1 cannot calculate an end-to-end CSPF path to node PE-10 since this would require TE topology information from area 0.0.0.0 and area 0.0.0.2.

Each node announces its router-ID and each attached link that is part of that area, hence the 19 opaque LSAs in area 0.0.0.1.

Note in [Figure 88](#) that the LSP should pass through node PE-3 and node P-8. In order to prefer a dynamic path from PE-1 to P-4 via PE-3 rather than through PE-2, it is necessary to configure on

Configuration

PE-1 a lower IGP metric on the interface to PE-3 (the default metric is derived from the interface speed; in this case the metric is 10 by default).

```
*A:PE-1>config>router>ospf# area 1 interface "int-PE-1-PE-3" metric 5
```

Similarly, in the core, the IGP metric between P-4 <=> P-5 and P-6 <=> P-7 is increased to force the LSP to pass through the core P-8 node.

```
*A:P-4>config>router>ospf# area 0 interface "int-P-4-P-5" metric 1000
*A:P-6>config>router>ospf# area 0 interface "int-P-6-P-7" metric 1000
```

Other metrics have also been manipulated as indicated on [Figure 88](#).

MPLS Path Configuration

Since automatic ABR selection is performed, an empty MPLS path is enough on the head-end node PE-1. Using an empty MPLS path will ease the provisioning process and brings consistency since this empty MPLS path can be used for both intra and inter-area/level type LSPs.

```
*A:PE-1# configure router mpls
*A:PE-1>config>router>mpls# path path-PE-1-PE-10 no shutdown
*A:PE-1>config>router>mpls#
```

MPLS LSP Configuration

Configure an LSP on PE-1 to PE-10 and include the previously created MPLS path as primary path. Enable CSPF and Fast Reroute (FRR) facility on the LSP.

```
*A:PE-1# configure router mpls
*A:PE-1>config>router>mpls# lsp LSP-PE-1-PE-10
*A:PE-1>config>router>mpls>lsp$ to 192.0.2.10
*A:PE-1>config>router>mpls>lsp$ cspf
*A:PE-1>config>router>mpls>lsp$ fast-reroute facility
*A:PE-1>config>router>mpls>lsp>frr$ exit
*A:PE-1>config>router>mpls>lsp$ primary "path-PE-1-PE-10" no shutdown
*A:PE-1>config>router>mpls>lsp$ no shutdown
*A:PE-1>config>router>mpls>lsp$
```

At this stage the LSP is in an operational Down state with a failure code of noCspfRouteToDestination.

In order to get around the intra-area CSPF confinement, enable the ERO-expansion feature on all possible ABR nodes.

```
*A:P-4# configure router mpls cspf-on-loose-hop
*A:P-6# configure router mpls cspf-on-loose-hop

*A:P-7# configure router mpls cspf-on-loose-hop
*A:P-5# configure router mpls cspf-on-loose-hop
```

Note that cspf-on-loose-hop is only required if FRR or TE parameters are configured on the LSP. If any of these parameters is configured on the LSP and one of the ABRs along the path is not configured with cspf-on-loose-hop, the LSP will stay operationally down with Failure Code: badNode and an indication of the interface address of the Failure Node.

```
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-10" path detail

<snipped>

From          : 192.0.2.1                               To          : 192.0.2.10
```

MPLS LSP Configuration

```
Adm State      : Up                               Oper State    : Down

<snipped>

Failure Code: badNode                             Failure Node: 192.168.34.2

<snipped>
```

The LSP path can also contain other strict and/or loose hops. Note however that cspf-on-loose-hop must be configured under MPLS whenever loose hops are configured in the MPLS path. This command is needed to trigger ERO expansion and is only required for inter-area LSPs on all possible ABR nodes and all nodes not belonging to the ‘ingress’ area (namely, the same area as the iLER) which have a ‘loose hop’ reference in the MPLS path. However, for simplicity it can be configured on all nodes without having a negative effect.

The following trace shows the ERO calculation on the head-end to the first ABR.

```
*A:PE-1# debug router rsvp packet path detail

2 2015/02/24 10:11:23.79 UTC MINOR: DEBUG #2001 Base RSVP
"RSVP: PATH Msg
Send PATH From:192.0.2.1, To:192.0.2.10
      TTL:255, Checksum:0xff83, Flags:0x0
Session   - EndPt:192.0.2.10, TunnId:1, ExtTunnId:192.0.2.1
SessAttr  - Name:LSP-PE-1-PE-10::path-PE-1-PE-10
            SetupPri:7, HoldPri:0, Flags:0x17
RSVPHop   - Ctype:1, Addr:192.168.13.1, LIH:3
TimeValue - RefreshPeriod:30
SendTempl - Sender:192.0.2.1, LspId:47620
SendTSpec - Ctype:QOS, CDR:0.000 bps, PBS:0.000 bps, PDR:infinity
            MPU:20, MTU:1564
LabelReq  - IfType:General, L3ProtID:2048
RRO       - IpAddr:192.168.13.1, Flags:0x0
ERO       - IPv4Prefix 192.168.13.2/32, Strict
            IPv4Prefix 192.168.34.2/32, Strict
            IPv4Prefix 192.0.2.10/32, Loose
FRRObj    - SetupPri:7, HoldPri:0, HopLimit:16, BW:0.000 bps, Flags:0x2
            ExcAny:0x0, IncAny:0x0, IncAll:0x0
"
```

On the P-4 ABR the ERO is expanded to include the nodes of area 0.0.0.0 of which P-4 is also part. The RRO contains all the hops the PATH message has passed so far.

```
*A:P-4# debug router rsvp packet path detail

14 2015/02/24 10:04:49.70 UTC MINOR: DEBUG #2001 Base RSVP
"RSVP: PATH Msg
Send PATH From:192.0.2.1, To:192.0.2.10
      TTL:253, Checksum:0x8dff, Flags:0x0
Session   - EndPt:192.0.2.10, TunnId:1, ExtTunnId:192.0.2.1
SessAttr  - Name:LSP-PE-1-PE-10::path-PE-1-PE-10
            SetupPri:7, HoldPri:0, Flags:0x17
RSVPHop   - Ctype:1, Addr:192.168.48.1, LIH:4
TimeValue - RefreshPeriod:30
```

```

SendTempl - Sender:192.0.2.1, LspId:47620
SendTSpec - Ctype:QOS, CDR:0.000 bps, PBS:0.000 bps, PDR:infinity
           MPU:20, MTU:1564
LabelReq  - IfType:General, L3ProtID:2048
RRO       - IpAddr:192.168.48.1, Flags:0x0
           IpAddr:192.168.34.1, Flags:0x0
           IpAddr:192.168.13.1, Flags:0x0
ERO       - IPv4Prefix 192.168.48.2/32, Strict
           IPv4Prefix 192.168.58.1/32, Strict
           IPv4Prefix 192.0.2.10/32, Loose
FRRObj    - SetupPri:7, HoldPri:0, HopLimit:16, BW:0.000 bps, Flags:0x2
           ExcAny:0x0, IncAny:0x0, IncAll:0x0
"

```

Finally, the P-5 ABR will expand the ERO to the final destination PE-10:

```

*A:P-5# debug router rsvp packet path detail

10 2015/02/24 10:04:19.74 UTC MINOR: DEBUG #2001 Base RSVP
"RSVP: PATH Msg
Send PATH From:192.0.2.1, To:192.0.2.10
           TTL:251, Checksum:0x365e, Flags:0x0
Session   - EndPt:192.0.2.10, TunnId:1, ExtTunnId:192.0.2.1
SessAttr  - Name:LSP-PE-1-PE-10::path-PE-1-PE-10
           SetupPri:7, HoldPri:0, Flags:0x17
RSVPHop   - Ctype:1, Addr:192.168.105.1, LIH:5
TimeValue - RefreshPeriod:30
SendTempl - Sender:192.0.2.1, LspId:47620
SendTSpec - Ctype:QOS, CDR:0.000 bps, PBS:0.000 bps, PDR:infinity
           MPU:20, MTU:1564
LabelReq  - IfType:General, L3ProtID:2048
RRO       - IpAddr:192.168.105.1, Flags:0x0
           IpAddr:192.168.58.2, Flags:0x0
           IpAddr:192.168.48.1, Flags:0x0
           IpAddr:192.168.34.1, Flags:0x0
           IpAddr:192.168.13.1, Flags:0x0
ERO       - IPv4Prefix 192.168.105.2/32, Strict
FRRObj    - SetupPri:7, HoldPri:0, HopLimit:16, BW:0.000 bps, Flags:0x2
           ExcAny:0x0, IncAny:0x0, IncAll:0x0
"

```

The MPLS LSP is now operational up and the LSP path can be shown in detail on the head-end, PE-1:

```

*A:PE-1# show router mpls lsp "LSP-PE-1-PE-10" path detail

=====
MPLS LSP LSP-PE-1-PE-10 Path (Detail)
=====
Legend :
  @ - Detour Available          # - Detour In Use
  b - Bandwidth Protected      n - Node Protected
  s - Soft Preemption
  S - Strict                    L - Loose
  A - ABR

```

MPLS LSP Configuration

```

=====
-----
LSP LSP-PE-1-PE-10 Path path-PE-1-PE-10
-----
LSP Name      : LSP-PE-1-PE-10                Path LSP ID : 47620
From          : 192.0.2.1                      To         : 192.0.2.10
Adm State     : Up                            Oper State  : Up
Path Name     : path-PE-1-PE-10              Path Type   : Primary
Path Admin    : Up                            Path Oper   : Up
OutInterface  : 1/1/2                        Out Label   : 131071
Path Up Time  : 0d 00:02:06                  Path Dn Time: 0d 00:00:00
Retry Limit   : 0                            Retry Timer  : 30 sec
RetryAttempt  : 0                            NextRetryIn : 0 sec

Adspec        : Disabled                    Oper Adspec  : Disabled
CSPF          : Enabled                    Oper CSPF    : Enabled
Least Fill    : Disabled                  Oper LeastF* : Disabled
FRR           : Enabled                    Oper FRR     : Enabled
FRR NodePro*  : Enabled                    Oper FRR NP  : Enabled
FR Hop Limit  : 16                         Oper FRHopL* : 16
FR Prop Adm*  : Disabled                    Oper FRProp* : Disabled
Prop Adm Grp  : Disabled                    Oper PropAG  : Disabled
Inter-area    : True

Neg MTU       : 1560                        Oper MTU     : 1560
Bandwidth     : No Reservation              Oper Bw      : 0 Mbps
Hop Limit     : 255                         Oper HopLim* : 255
Record Route  : Record                      Oper RecRou* : Record
Record Label  : Record                      Oper RecLab* : Record
SetupPrior*   : 7                           Oper SetupP* : 7
Hold Prior*   : 0                           Oper HoldPr* : 0
Class Type    : 0                           Oper CT      : 0
Backup CT     : None

MainCT Retry  : n/a
    Rem       :
MainCT Retry  : 0
    Limit     :
Include Grps  :
None
Exclude Grps :
None

Adaptive      : Enabled                    Oper Metric  : 15
Preference    : n/a
Path Trans    : 1
Failure Code  : noError
ExplicitHops  :
    No Hops Specified
Actual Hops   :
    192.168.13.1 (192.0.2.1) @ n
    -> 192.168.13.2 (192.0.2.3) @
    -> 192.168.34.2 (192.0.2.4) @ n
    -> 192.168.48.2 @
    -> 192.168.58.1 @
    -> 192.168.105.2
ComputedHops  :
    192.168.13.1(S)
    -> 192.168.13.2(S)
    -> 192.168.34.2(SA)

Record Label   : N/A
Record Label   : 131071
Record Label   : 131071
Record Label   : 131071
Record Label   : 131071
Record Label   : 131071

```

Inter-Area TE Point-to-Point LSPs

```
-> 192.0.2.10 (L)
ResigEligib*: False
LastResignal: n/a                                CSPF Metric : 15
=====
* indicates that the corresponding row element may have been truncated.
*A:PE-1#
```

ABR Node Protection

At this stage, the LSP is configured with facility FRR protection; link and node protection will be offered within each area. Dynamic ABR node protection requires the setup of a bypass tunnel from the PLR (node just upstream of the ABR) to the MP (node just downstream of the ABR). Two things are required for this:

- Firstly, the PLR node (part of area x) needs to know the system IP address of MP node (part of area y) to setup the bypass. For this reason, the node-ID of the MP is included in the RESV message so that the PLR can link the manual bypass tunnel to the primary path to protect the ABR.
- Secondly, the other ABR node receiving the RSVP bypass PATH message for the protected ABR needs to do an ERO expansion towards MP node. For this reason, the XRO object is included in the RSVP bypass PATH message, containing the node-ID of the protected ABR. As an example, a bypass PATH message is shown below on node PE-3.

The XRO object includes the system IP address of the protected ABR node (P-4) and the ERO object has MP node (P-8) as loose destination:

```
*A:PE-3# debug router rsvp packet path detail

3 2015/02/24 12:01:32.69 UTC MINOR: DEBUG #2001 Base RSVP
"RSVP: PATH Msg
Send PATH From:192.0.2.3, To:192.0.2.8
      TTL:17, Checksum:0xfddd, Flags:0x0
Session   - EndPt:192.0.2.8, TunnId:61442, ExtTunnId:192.0.2.3
SessAttr  - Name:bypass-node192.0.2.4-61442
            SetupPri:7, HoldPri:0, Flags:0x2
RSVPHop   - Ctype:1, Addr:192.168.36.1, LIH:3
TimeValue - RefreshPeriod:30
SendTempl - Sender:192.0.2.3, LspId:4
SendTSpec - Ctype:QOS, CDR:0.000 bps, PBS:0.000 bps, PDR:infinity
            MPU:20, MTU:1564
LabelReq  - IfType:General, L3ProtID:2048
RRO       - IpAddr:192.168.36.1, Flags:0x0
ERO       - IPv4Prefix 192.168.36.2/32, Strict
            IPv4Prefix 192.0.2.8/32, Loose
XRO       - IPv4Prefix: 192.0.2.4/32, Attribute: Node, LBit: Exclude
AdSpec    - General BreakBit:0, NumISHops:0, PathBwEstimate:0
            MinPathLatency:4294967295, CompPathMTU:1564
            Controlled BreakBit:0
"
```

Node-ID Inclusion in the RESV Message

P-8 will be the MP for the bypass of ABR P-4 and PE-10 will be the MP for the bypass of ABR P-5. So P-8 and PE-10 need to include their node-ID in the RESV message, inside the Record Route Object (RRO).

```
*A:P-8# configure router rsvp node-id-in-rro include
*A:PE-10# configure router rsvp node-id-in-rro include
```

The default is **node-id-in-rro exclude**. As an example, the RESV message received on PLR node (PE-3) is shown below. The RRO contains the MP node (P-8) information in SLIL format:

```
*A:PE-3# debug router rsvp packet resv detail

3 2015/02/24 14:01:25.69 UTC MINOR: DEBUG #2001 Base RSVP
"RSVP: RESV Msg
Send RESV From:192.168.13.2, To:192.168.13.1
      TTL:255, Checksum:0xc18, Flags:0x0
Session      - EndPt:192.0.2.10, TunnId:1, ExtTunnId:192.0.2.1
RSVPHop      - Ctype:1, Addr:192.168.13.2, LIH:3
TimeValue    - RefreshPeriod:30
Style        - SE
FlowSpec     - Ctype:QOS, CDR:0.000 bps, PBS:0.000 bps, PDR:infinity
              MPU:20, MTU:1560, RSpecRate:0, RSpecSlack:0
FilterSpec   - Sender:192.0.2.1, LspId:47626, Label:131071
RRO          - <snipped>

              SystemIp:192.0.2.8, Flags:0x29
              Label:131071, Flags:0x1
              InterfaceIp:192.168.48.2, Flags:0x9
              Label:131071, Flags:0x1

<snipped>
"
```

Bypass Configuration For ABR Protection

Since dynamic ABR protection is supported and used in this example, no explicit MBTs are configured to protect the ABRs. Each PLR first checks if an MBT tunnel exists between PLR and MP matching the constraints and protecting the ABR. If no MBT is available, the PLR will signal a bypass tunnel in a dynamic way towards MP node.

Figure 89 shows the two dynamic ABR node protections that are signaled for this LSP.

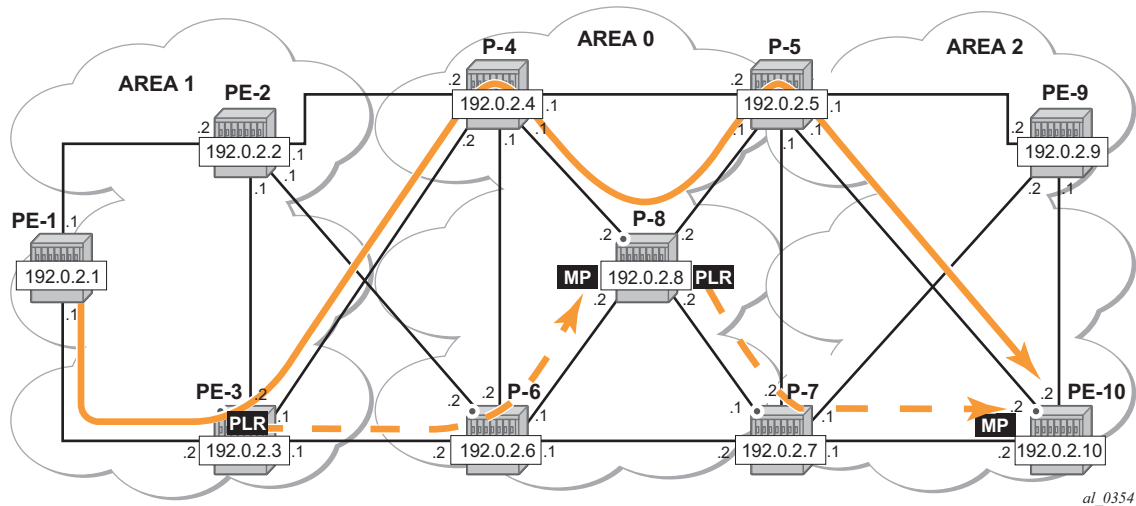


Figure 89: ABR Protection

Figure 90 shows the complete picture of all the FRR protections and indicates each node/link protection in the setup.

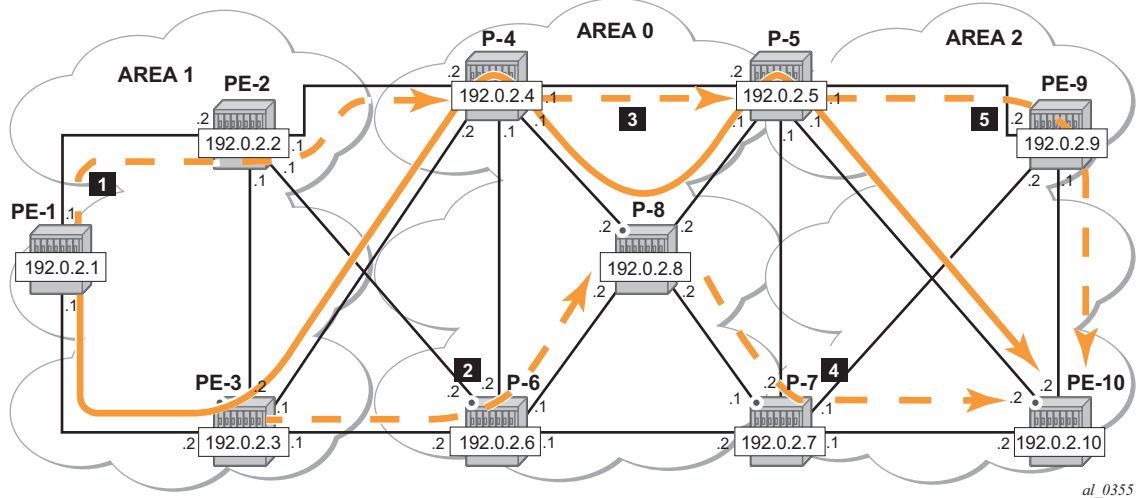


Figure 90: Protection of All Nodes/Links Along the LSP Path

This can be seen in the detailed show output of the LSP path:

```
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-10" path detail
```

<snipped>

```
-----
LSP LSP-PE-1-PE-10 Path path-PE-1-PE-10
-----
```

LSP Name	: LSP-PE-1-PE-10	Path LSP ID	: 47620
From	: 192.0.2.1	To	: 192.0.2.10
Adm State	: Up	Oper State	: Up
Path Name	: path-PE-1-PE-10	Path Type	: Primary
Path Admin	: Up	Path Oper	: Up

<snipped>

```
Inter-area : True
```

<snipped>

```
Actual Hops :
```

192.168.13.1 (192.0.2.1) @ n	Record Label	: N/A
-> 192.168.13.2 (192.0.2.3) @ n	Record Label	: 131070
-> 192.168.34.2 (192.0.2.4) @ n	Record Label	: 131070
-> 192.0.2.8 (192.0.2.8) @ n	Record Label	: 131069
-> 192.168.48.2 @ n	Record Label	: 131069
-> 192.0.2.5 (192.0.2.5) @	Record Label	: 131069
-> 192.168.58.1 @	Record Label	: 131069
-> 192.0.2.10 (192.0.2.10)	Record Label	: 131068
-> 192.168.105.2	Record Label	: 131068

<snipped>

Note that there are two entries for P-8, P-5 and PE-10 in the ‘Actual Hops’ section in the previous output: one for the interface IP address and one for the system IP address. This is a consequence of configuring **node-id-in-rro include** on P-8, P-5 and PE-10.

Note: The **node-id-in-rro include** command is not mandatory for this example on ABR node P-5 but to be future safe (for example, to cover cases where a new LSP is established in the network and P-5 acts as an MP node while the corresponding PLR node for that new LSP is in another area), this RSVP command can be enabled on all possible MP nodes in the network.

The details of the bypass tunnel can be shown with the following command:

```
*A:PE-3# show router mpls bypass-tunnel protected-lsp detail
=====
MPLS Bypass Tunnels (Detail)
=====
-----
bypass-node192.0.2.4-61442
-----
To                : 192.0.2.8                State           : Up
Out I/F           : 1/1/2                    Out Label        : 131071
Up Time           : 0d 00:01:54              Active Time      : n/a
Reserved BW       : 0 Kbps                   Protected LSP Count : 1
Type              : Dynamic                  Bypass Path Cost  : 100
Setup Priority    : 7                        Hold Priority     : 0
Class Type        : 0
Exclude Node      : 192.0.2.4                Inter-Area        : True
Computed Hops     :
    192.168.36.1(S)                        Egress Admin Groups : None
    -> 192.168.36.2(SA)                     Egress Admin Groups : None
    -> 192.0.2.8(L)                         Egress Admin Groups : None
Actual Hops       :
    192.168.36.1 (192.0.2.3)                Record Label       : N/A
    -> 192.168.36.2 (192.0.2.6)              Record Label       : 131071
    -> 192.0.2.8 (192.0.2.8)                 Record Label       : 131070
    -> 192.168.68.2                          Record Label       : 131070
Last Resignal     :
Attempted At      : n/a                     Resignal Reason    : n/a
Resignal Status   : n/a                     Reason             : n/a

Protected LSPs -
LSP Name          : LSP-PE-1-PE-10::path-PE-1-PE-10
From              : 192.0.2.1                To                : 192.0.2.10
Avoid Node/Hop    : 192.0.2.4                Downstream Label   : 131071
Bandwidth         : 0 Kbps
=====
*A:PE-3#
```

The LSP could be further protected with one or more additional secondary paths, pre-signaled or not, but this is outside the scope of this example.

When a link or node failure occurs along the LSP path, FRR protection kicks in and end-to-end path re-optimization is executed: a PATHERR message is forwarded to the head-end. Upon receiving the PATHERR message the head-end calculates a new path.

Admin Groups

To support admin-groups for inter-area LSPs, the ingress node (PE-1) must propagate the admin-groups within the Session Attribute object (SA) of the PATH message so that the ABRs along the path receive the Admin Group restrictions they have to take into account when further expanding the ERO in the PATH message.

In [Figure 91](#) the LSP path avoids the link between P-4 and P-8. This will be done by assigning admin group red to the link between P-4 and P-8 and then configuring the LSP to exclude the admin group red.

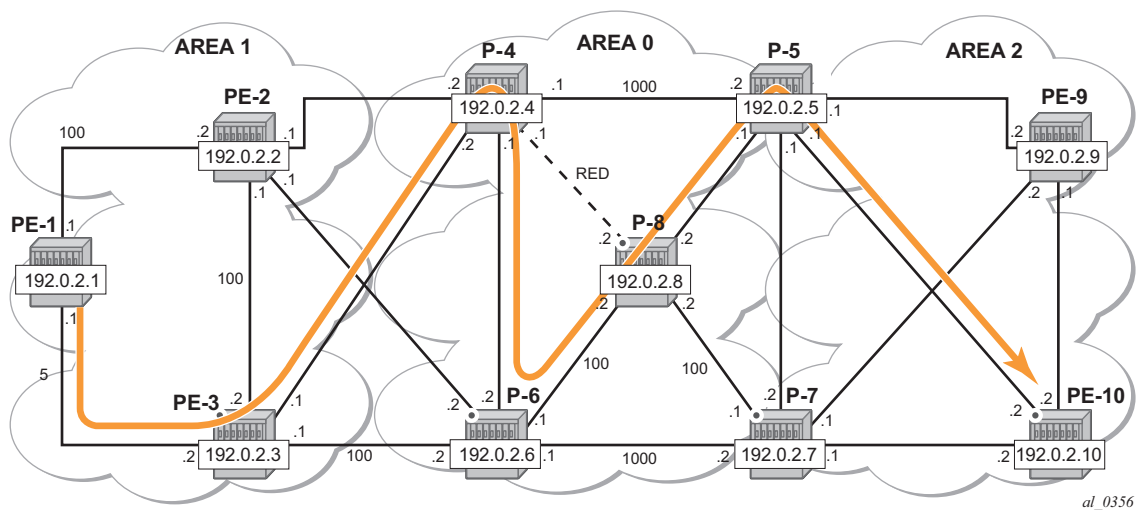


Figure 91: Admin Group Example

Admin Group Configuration

On P-4, configure admin group red and assign a group value to it (in the example value 11 is used, but this can be any value between 0 and 31). Assign admin group red to the link to P-8.

Note that this admin-group configuration is required on all nodes in this example.

```
*A:Px# configure router if-attribute
*A:Px>config>router>if-attr# admin-group red value 11
*A:Px>config>router>if-attr#

*A:P-4# configure router mpls
*A:P-4>config>router>mpls# interface "int-P-4-P-8" admin-group "red"
*A:P-4>config>router>mpls#
```

On PE-1, change the LSP configuration as follows:

```
*A:PE-1# configure router mpls
*A:PE-1>config>router>mpls# lsp "LSP-PE-1-PE-10"
*A:PE-1>config>router>mpls>lsp# exclude "red"
*A:PE-1>config>router>mpls>lsp# propagate-admin-group
*A:PE-1>config>router>mpls>lsp# exit
*A:PE-1>config>router>mpls# info
-----
      interface "system"
        no shutdown
      exit
      interface "int-PE-1-PE-2"
        no shutdown
      exit
      interface "int-PE-1-PE-3"
        no shutdown
      exit
      path "path-PE-1-PE-10"
        no shutdown
      exit
      lsp "LSP-PE-1-PE-10"
        to 192.0.2.10
        cspf
        exclude "red"
        propagate-admin-group
        fast-reroute facility
      exit
      primary "path-PE-1-PE-10"
      exit
      no shutdown
    exit
  no shutdown
-----
*A:PE-1>config>router>mpls#
```

Note the **propagate-admin-group** command is required to include the admin group properties in the SA object of the PATH message. Admin-group value is mapped to a 32-bitmap. In this

example, value 11 means that the 12th bit is set, which means in binary 100000000000 or hex 0x800.

```
*A:PE-1# debug router rsvp packet path detail

8 2015/02/24 10:18:17.79 UTC MINOR: DEBUG #2001 Base RSVP
"RSVP: PATH Msg
Send PATH From:192.0.2.1, To:192.0.2.10
      TTL:255, Checksum:0xf76d, Flags:0x0
Session   - EndPt:192.0.2.10, TunnId:1, ExtTunnId:192.0.2.1
SessAttr  - Name:LSP-PE-1-PE-10::path-PE-1-PE-10
           SetupPri:7, HoldPri:0, Flags:0x17
           Ctype:RA, ExcAny:0x800, IncAny:0x0, IncAll:0x0
RSVPHop   - Ctype:1, Addr:192.168.13.1, LIH:3
TimeValue - RefreshPeriod:30
SendTempl - Sender:192.0.2.1, LspId:47624
SendTSpec - Ctype:QOS, CDR:0.000 bps, PBS:0.000 bps, PDR:infinity
           MPU:20, MTU:1564
LabelReq  - IfType:General, L3ProtID:2048
RRO       - IpAddr:192.168.13.1, Flags:0x0
ERO       - IPv4Prefix 192.168.13.2/32, Strict
           IPv4Prefix 192.168.34.2/32, Strict
           IPv4Prefix 192.0.2.10/32, Loose
FRRObj    - SetupPri:7, HoldPri:0, HopLimit:16, BW:0.000 bps, Flags:0x2
           ExcAny:0x0, IncAny:0x0, IncAll:0x0
"
```

The two sets of output below show that when P-4 expands the ERO it now excludes the link to node P-8 for the path calculation and the path is setup through P-6, P-8 and P-5.

```
*A:P-4# debug router rsvp packet path detail

20 2015/02/24 10:11:43.69 UTC MINOR: DEBUG #2001 Base RSVP
"RSVP: PATH Msg
Send PATH From:192.0.2.1, To:192.0.2.10
      TTL:253, Checksum:0x6627, Flags:0x0
Session   - EndPt:192.0.2.10, TunnId:1, ExtTunnId:192.0.2.1
SessAttr  - Name:LSP-PE-1-PE-10::path-PE-1-PE-10
           SetupPri:7, HoldPri:0, Flags:0x17
           Ctype:RA, ExcAny:0x800, IncAny:0x0, IncAll:0x0
RSVPHop   - Ctype:1, Addr:192.168.46.1, LIH:3
TimeValue - RefreshPeriod:30
SendTempl - Sender:192.0.2.1, LspId:47624
SendTSpec - Ctype:QOS, CDR:0.000 bps, PBS:0.000 bps, PDR:infinity
           MPU:20, MTU:1564
LabelReq  - IfType:General, L3ProtID:2048
RRO       - IpAddr:192.168.46.1, Flags:0x0
           IpAddr:192.168.34.1, Flags:0x0
           IpAddr:192.168.13.1, Flags:0x0
ERO       - IPv4Prefix 192.168.46.2/32, Strict
           IPv4Prefix 192.168.68.2/32, Strict
           IPv4Prefix 192.168.58.1/32, Strict
           IPv4Prefix 192.0.2.10/32, Loose
```

Admin Groups

```
FRRObj      - SetupPri:7, HoldPri:0, HopLimit:16, BW:0.000 bps, Flags:0x2
              ExcAny:0x0, IncAny:0x0, IncAll:0x0
"

*A:PE-1# show router mpls lsp "LSP-PE-1-PE-10" path detail
=====
<snipped>
=====
-----
LSP LSP-PE-1-PE-10 Path path-PE-1-PE-10
-----
LSP Name      : LSP-PE-1-PE-10                Path LSP ID : 47624
From          : 192.0.2.1                      To          : 192.0.2.10
Adm State     : Up                            Oper State  : Up
Path Name     : path-PE-1-PE-10               Path Type   : Primary
Path Admin    : Up                            Path Oper   : Up

<snipped>

Actual Hops :
    192.168.13.1 (192.0.2.1) @ n                Record Label : N/A
-> 192.168.13.2 (192.0.2.3)                    Record Label : 131071
-> 192.168.34.2 (192.0.2.4)                    Record Label : 131071
-> 192.168.46.2                                Record Label : 131070
-> 192.0.2.8 (192.0.2.8)                      Record Label : 131071
-> 192.168.68.2                                Record Label : 131071
-> 192.0.2.5 (192.0.2.5)                      Record Label : 131071
-> 192.168.58.1                                Record Label : 131071
-> 192.0.2.10 (192.0.2.10)                   Record Label : 131071
-> 192.168.105.2                             Record Label : 131071
<snipped>
```

Shared Risk Link Groups (SRLG)

Shared Risk Link Groups are also supported in the context of inter-area TE LSPs. SRLGs refer to situations where links in a network share a common fiber (or a common physical attribute). If one link fails, other links in the group may fail as well. Links in the group have a shared risk.

The MPLS TE SRLG feature enhances backup tunnel path selection so that a backup tunnel avoids using links that are in the same SRLG.

Consider the setup in [Figure 92](#), where an inter-area LSP is setup from PE-1 to PE-10 and the path goes through P-8 because of a lower IGP metric. To protect against a node failure of P-8, P-4 (PLR) would normally setup an FRR backup directly to P-5 (MP), because of the lower IGP metric (P-4 to P-5:1000) compared to the IGP traffic via P-6 (P-4 to P-6 to P-7 to P-5:1020).

However, imagine that in this setup the P-4 <=> P-5 link and the P-4 <=> P-8 links are part of the same transmission bundle. In this case a cut of that fiber bundle will bring down both the primary and the backup path.

This can be avoided by configuring these two links in the same SRLG group and enabling `srlg-frr strict` on P-4. In that case the backup will be setup via P-6 as indicated by the dotted line in [Figure 92](#)

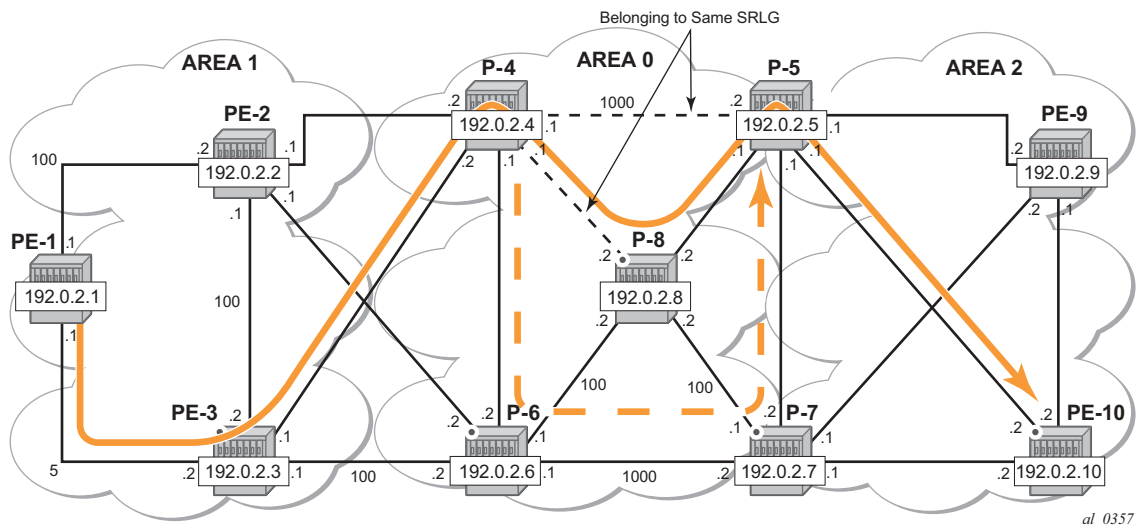


Figure 92: Share Risk Link Groups

SRLG Configuration

On P-4 configure an SRLG group and add the link to P-5 and the link to P-8 to this SRLG group and enable srlg-frr strict.

Note that the srlg-group configuration is required on all nodes that use srlg groups and on the ABR used by the inter-area TE LSP.

```
*A:Px# configure router if-attribute
*A:Px>config>router>if-attr# srlg-group bundle-red value 1
*A:Px>config>router>if-attr#

*A:P-4# configure router mpls
*A:P-4>config>router>mpls# interface "int-P-4-P-5" srlg-group "bundle-red"
*A:P-4>config>router>mpls# interface "int-P-4-P-8" srlg-group "bundle-red"
*A:P-4>config>router>mpls# srlg-frr strict
*A:P-4>config>router>mpls#
```

LSP Configuration

Remove the admin group restriction from the LSP.

```
*A:PE-1>config>router>mpls# lsp "LSP-PE-1-PE-10"
*A:PE-1>config>router>mpls>lsp# no exclude "red"
*A:PE-1>config>router>mpls>lsp# no propagate-admin-group
*A:PE-1>config>router>mpls>lsp# info
-----
                to 192.0.2.10
                cspf
                fast-reroute facility
                exit
                primary "path-PE-1-PE-10"
                exit
                no shutdown
-----
*A:PE-1>config>router>mpls>lsp#
```

Now check the LSP path on PE-1 and verify that FRR protection is in place.

```
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-10" path detail
=====
MPLS LSP LSP-PE-1-PE-10 Path (Detail)
=====
Legend :
  @ - Detour Available          # - Detour In Use
  b - Bandwidth Protected      n - Node Protected
  s - Soft Preemption
  S - Strict                    L - Loose
  A - ABR
=====
```



```
-----
LSP LSP-PE-1-PE-10 Path path-PE-1-PE-10
-----
```

```

LSP Name      : LSP-PE-1-PE-10          Path LSP ID : 47636
From          : 192.0.2.1                To          : 192.0.2.10
Adm State     : Up                      Oper State   : Up
Path Name     : path-PE-1-PE-10         Path Type    : Primary
Path Admin    : Up                      Path Oper    : Up
OutInterface  : 1/1/2                   Out Label    : 131071
Path Up Time  : 0d 00:00:25             Path Dn Time : 0d 00:00:00
Retry Limit   : 0                       Retry Timer  : 30 sec
RetryAttempt  : 0                       NextRetryIn  : 0 sec

```

```

Adspec        : Disabled                Oper Adspec  : Disabled
CSPF          : Enabled                 Oper CSPF    : Enabled
Least Fill    : Disabled                Oper LeastF* : Disabled
FRR           : Enabled                 Oper FRR     : Enabled
FRR NodePro*  : Enabled                 Oper FRR NP  : Enabled
FR Hop Limit  : 16                     Oper FRHopL* : 16
FR Prop Adm*  : Disabled                Oper FRProp* : Disabled
Prop Adm Grp  : Disabled                Oper PropAG  : Disabled
Inter-area    : True

```

```

Neg MTU       : 1560                    Oper MTU     : 1560
Bandwidth     : No Reservation           Oper Bw      : 0 Mbps
Hop Limit     : 255                     Oper HopLim* : 255
Record Route  : Record                  Oper RecRou* : Record
Record Label  : Record                  Oper RecLab* : Record
SetupPrior*   : 7                      Oper SetupP* : 7
Hold Prior*   : 0                      Oper HoldPr* : 0
Class Type    : 0                      Oper CT      : 0
Backup CT     : None
MainCT Retry  : n/a
Rem           :
MainCT Retry  : 0
Limit        :
Include Grps  :                          Oper InclGr* :
None                                                  None
Exclude Grps :                          Oper ExclGr* :
None                                                  None

```

```

Adaptive      : Enabled                 Oper Metric  : 15
Preference    : n/a
Path Trans    : 12                     CSPF Queries: 11
Failure Code  : noError                 Failure Node : n/a
ExplicitHops  :

```

```
    No Hops Specified
```

```

Actual Hops :
    192.168.13.1 (192.0.2.1) @ n
-> 192.168.13.2 (192.0.2.3) @ n
-> 192.168.34.2 (192.0.2.4) @ n
-> 192.0.2.8 (192.0.2.8)
-> 192.168.48.2
-> 192.0.2.5 (192.0.2.5)
-> 192.168.58.1
-> 192.0.2.10 (192.0.2.10)
-> 192.168.105.2
ComputedHops:
    192.168.13.1(S)

```

```

Record Label   : N/A
Record Label   : 131071
Record Label   : 131071
Record Label   : 131071
Record Label   : 131071
Record Label   : 131071
Record Label   : 131071
Record Label   : 131071
Record Label   : 131071

```

Shared Risk Link Groups (SRLG)

```
-> 192.168.13.2(S)
-> 192.168.34.2(SA)
-> 192.0.2.10(L)
ResigEligib*: False
LastResignal: n/a                                CSPF Metric : 15
=====
* indicates that the corresponding row element may have been truncated.
*A:PE-1#
```

On P-4 check the SRLG configuration and verify that the backup is setup via P-6 rather than via P-5.

```
*A:P-4# show router if-attribute srlg-group

=====
Interface Srlg Groups
=====
Group Name                Group Value    Penalty Weight
-----
bundle-red                1              0
-----
No. of Groups: 1
=====
*A:P-4#

*A:P-4# show router mpls interface

=====
MPLS Interfaces
=====
Interface                Port-id        Adm    Opr    TE-metric
-----
system                   system         Up     Up     None
  Admin Groups           None
  SRLG Groups            None
int-P-4-P-5              1/1/1         Up     Up     None
  Admin Groups           None
  SRLG Groups            bundle-red
int-P-4-P-6              1/1/3         Up     Up     None
  Admin Groups           None
  SRLG Groups            None
int-P-4-P-8              1/2/1         Up     Up     None
  Admin Groups           red
  SRLG Groups            bundle-red
int-P-4-PE-2             1/1/2         Up     Up     None
  Admin Groups           None
  SRLG Groups            None
int-P-4-PE-3             1/1/4         Up     Up     None
  Admin Groups           None
  SRLG Groups            None
-----
Interfaces : 6
=====
*A:P-4#
```

```
*A:P-4# show router mpls bypass-tunnel protected-lsp detail
```

```
=====
MPLS Bypass Tunnels (Detail)
=====
-----
bypass-node192.0.2.8-61444
-----
To          : 192.168.57.1      State          : Up
Out I/F     : 1/1/3            Out Label      : 131068
Up Time    : 0d 00:02:34       Active Time    : n/a
Reserved BW : 0 Kbps           Protected LSP Count : 2
Type       : Dynamic           Bypass Path Cost : 1020
Setup Priority : 7              Hold Priority    : 0
Class Type  : 0
Exclude Node : None            Inter-Area      : False
Computed Hops :
    192.168.46.1 (S)           Egress Admin Groups : None
    -> 192.168.46.2 (S)         Egress Admin Groups : None
    -> 192.168.67.2 (S)         Egress Admin Groups : None
    -> 192.168.57.1 (S)         Egress Admin Groups : None
Actual Hops :
    192.168.46.1 (192.0.2.4)   Record Label      : N/A
    -> 192.168.46.2 (192.0.2.6) Record Label      : 131068
    -> 192.168.67.2 (192.0.2.7) Record Label      : 131069
    -> 192.168.57.1 (192.0.2.5) Record Label      : 131066
Last Resignal :
Attempted At : n/a             Resignal Reason   : n/a
Resignal Status: n/a           Reason            : n/a

Protected LSPs -
LSP Name      : LSP-PE-1-PE-10::path-PE-1-PE-10
From          : 192.0.2.1      To                : 192.0.2.10
Avoid Node/Hop : 192.0.2.8     Downstream Label   : 131069
Bandwidth     : 0 Kbps

LSP Name      : LSP-PE-1-PE-10::path-PE-1-PE-10
From          : 192.0.2.1      To                : 192.0.2.10
Avoid Node/Hop : 192.0.2.8     Downstream Label   : 131071
Bandwidth     : 0 Kbps

=====
*A:P-4#
```

Conclusion

Inter-area TE P2P LSPs can be setup based on ERO expansion. With this feature the head-end does a partial CSPF calculation to its local ABR. On receiving a PATH message with a loose hop ERO this ABR does a partial or full CSPF calculation to the next ABR to reach the final destination.

FRR protection within the area is available. FRR node protection of the ABR is possible through an MBT on the PLR (node just upstream of the ABR) to the MP (node just downstream of the ABR) or through a dynamically signaled bypass tunnel on the PLR. Dynamic ABR node protection requires that the node-ID of the MP node is propagated in the RESV message and that an XRO object is included in the bypass PATH message which makes it possible for the ABR to calculate a path to an MP node.

TE features like BW, path prioritization, path pre-emption, graceful shutdown are supported, as well as propagation of the session attribute with affinity along the LSP path (admin groups) and SRLG.

LDP over RSVP Using OSPF as IGP

In This Chapter

This section provides information about Label Distribution Protocol (LDP) over Resource Reservation Protocol for Traffic Engineering (RSVP-TE), also called LDPoRSVP, that uses RSVP Label Switched Paths (LSPs) as a transport vehicle to carry the packets using LDP LSPs.

Topics in this section include:

- [Applicability on page 522](#)
- [Overview on page 523](#)
- [Configuration on page 525](#)
- [Additional Topics on page 552](#)
- [Conclusion on page 562](#)

Applicability

This section is applicable to all of the 7750 and 7450 series. Tested on release 13.0.R.1. No pre-requisites are required.

Overview

Introduction

Only user packets are tunneled over the RSVP LSPs, targeted LDP (T-LDP) control messages are still sent unlabeled using the IGP shortest path. Since LDP does not have traffic engineering (TE), it can now benefit from the RSVP-TE features.

The main advantage of LDPoRSVP is seen in large networks. A full mesh of intra-area RSVP LSPs between PE nodes (which in some cases is not scalable) is not needed anymore. While an LER may not have that many tunnels, any transit node will potentially have thousands of LSPs, and if each transit node also has to deal with detour tunnels or bypass tunnels, this number can make the LSR overly burdened.

LDPoRSVP can be configured in an intra-area domain and an inter-area domain. Any router in a given area can be a stitching point for LDP over RSVP. LDPoRSVP introduces a new tunnel type, tunnel-in-tunnel (in addition to the existing LDP tunnel type and RSVP tunnel type). If multiple tunnel types match the destination PE Forwarding Equivalence Class (FEC) lookup, LDP will prefer an LDP tunnel over an LDPoRSVP tunnel by default.

First, it is important to understand how LDP FEC resolution is working (with LDPoRSVP enabled). A more detailed explanation can be found later on in this section. The ingress LER receives an LDP label message including a FEC with prefix **P** and label **L** from peer **A** by a T-LDP session. LDP tries to resolve prefix **P** by performing a lookup in the Routing Table Manager (RTM). The result of this is a next-hop (NH) to the destination PE, either an intra-area PE (intra-area context) or an ABR (inter-area context). When the NH matches the targeted LDP peer, LDP performs a second lookup for that NH in the Tunnel Table Manager (TTM) which returns a user configured RSVP LSP with the best metric. If there are more than one configured RSVP LSPs with the best metric, LDP selects the first available RSVP LSP. If all user configured RSVP LSPs are down, no more action is taken. If the user did not configure any RSVP LSPs under the T-LDP context, the lookup in TTM will return the first available RSVP LSP which terminates on the ABR (inter-area) or intra-area PE with the lowest metric.

If the lookup in TTM results in no RSVP LSP, the system can fall back to link-level LDP (iLDP). In that way, it is possible that the NH is reachable using iLDP. Accordingly, the egress label will be installed on the ingress LER.

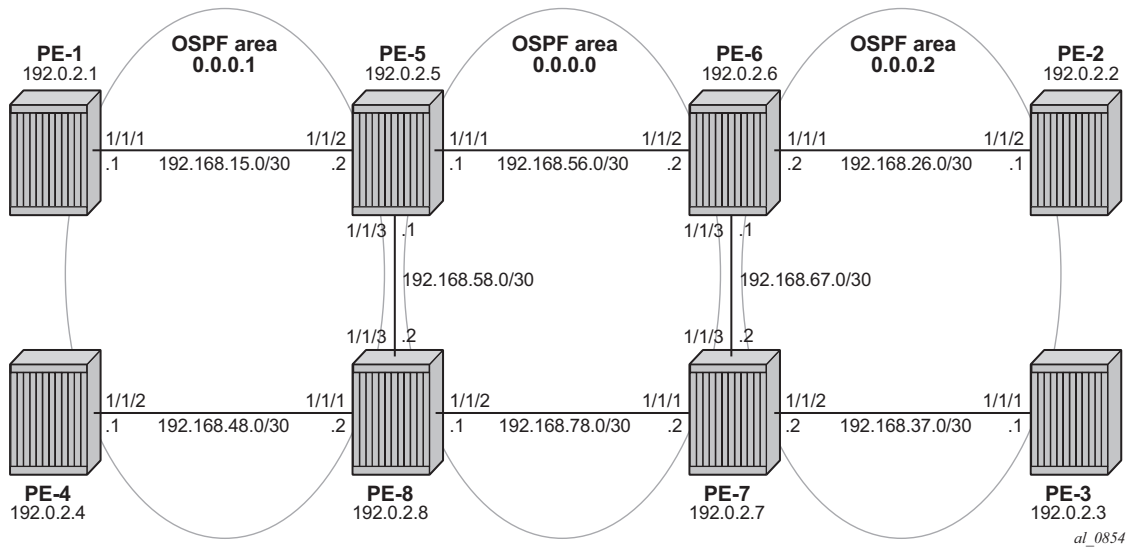


Figure 93: Initial Topology

OSPF area 0.0.0.1 and OSPF area 0.0.0.2 should be seen as two metro areas, connected to each other via a core area, represented by OSPF backbone area (area 0.0.0.0). Therefore, P-5, P-6, P-7 and P-8 are all acting as area border routers (ABRs). LDPoRSVP principles will be shown for intra-area PE communication (PE-1 <=> PE-4) and inter-area communication (PE-1 <=> PE-2).

Configuration

Step 1. Configuring the IP/MPLS network.

The system addresses and IP interface addresses are configured according to [Figure 93 on page 524](#). An interior gateway protocol (IGP) is needed to distribute routing information on all PEs. In this case, the IGP is OSPF using the backbone area (area 0.0.0.0) in the core and normal areas (area 0.0.0.1 and area 0.0.0.2) in the two metro regions, connected towards the backbone area via ABRs. A configuration example is shown below for PE-1 and P-5. A similar configuration can be derived for the other P and PE nodes.

```
A:PE-1# configure router ospf
      traffic-engineering
      area 0.0.0.1
        interface "system"
      exit
      interface "int-PE-1-P-5"
        interface-type point-to-point
      exit
    exit

A:P-5# configure router ospf
      traffic-engineering
      area 0.0.0.0
        interface "system"
      exit
      interface "int-P-5-P-6"
        interface-type point-to-point
      exit
      interface "int-P-5-P-8"
        interface-type point-to-point
      exit
    exit
  area 0.0.0.1
    interface "int-P-5-PE-1"
      interface-type point-to-point
    exit
  exit
```

Since fast reroute will be enabled on the RSVP LSPs in the core area, traffic engineering is needed on the IGP. By doing this, OSPF will generate opaque LSAs which are collected in a traffic engineering database (TED), separate from the traditional OSPF topology database. OSPF interfaces are set up as type point-to-point to improve convergence, no DR/BDR election process is performed.¹

On all nodes originating/terminating a T-LDP session, an explicit **ldp-over-rsvp** parameter must be configured to enable this OSPF instance for LDPoRSVP. In the example, this becomes.

1. Convergence is out of the scope of this document.

Configuration

```
A:PE-[1..4]# configure router ospf ldp-over-rsvp
A:P-[5..8]# configure router ospf ldp-over-rsvp
```

To verify that OSPF neighbors are up (state:full, **show router ospf neighbor**) is performed. To check if IP interface addresses/subnets are known on all PEs, **show router route-table** or **show router fib IOM-card-slot** will display the content of the forwarding information base (FIB).

```
*A:PE-1# show router ospf neighbor
=====
OSPFv2 (0) all neighbors
=====
Interface-Name          Rtr Id          State      Pri  RetxQ  TTL
Area-Id
-----
int-PE-1-P-5            192.0.2.5       Full       1    0      34
0.0.0.1
-----
No. of Neighbors: 1
=====
*A:PE-1#
```

```
*A:PE-1# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]          Type  Proto  Age      Pref
Next Hop[Interface Name]    Metric
-----
192.0.2.1/32                Local  Local  01h10m49s  0
system                      0
192.0.2.2/32                Remote  OSPF   00h01m01s  10
192.168.15.2                30
192.0.2.3/32                Remote  OSPF   00h00m48s  10
192.168.15.2                40
192.0.2.4/32                Remote  OSPF   00h00m33s  10
192.168.15.2                30
192.0.2.5/32                Remote  OSPF   00h01m19s  10
192.168.15.2                10
192.0.2.6/32                Remote  OSPF   00h01m13s  10
192.168.15.2                20
192.0.2.7/32                Remote  OSPF   00h00m48s  10
192.168.15.2                30
192.0.2.8/32                Remote  OSPF   00h00m33s  10
192.168.15.2                20
192.168.15.0/30             Local  Local  01h10m37s  0
int-PE-1-P-5                0
192.168.26.0/30             Remote  OSPF   00h01m07s  10
192.168.15.2                30
192.168.37.0/30             Remote  OSPF   00h00m48s  10
192.168.15.2                40
192.168.48.0/30             Remote  OSPF   00h00m33s  10
192.168.15.2                30
192.168.56.0/30             Remote  OSPF   00h01m19s  10
192.168.15.2                20
192.168.58.0/30             Remote  OSPF   00h01m19s  10
192.168.15.2                20
```

LDP over RSVP Using OSPF as IGP

```

192.168.67.0/30                               Remote  OSPF      00h01m07s  10
        192.168.15.2                           30
192.168.78.0/30                               Remote  OSPF      00h00m39s  10
        192.168.15.2                           30
-----
No. of Routes: 16
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====
*A:PE-1#

*A:PE-1# show router fib 1
=====
FIB Display
=====
Prefix [Flags]                                Protocol
  NextHop
-----
192.0.2.1/32                                LOCAL
    192.0.2.1 (system)
192.0.2.2/32                                OSPF
    192.168.15.2 (int-PE-1-P-5)
192.0.2.3/32                                OSPF
    192.168.15.2 (int-PE-1-P-5)
192.0.2.4/32                                OSPF
    192.168.15.2 (int-PE-1-P-5)
192.0.2.5/32                                OSPF
    192.168.15.2 (int-PE-1-P-5)
192.0.2.6/32                                OSPF
    192.168.15.2 (int-PE-1-P-5)
192.0.2.7/32                                OSPF
    192.168.15.2 (int-PE-1-P-5)
192.0.2.8/32                                OSPF
    192.168.15.2 (int-PE-1-P-5)
192.168.15.0/30                             LOCAL
    192.168.15.0 (int-PE-1-P-5)
192.168.26.0/30                             OSPF
    192.168.15.2 (int-PE-1-P-5)
192.168.37.0/30                             OSPF
    192.168.15.2 (int-PE-1-P-5)
192.168.48.0/30                             OSPF
    192.168.15.2 (int-PE-1-P-5)
192.168.56.0/30                             OSPF
    192.168.15.2 (int-PE-1-P-5)
192.168.58.0/30                             OSPF
    192.168.15.2 (int-PE-1-P-5)
192.168.67.0/30                             OSPF
    192.168.15.2 (int-PE-1-P-5)
192.168.78.0/30                             OSPF
    192.168.15.2 (int-PE-1-P-5)
-----
Total Entries : 16
=====
*A:PE-1#

```

Configuration

The next step in the process of setting up the IP/MPLS network, is enabling the IP interfaces in the MPLS and RSVP context on all involved nodes (PE and P nodes). By default, the system interface is put automatically within the MPLS/RSVP context. When an interface is put in the MPLS context, the system also copies it into the RSVP context. Explicit enabling of MPLS and RSVP context is done by the **no shutdown** command. The following output displays the MPLS/RSVP configuration for PE-1.

```
A:PE-1# configure router mpls no shutdown
```

```
A:PE-1# configure router rsvp no shutdown
```

```
A:PE-1# configure router mpls
      interface "system"
      exit
      interface "int-PE-1-P-5"
      exit
      no shutdown
```

```
A:PE-1# configure router rsvp
      interface "system"
      exit
      interface "int-PE-1-P-5"
      exit
      no shutdown
```

Step 2. Configure the RSVP LSPs. In both metro areas RSVP LSPs are set up from all PEs towards the ABRs, no intra-area PE-PE RSVP LSPs are needed. In the core/backbone, a full RSVP LSP mesh is required. To simplify the RSVP LSP configuration, no fast reroute is enabled on the RSVP LSPs in the metro areas, only in the backbone area. All RSVP paths are set up as **strict**. As an example, the configuration commands for PE-1 and P-5 node will look like the following output.

```
*A:PE-1# configure router mpls
      interface "system"
        no shutdown
      exit
      interface "int-PE-1-P-5"
        no shutdown
      exit
      path "path-PE-1-P-5"
        hop 1 192.168.15.2 strict
        no shutdown
      exit
      path "path-PE-1-P-5-P-8"
        hop 10 192.168.15.2 strict
        hop 20 192.168.58.2 strict
        no shutdown
      exit
      lsp "LSP-PE-1-P-5"
        to 192.0.2.5
        primary "path-PE-1-P-5"
        exit
        no shutdown
      exit
      lsp "LSP-PE-1-P-8"
        to 192.0.2.8
        primary "path-PE-1-P-5-P-8"
        exit
        no shutdown
      exit
      no shutdown
```

```
*A:P-5# configure router mpls
      interface "system"
        no shutdown
      exit
      interface "int-P-5-P-6"
        no shutdown
      exit
      interface "int-P-5-P-8"
        no shutdown
      exit
      interface "int-P-5-PE-1"
        no shutdown
      exit
      path "path-P-5-P-6"
        hop 10 192.168.56.2 strict
        no shutdown
      exit
      path "path-P-5-P-8"
        hop 10 192.168.58.2 strict
        no shutdown
```

Configuration

```
exit
path "path-P-5-P-6-P-7"
    hop 10 192.168.56.2 strict
    hop 20 192.168.67.2 strict
    no shutdown
exit
path "path-P-5-PE-1"
    hop 10 192.168.15.1 strict
    no shutdown
exit
path "path-P-5-P-8-PE-4"
    hop 10 192.168.58.2 strict
    hop 20 192.168.48.1 strict
    no shutdown
exit
lsp "LSP-P-5-PE-1"
    to 192.0.2.1
    primary "path-P-5-PE-1"
    exit
    no shutdown
exit
lsp "LSP-P-5-PE-4"
    to 192.0.2.4
    primary "path-P-5-P-8-PE-4"
    exit
    no shutdown
exit
lsp "LSP-P-5-P-6"
    to 192.0.2.6
    cspf
    fast-reroute facility
    exit
    primary "path-P-5-P-6"
    exit
    no shutdown
exit
lsp "LSP-P-5-P-7"
    to 192.0.2.7
    cspf
    fast-reroute facility
    exit
    primary "path-P-5-P-6-P-7"
    exit
    no shutdown
exit
lsp "LSP-P-5-P-8"
    to 192.0.2.8
    cspf
    fast-reroute facility
    exit
    primary "path-P-5-P-8"
    exit
    no shutdown
exit
no shutdown
```

To display the state of RSVP LSPs, several show commands can be used. A command to show the TTM is **show router tunnel-table** with parameter **rsvp** to reference to RSVP LSP signaling protocol. By default, an RSVP LSP has preference **7**.

```
*A:PE-1# show router mpls lsp
=====
MPLS LSPs (Originating)
=====
```

LSP Name	To	Tun Id	Fastfail Config	Adm	Opr
LSP-PE-1-P-5	192.0.2.5	1	No	Up	Up
LSP-PE-1-P-8	192.0.2.8	2	No	Up	Up

```
-----
LSPs : 2
=====
*A:PE-1#
*A:PE-1# show router tunnel-table
=====
Tunnel Table (Router: Base)
=====
```

Destination	Owner	Encap	TunnelId	Pref	Nexthop	Metric
192.0.2.5/32	rsvp	MPLS	1	7	192.168.15.2	16777215
192.0.2.8/32	rsvp	MPLS	2	7	192.168.15.2	16777215

```
-----
Flags: B = BGP backup route available
       E = inactive best-external BGP route
=====
*A:PE-1#
*A:P-5# show router mpls lsp
=====
MPLS LSPs (Originating)
=====
```

LSP Name	To	Tun Id	Fastfail Config	Adm	Opr
LSP-P-5-PE-1	192.0.2.1	1	No	Up	Up
LSP-P-5-PE-4	192.0.2.4	2	No	Up	Up
LSP-P-5-P-6	192.0.2.6	3	Yes	Up	Up
LSP-P-5-P-7	192.0.2.7	4	Yes	Up	Up
LSP-P-5-P-8	192.0.2.8	5	Yes	Up	Up

```
-----
LSPs : 5
=====
*A:P-5#
*A:P-5# show router tunnel-table
=====
Tunnel Table (Router: Base)
=====
```

Destination	Owner	Encap	TunnelId	Pref	Nexthop	Metric
192.0.2.1/32	rsvp	MPLS	1	7	192.168.15.1	16777215
192.0.2.4/32	rsvp	MPLS	2	7	192.168.58.2	16777215
192.0.2.6/32	rsvp	MPLS	3	7	192.168.56.2	10
192.0.2.7/32	rsvp	MPLS	4	7	192.168.56.2	20
192.0.2.8/32	rsvp	MPLS	5	7	192.168.58.2	10

Configuration

```
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
*A:P-5#
```

By default, the metric for strict LSPs configured without constrained shortest path first (CSPF) (RSVP LSPs in metro areas) is infinite (value = 16777215). The LSP metric for CSPF LSPs (RSVP LSPs in the core area) follows the IGP cost. LSP metrics can be explicitly set on the LSP level, see also in the [Additional Topics on page 552](#) section.

```
*A:PE-1# configure router mpls lsp "LSP-PE-1-P-5" metric
- metric <metric>
- no metric

<metric>                : <0..16777215>
```

Note that whenever an RSVP LSP comes up, it is, by default, eligible for LDPoRSVP, meaning that RSVP will signal to the relevant IGP (OSPF in this case) that the LSP should be included in the IGP/SPF run. The destination of the LSP (192.0.2.5) will be considered as a potential endpoint in the FEC resolution. With the **info detail** command, all default settings of a context are shown.

```
A:PE-1# configure router mpls lsp "LSP-PE-1-P-5"
A:PE-1>config>router>mpls>lsp# info detail
-----
to 192.0.2.5

<snipped>

      ldp-over-rsvp include

<snipped>

*A:PE-1# show router mpls lsp "LSP-PE-1-P-5" detail
=====
MPLS LSPs (Originating) (Detail)
=====
Type : Originating
-----
LSP Name      : LSP-PE-1-P-5
LSP Type      : RegularLsp
From          : 192.0.2.1
Adm State     : Up
LSP Tunnel ID : 1
To            : 192.0.2.5
Oper State    : Up

<snipped>

LdpOverRsvp  : Enabled
VprnAutoBind : Enabled

<snipped>

Primary(a)   : path-PE-1-P-5
Bandwidth    : 0 Mbps
Up Time      : 0d 00:04:55
=====
* indicates that the corresponding row element may have been truncated.
```



```
*A:PE-1#
```

To make an RSVP LSP ineligible for LDPoRSVP, use the **exclude** command.

```
A:PE-1# configure router mpls lsp <LSP-name> ldp-over-rsvp exclude
```

Step 3. Create T-LDP sessions according to RSVP LSPs. It is a must that when configuring an RSVP LSP eligible for LDPoRSVP, also a T-LDP session is initiated. This must be done on all PE and P nodes.

```
*A:PE-1# configure router ldp
      targeted-session
        peer 192.0.2.5
        exit
        peer 192.0.2.8
        exit
      exit
*A:PE-1#

*A:PE-1# show router ldp session
=====
LDP IPv4 Sessions
=====
Peer LDP Id          Adj Type  State          Msg Sent  Msg Recv  Up Time
-----
192.0.2.5:0          Targeted  Established    15        16        0d 00:01:04
192.0.2.8:0          Targeted  Established     5         7         0d 00:00:19
-----
No. of IPv4 Sessions: 2
=====

=====
LDP IPv6 Sessions
=====
Peer LDP Id
Adj Type            State          Msg Sent    Msg Recv    Up Time
-----
No Matching Entries Found
=====
*A:PE-1#
```

Step 4. Enable LDPoRSVP. This is done using the **tunneling** keyword inside the T-LDP session context. Configuration is needed on all PE and ABR nodes.

```
*A:PE-1# configure router ldp
      targeted-session
        peer 192.0.2.5
          tunneling
        exit
      exit
      peer 192.0.2.8
        tunneling
      exit
    exit
  exit
*A:PE-1>config>router>ldp#
```

As a result of the **tunneling** command, LDPoRSVP process (FEC resolving) is initiated. As already stated in the introduction, FEC resolution is a three-step process. First run an SPF calculation to the destination, then select an endpoint(s) close to that destination followed by a tunnel(s) to that endpoint. The next two steps go more into detail on this FEC resolution process. Step 5 will handle inter-area FEC resolving and Step 6 will handle intra-area FEC resolving.

Step 5. Inter-area FEC resolving (ingress LER is PE-1, egress LER is PE-2)

Step 5.1 Verification endpoint nodes and associated RSVP tunnels.

The first thing to do in the inter-area FEC resolving process is PE-1 performs an SPF calculation towards PE-2 with the purpose to search for an eligible endpoint, as close as possible to PE-2. An endpoint is eligible when a T-LDP session exists between PE-1 and the endpoint node, tunneling is configured on the endpoint node, PE-1 received a label for the destination FEC from the endpoint and an RSVP LSP exists between PE-1 and endpoint node that can be used for LDPoRSVP.

Endpoint node in OSPF area 1 can be either P-5 or P-8 (only those nodes have a T-LDP session towards PE-1). With **show router ldp bindings active prefixes prefix 192.0.2.2/32**, it can be concluded that P-5 will be the endpoint node.

```
*A:PE-1# show router ldp bindings active prefixes prefix 192.0.2.2/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
(IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
        (S) - Static (M) - Multi-homed Secondary Support
        (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
```

Prefix	Op	IngLbl	EgrLbl
EgrNextHop	EgrIf/LspId		
192.0.2.2/32	Push	--	131052
192.0.2.5	LspId 1		
192.0.2.2/32	Swap	131065	131052
192.0.2.5	LspId 1		

```
-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:PE-1#
*A:PE-1# show router mpls lsp
=====
MPLS LSPs (Originating)
=====
```

LSP Name	To	Tun Id	Fastfail Config	Adm	Opr
LSP-PE-1-P-5	192.0.2.5	1	No	Up	Up
LSP-PE-1-P-8	192.0.2.8	2	No	Up	Up

```
-----
LSPs : 2
=====
*A:PE-1#
*A:PE-1# show router tunnel-table
=====
Tunnel Table (Router: Base)
=====
```

```

Destination      Owner      Encap TunnelId Pref      Nexthop      Metric
-----
<snipped>

192.0.2.5/32      rsvp       MPLS  1          7          192.168.15.2  16777215

<snipped>
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
*A:PE-1#

```

Endpoint node in OSPF area 0 can be either P-6, P-7 or P-8 (only those nodes have a T-LDP session towards P-5). With **show router ldp bindings active prefixes prefix 192.0.2.2/32**, it can be concluded that P-6 will be the endpoint node.

```

*A:P-5# show router ldp bindings active prefixes prefix 192.0.2.2/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.5:0)
              (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use,  N - Label Not In Use, W - Label Withdrawn
      WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
      (S) - Static           (M) - Multi-homed Secondary Support
      (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                               Op           IngLbl      EgrLbl
EgrNextHop                           EgrIf/LspId
-----
192.0.2.2/32                         Push          --          131054
192.0.2.6                           LspId 3
192.0.2.2/32                         Swap          131052      131054
192.0.2.6                           LspId 3
-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:P-5#
*A:P-5# show router mpls lsp
=====
MPLS LSPs (Originating)
=====
LSP Name                               To           Tun      Fastfail  Adm  Opr
                                Id           Id       Config
-----
LSP-P-5-PE-1                         192.0.2.1    1        No        Up   Up
LSP-P-5-PE-4                         192.0.2.4    2        No        Up   Up

```

Configuration

```
LSP-P-5-P-6                192.0.2.6        3        Yes        Up        Up
LSP-P-5-P-7                192.0.2.7        4        Yes        Up        Up
LSP-P-5-P-8                192.0.2.8        5        Yes        Up        Up
-----
LSPs : 5
=====
*A:P-5#
*A:P-5# show router tunnel-table
=====
Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId  Pref    Nexthop      Metric
-----
<snipped>

192.0.2.6/32      rsvp      MPLS    3          7        192.168.56.2  10

<snipped>

-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
*A:P-5#
```

On node P-6, the same commands can be repeated for the final destination node (PE-2). Also there, an RSVP LSP towards PE-2 will be used as transport tunnel for user packets.

```
*A:P-6# show router ldp bindings active prefixes prefix 192.0.2.2/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.6:0)
              (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use,  N - Label Not In Use, W - Label Withdrawn
      WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
      (S) - Static          (M) - Multi-homed Secondary Support
      (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix          Op          IngLbl    EgrLbl
EgrNextHop      EgrIf/LspId
-----
192.0.2.2/32    Push        --        131069
192.0.2.2       LspId 4
192.0.2.2/32    Swap        131054    131069
192.0.2.2       LspId 4

-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:P-6#
*A:P-6# show router mpls lsp
```

```

=====
MPLS LSPs (Originating)
=====
LSP Name                               To           Tun      Fastfail  Adm  Opr
                                Id           Config
-----
LSP-P-6-P-5                          192.0.2.5    1        Yes       Up   Up
LSP-P-6-P-7                          192.0.2.7    2        Yes       Up   Up
LSP-P-6-P-8                          192.0.2.8    3        Yes       Up   Up
LSP-P-6-PE-2                         192.0.2.2    4        No        Up   Up
LSP-P-6-PE-3                         192.0.2.3    5        No        Up   Up
-----
LSPs : 5
=====
A:P-6#
*A:P-6# show router tunnel-table
=====
Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId  Pref    Nexthop      Metric
-----
<snipped>

192.0.2.2/32      rsvp      MPLS   4          7        192.168.26.1  16777215

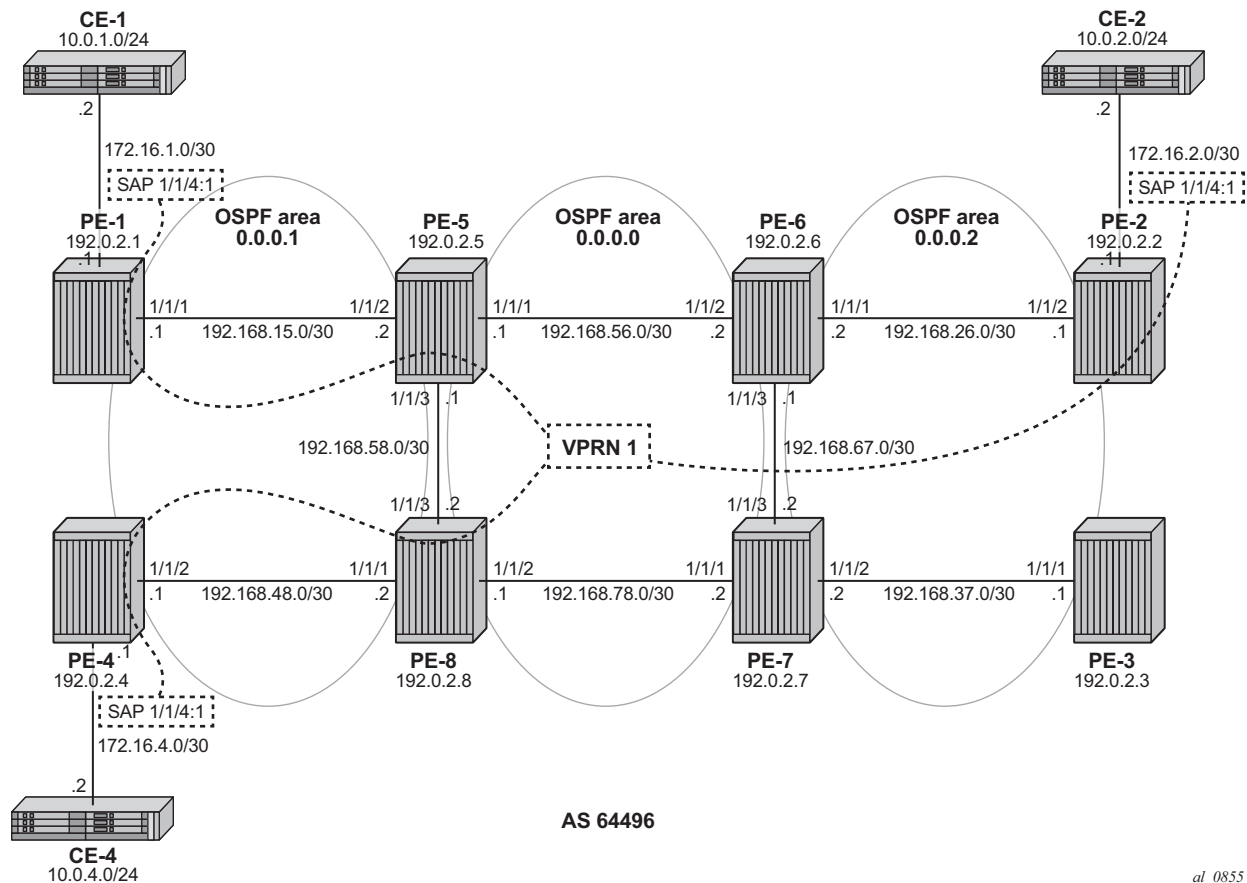
<snipped>

-----
Flags: B = BGP backup route available
       E = inactive best-external BGP route
=====
*A:P-6#

```

Nodes P-5 and P-6 act as stitching nodes to stitch RSVP LSPs. P-5 will stitch LSP-PE-1-P-5 and LSP-P-5-P-6 together while P-6 node will stitch LSP-P-5-P-6 and LSP-P-6-PE-2 together.

When the endpoints are defined, one corresponding RSVP LSP to those endpoints will be chosen (when ECMP=1). Selection criteria are as follows. When RSVP LSPs are configured under the T-LDP **tunneling** command (maximum 4), the one with the lowest LSP metric will be selected. When no RSVP LSPs are configured under the T-LDP **tunneling** command, LDP checks TTM for all available RSVP LSPs. The RSVP LSP with the least metric and operational state up will be selected.

Step 5.2 Traffic verification using a VPRN service.**Figure 94: VPRN 1 with LDPoRSVP and No Intra-Area PE Connectivity**

VPRN service 1 is set up between three PE nodes (PE-1/PE-2 and PE-4) using the **auto-bind tunnel resolution-filter ldp resolution filter** command. See also [Figure 94](#) for the exact addressing scheme.

```
*A:PE-1# configure service vprn 1 customer 1 create
      autonomous-system 64496
      route-distinguisher 64496:1
      auto-bind-tunnel
        resolution-filter
          ldp
        exit
      resolution filter
    exit
  vrf-target target:64496:1
  interface "int-PE-1-CE-1" create
    address 172.16.1.1/30
    sap 1/1/4:1 create
```



```

        exit
    exit
    static-route 10.0.1.0/24 next-hop 172.16.1.2
    no shutdown
    info
    exit all

```

In order to distribute VPRN information (vpn-ipv4 routes and VPRN service labels) across the service provider network, multi-protocol BGP (MP-BGP) is needed. MP-BGP is configured on PE-1, PE-2 and PE-4 with P-5 (192.0.2.5) acting as route reflector (RR). In this way no full BGP mesh between the three PE-nodes is needed, only a BGP peering towards RR.

```

*A:PE-1# configure router bgp
      group "internal"
        family ipv4 vpn-ipv4
        type internal
        neighbor 192.0.2.5
        exit
    exit
    no shutdown

```

```

*A:P-5# configure router bgp
      group "internal"
        family ipv4 vpn-ipv4
        type internal
        cluster 1.1.1.1
        neighbor 192.0.2.1
        exit
        neighbor 192.0.2.2
        exit
        neighbor 192.0.2.4
        exit
    exit
    no shutdown

```

If user traffic is monitored between PE-1 (ingress LER) and PE-2 (egress LER) three labels should be seen. The outer label is the transport label (distributed using RSVP protocol), the inner label is the service label (distributed using MP-BGP). LDPoRSVP will add an extra MPLS label between transport and service label (distributed using LDP). This middle label is used to tell the endpoint nodes (P-5 and P-6 acting as ABR) what to do.

Translated into show commands for traffic ingressing port 1/1/2 on P-5 node (PE-1<=>P-5 link):

Transport label 131071 is added as the top RSVP label on each user packet

```

*A:PE-1# show router rsvp session lsp-name "LSP-PE-1-P-5::path-PE-1-P-5" detail
=====
RSVP Sessions (Detailed)
=====
-----
LSP : LSP-PE-1-P-5::path-PE-1-P-5
-----

```

Configuration

```
From          : 192.0.2.1          To          : 192.0.2.5
Tunnel ID     : 1                  LSP ID      : 30720
Style         : SE                 State        : Up
Session Type  : Originate
In Interface  : n/a                Out Interface : 1/1/1
In Label      : n/a                Out Label    : 131071
Previous Hop  : n/a                Next Hop     : 192.168.15.2
<snipped>
```

LDPoRSVP label 131052 is added as the middle LDP label on each user packet

```
*A:PE-1# show router ldp bindings active prefixes prefix 192.0.2.2/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
      (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
       WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
       (S) - Static      (M) - Multi-homed Secondary Support
       (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                               Op           IngLbl      EgrLbl
EgrNextHop                           EgrIf/LspId
-----
192.0.2.2/32                         Push          --         131052
192.0.2.5                           LspId 1
192.0.2.2/32                         Swap          131065     131052
192.0.2.5                           LspId 1
-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:PE-1#
```

Service label 131061² is added as the inner MP-BGP label on each user packet

```
*A:PE-1# show router bgp neighbor 192.0.2.5 received-routes vpn-ipv4
=====
BGP Router ID:192.0.2.1      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
              l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP VPN-IPv4 Routes
=====
Flag Network                                     LocalPref  MED
```

2.This label will not change at endpoint nodes (P-5 and P-6). Ingress LER (PE-1) will push the service label to the user packet while the egress LER (PE-2) will pop the service label.

```

                Nexthop (Router)
                As-Path
                Path-Id      Label
-----
i      64496:1:10.0.1.0/24      100      None
      192.0.2.1                None      131061
      No As-Path
u*>i  64496:1:10.0.2.0/24      100      None
      192.0.2.2                None      131061
      No As-Path
u*>i  64496:1:10.0.4.0/24      100      None
      192.0.2.4                None      131061
      No As-Path
i      64496:1:172.16.1.0/30    100      None
      192.0.2.1                None      131061
      No As-Path
u*>i  64496:1:172.16.2.0/30    100      None
      192.0.2.2                None      131061
      No As-Path
u*>i  64496:1:172.16.4.0/30    100      None
      192.0.2.4                None      131061
      No As-Path
-----
Routes : 6
=====
*A:PE-1#

```

Translated into show commands for traffic ingressing port 1/1/2 on P-6 node (P-5<=>P-6 link):

Transport label 131068 is added as the top RSVP label on each user packet.

```

*A:P-5# show router rsvp session lsp-name "LSP-P-5-P-6::path-P-5-P-6" detail
=====
RSVP Sessions (Detailed)
=====
-----
LSP : LSP-P-5-P-6::path-P-5-P-6
-----
From          : 192.0.2.5      To          : 192.0.2.6
Tunnel ID     : 3              LSP ID      : 22016
Style         : SE             State         : Up
Session Type  : Originate
In Interface  : n/a            Out Interface : 1/1/1
In Label      : n/a            Out Label    : 131068
Previous Hop  : n/a            Next Hop     : 192.168.56.2

<snipped>

=====
*A:P-5#

```

LDPoRSVP label 131054 is added as the middle LDP label on each user packet.

```

*A:P-5# show router ldp bindings active prefixes prefix 192.0.2.2/32
=====

```

Configuration

```
LDP Bindings (IPv4 LSR ID 192.0.2.5:0)
(IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
        (S) - Static (M) - Multi-homed Secondary Support
        (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                                Op          IngLbl      EgrLbl
EgrNextHop                           EgrIf/LspId
-----
192.0.2.2/32                          Push        --          131054
192.0.2.6                             LspId 3
192.0.2.2/32                          Swap        131052      131054
192.0.2.6                             LspId 3
-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:P-5#
```

Service label 131061 is added as the inner MP-BGP label on each user packet.

Translated into show commands for traffic ingressing port 1/1/2 on PE-2 node (P-6<=>PE-2 link).

Transport label 131071 is added as the top RSVP label on each user packet.

```
*A:P-6# show router rsvp session lsp-name "LSP-P-6-PE-2::path-P-6-PE-2" detail
=====
RSVP Sessions (Detailed)
=====
LSP : LSP-P-6-PE-2::path-P-6-PE-2
-----
From          : 192.0.2.6          To          : 192.0.2.2
Tunnel ID     : 4                  LSP ID      : 37376
Style         : SE                 State        : Up
Session Type  : Originate
In Interface  : n/a                Out Interface : 1/1/1
In Label      : n/a                Out Label    : 131071
Previous Hop  : n/a                Next Hop     : 192.168.26.1

<snipped>

=====
*A:P-6#
```

LDPoRSVP label 131069 is added as the middle LDP label on each user packet.

```
*A:P-6# show router ldp bindings active prefixes prefix 192.0.2.2/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.6:0)
```

```

(IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
       WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
       (S) - Static          (M) - Multi-homed Secondary Support
       (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                               Op           IngLbl    EgrLbl
EgrNextHop                           EgrIf/LspId
-----
192.0.2.2/32                         Push          --        131069
192.0.2.2                           LspId 4
192.0.2.2/32                         Swap          131054    131069
192.0.2.2                           LspId 4
-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:P-6#

```

Service label 131061 is added as the inner MP-BGP label on each user packet.

Step 6. Intra-area FEC resolving (ingress LER is PE-1, egress LER is PE-4).

Step 6.1 Verification endpoint node and associated RSVP tunnel.

The first thing to do in the intra-area FEC resolving process is PE-1 performs an SPF calculation towards PE4 with the purpose to search for an eligible endpoint, as close as possible to PE-4. An endpoint is eligible when a T-LDP session exists between PE-1 and the endpoint node, tunneling is configured on the endpoint node, PE-1 received a label for the destination FEC from the endpoint and an RSVP LSP exists between PE-1 and endpoint node that can be used for LDPoRSVP.

First endpoint node in OSPF area 1 can be either P-5 or P-8 (only those nodes have a T-LDP session towards PE-1). With **show router ldp bindings active prefixes prefix 192.0.2.4/32** it can be concluded that P-5 will be the endpoint node.

```
*A:PE-1# show router ldp bindings active prefixes prefix 192.0.2.4/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
      (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use,  N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
        (S) - Static          (M) - Multi-homed Secondary Support
        (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                               Op           IngLbl      EgrLbl
EgrNextHop                           EgrIf/LspId
-----
192.0.2.4/32                         Push          --          131053
192.0.2.5                           LspId 1
192.0.2.4/32                         Swap          131067      131053
192.0.2.5                           LspId 1

-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:PE-1#
*A:PE-1# show router mpls lsp
=====
MPLS LSPs (Originating)
=====
LSP Name                             To           Tun      Fastfail  Adm  Opr
                                Id          Config
-----
LSP-PE-1-P-5                        192.0.2.5    1        No        Up   Up
LSP-PE-1-P-8                        192.0.2.8    2        No        Up   Up
-----
LSPs : 2
=====
*A:PE-1#
*A:PE-1# show router tunnel-table
=====
Tunnel Table (Router: Base)
```

```

=====
Destination      Owner      Encap TunnelId Pref      Nexthop      Metric
-----
192.0.2.5/32     rsvp      MPLS  1          7          192.168.15.2  16777215

<snipped>

=====
*A:PE-1#

```

On node P-5, the same commands can be repeated for the final destination node (PE-4). Also there, an RSVP LSP towards PE-4 will be used as transport tunnel for user packets can be seen.

```

*A:P-5# show router ldp bindings active prefixes prefix 192.0.2.4/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.5:0)
(IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
        (S) - Static          (M) - Multi-homed Secondary Support
        (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix      Op      IngLbl  EgrLbl
EgrNextHop  EgrIf/LspId
-----
192.0.2.4/32  Push      --      131069
192.0.2.4    LspId 2
192.0.2.4/32  Swap      131053  131069
192.0.2.4    LspId 2

-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:P-5#
*A:P-5# show router mpls lsp
=====
MPLS LSPs (Originating)
=====
LSP Name      To      Tun      Fastfail  Adm  Opr
                Id      Config
-----
LSP-P-5-PE-1  192.0.2.1  1      No      Up   Up
LSP-P-5-PE-4  192.0.2.4  2      No      Up   Up
LSP-P-5-P-6   192.0.2.6  3      Yes     Up   Up
LSP-P-5-P-7   192.0.2.7  4      Yes     Up   Up
LSP-P-5-P-8   192.0.2.8  5      Yes     Up   Up
-----
LSPs : 5
=====
*A:P-5#
*A:P-5# show router tunnel-table
=====

```

```
Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId  Pref      Nexthop      Metric
-----
<snipped>

192.0.2.4/32      rsvp       MPLS   2          7          192.168.58.2  16777215

<snipped>

=====
*A:P-5#
```

P-5 node acts as a stitching node to stitch RSVP LSPs. P-5 will stitch LSP-PE-1-P-5 and LSP-P-5-PE-4 together.

When the endpoint node (P-5) is defined, the corresponding RSVP LSP to this endpoint will be chosen. Selection criteria are as follows (when ECMP=1). When RSVP LSPs are configured under the T-LDP **tunneling** command (maximum 4), the one with the lowest LSP metric will be selected. When no RSVP LSPs are configured under the T-LDP **tunneling** command, LDP checks TTM for all available RSVP LSPs. The RSVP LSP with the lowest metric and operational state **up** will be selected.

Step 6.2 Traffic verification using a VPRN service (see [Figure 94 on page 540](#)).

If user traffic between PE-1 (ingress LER) and PE-4 (egress LER) is monitored, three labels are seen. The outer label is the transport label (distributed using RSVP protocol), the inner label is the service label (distributed using MP-BGP). LDPoRSVP will add an extra MPLS label between transport and service label (distributed using LDP). This middle label is used to tell the endpoint node (P-5) what to do.

Translated into show commands for traffic ingressing port 1/1/2 on P-5 node (PE-1<=>P-5 link):

Transport label 131071 is added as the top RSVP label on each user packet.

```
*A:PE-1# show router rsvp session lsp-name "LSP-PE-1-P-5::path-PE-1-P-5" detail
=====
RSVP Sessions (Detailed)
=====
LSP : LSP-PE-1-P-5::path-PE-1-P-5
-----
From          : 192.0.2.1          To          : 192.0.2.5
Tunnel ID     : 1                LSP ID      : 30720
Style         : SE               State        : Up
Session Type  : Originate
In Interface  : n/a              Out Interface : 1/1/1
In Label      : n/a              Out Label    : 131071
Previous Hop  : n/a              Next Hop     : 192.168.15.2

<snipped>
```


LDPoRSVP label 131053 is added as the middle LDP label on each user packet.

```
*A:PE-1# show router ldp bindings active prefixes prefix 192.0.2.4/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
      (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use,  N - Label Not In Use, W - Label Withdrawn
      WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
      (S) - Static           (M) - Multi-homed Secondary Support
      (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                               Op           IngLbl      EgrLbl
EgrNextHop                           EgrIf/LspId
-----
192.0.2.4/32                         Push          --         131053
192.0.2.5                           LspId 1
192.0.2.4/32                         Swap          131067     131053
192.0.2.5                           LspId 1
-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:PE-1#
```

Service label 131061 is added as the inner MP-BGP label on each user packet³.

```
*A:PE-1# show router bgp neighbor 192.0.2.5 received-routes vpn-ipv4
=====
BGP Router ID:192.0.2.1      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
              l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP VPN-IPv4 Routes
=====
Flag  Network                               LocalPref  MED
      Nexthop (Router)                     Path-Id    Label
      As-Path
-----
i      64496:1:10.0.1.0/24                   100        None
      192.0.2.1                             None       131061
      No As-Path
u*>i   64496:1:10.0.2.0/24                   100        None
```

3. This label will not change at endpoint node (P-5). Ingress LER (PE-1) will push the service label to the user packet while the egress LER (PE-4) will pop the service label.

Configuration

```

        192.0.2.2                                None          131061
        No As-Path
u*>i 64496:1:10.0.4.0/24                          100             None
        192.0.2.4                                None          131061
        No As-Path
i    64496:1:172.16.1.0/30                        100             None
        192.0.2.1                                None          131061
        No As-Path
u*>i 64496:1:172.16.2.0/30                        100             None
        192.0.2.2                                None          131061
        No As-Path
u*>i 64496:1:172.16.4.0/30                        100             None
        192.0.2.4                                None          131061
        No As-Path
-----
Routes : 6
=====
*A:PE-1#
```

Translated into show commands for traffic ingressing port 1/1/2 on node PE-4 (PE-4<=>P-8 link):

Transport label 131071 is added as the top RSVP label on each user packet.

```
*A:P-5# show router mpls lsp "LSP-P-5-PE-4" path detail 4
=====
MPLS LSP LSP-P-5-PE-4 Path (Detail)
=====
Legend :
    @ - Detour Available          # - Detour In Use
    b - Bandwidth Protected       n - Node Protected
    s - Soft Preemption
    S - Strict                    L - Loose
    A - ABR
=====
LSP LSP-P-5-PE-4 Path path-P-5-P-8-PE-4
-----
LSP Name      : LSP-P-5-PE-4                Path LSP ID : 3072
From          : 192.0.2.5                    To          : 192.0.2.4
Adm State     : Up                          Oper State  : Up
Path Name     : path-P-5-P-8-PE-4           Path Type   : Primary
Path Admin    : Up                          Path Oper   : Up
OutInterface  : 1/1/3                       Out Label   : 131068
Path Up Time  : 0d 00:17:53                 Path Dn Time: 0d 00:00:00
Retry Limit   : 0                           Retry Timer  : 30 sec
RetryAttempt  : 0                           NextRetryIn : 0 sec

<snipped>

ExplicitHops:
  192.168.58.2(S)    -> 192.168.48.1(S)
```

⁴ show router rsvp session lsp-name LSP-P-5-PE-4::path-P-5-P-8-PE-4 detail cannot be used since it only shows the outgoing RSVP label towards node P-8. On node P-8, RSVP transport label 131068 will be swapped into RSVP transport label 131071 for the link P-8 <=> PE-4.

```

Actual Hops :
    192.168.58.1 (192.0.2.5)          Record Label      : N/A
    -> 192.168.58.2 (192.0.2.8)      Record Label      : 131068
    -> 192.168.48.1                  Record Label      : 131071
ResigEligib*: False
LastResignal: n/a                      CSPF Metric : N/A
=====
* indicates that the corresponding row element may have been truncated.
*A:P-5#

```

LDPoRSVP label 131069 is added as the middle LDP label on each user packet.

```

*A:P-5# show router ldp bindings active prefixes prefix 192.0.2.4/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.5:0)
(IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use,  N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
        (S) - Static          (M) - Multi-homed Secondary Support
        (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====

```

Prefix	Op	IngLbl	EgrLbl
EgrNextHop	EgrIf/LspId		
192.0.2.4/32	Push	--	131069
192.0.2.4	LspId 2		
192.0.2.4/32	Swap	131053	131069
192.0.2.4	LspId 2		

```

-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:P-5#

```

Service label 131061 is added as the inner MP-BGP label on each user packet.

Additional Topics

prefer-tunnel-in-tunnel

If the next-hop router advertised the same FEC over link-level LDP (iLDP), LDP will prefer the iLDP tunnel by default unless the user explicitly changed the default preference using the **prefer-tunnel-in-tunnel** command. In this case an LDPoRSVP tunnel will have precedence.

Until now in this example, no RSVP LSPs are configured inside the **ldp targeted-session peer tunneling** context. Therefore, two additional strict non-cspf RSVP LSPs are added between ingress LER PE-5 node and egress LER P-1 node. Both LSPs will have an explicit metric setting and will be applied inside the **ldp tunneling** context. On the Layer 3 interface between PE-1 and P-5 node, iLDP is enabled.

```
A:PE-1# configure router ldp
      interface-parameters
        interface "int-PE-1-P-5"
      exit
    exit

A:P-5# configure router ldp
      interface-parameters
        interface "int-P-5-PE-1"
      exit
    exit

*A:PE-1# configure router mpls
      lsp "LSP-PE-1-P-5-metric100"
        to 192.0.2.5
        metric 100
        primary "path-PE-1-P-5"
      exit
    no shutdown
  exit
  lsp "LSP-PE-1-P-5-metric200"
    to 192.0.2.5
    metric 200
    primary "path-PE-1-P-5"
  exit
  no shutdown
exit

*A:PE-1# configure router ldp
      targeted-session
        peer 192.0.2.5
          tunneling
            lsp "LSP-PE-1-P-5-metric100"
            lsp "LSP-PE-1-P-5-metric200"
          exit
        exit
```

```

        exit
    exit

```

TTM on node PE-1 will look like this:

```

*A:PE-1# show router tunnel-table
=====
Tunnel Table (Router: Base)
=====
Destination          Owner      Encap TunnelId  Pref    Nexthop      Metric
-----
<snipped>

192.0.2.5/32         rsvp      MPLS    1           7       192.168.15.2 16777215
192.0.2.5/32         rsvp      MPLS    3           7       192.168.15.2 100
192.0.2.5/32         rsvp      MPLS    4           7       192.168.15.2 200
192.0.2.5/32         ldp       MPLS   65537       9       192.168.15.2 10

<snipped>

=====

```

Four LSPs are setup towards P-5 node, three RSVP LSPs and one LDP LSP. Tunnel ID 1 is a reference to LSP-PE-1-P-5. Tunnel ID 3 is a reference to LSP-PE-1-P-5-metric100. Tunnel ID 4 is a reference to LSP-PE-1-P-5-metric200 and owner LDP is a reference to iLDP.

Taken into account the FEC resolution rules, iLDP will win (no LDPoRSVP tunnel will be used).

```

*A:PE-1# show router ldp bindings active prefixes prefix 192.0.2.5/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
      (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use,  N - Label Not In Use, W - Label Withdrawn
      WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
      (S) - Static          (M) - Multi-homed Secondary Support
      (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                               Op      IngLbl  EgrLbl
EgrNextHop                           EgrIf/LspId
-----
192.0.2.5/32                         Push    --      131055
192.168.15.2                         1/1/1
192.0.2.5/32                         Swap    131068  131055
192.168.15.2                         1/1/1

-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:PE-1#

```

This behavior can be changed by setting the **prefer-tunnel-in-tunnel** command in the LDP context. Now, the LDPoRSVP tunnel with the best (= lowest) metric is taken.

```
*A:PE-1# configure router ldp prefer-tunnel-in-tunnel
*A:PE-1# show router ldp bindings active prefixes prefix 192.0.2.5/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
          (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use,  N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
        (S) - Static          (M) - Multi-homed Secondary Support
        (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                                     Op          IngLbl  EgrLbl
EgrNextHop                               EgrIf/LspId
-----
192.0.2.5/32                             Push        --      131055
192.0.2.5                               LspId 3
192.0.2.5/32                             Swap        131068  131055
192.0.2.5                               LspId 3
-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:PE-1#
*A:PE-1# show router mpls lsp
=====
MPLS LSPs (Originating)
=====
LSP Name                                To          Tun    Fastfail  Adm  Opr
                                         Id          Config
-----
LSP-PE-1-P-5                           192.0.2.5   1       No        Up   Up
LSP-PE-1-P-8                           192.0.2.8   2       No        Up   Up
LSP-PE-1-P-5-metric100                  192.0.2.5   3       No        Up   Up
LSP-PE-1-P-5-metric200                  192.0.2.5   4       No        Up   Up
-----
LSPs : 4
=====
*A:PE-1#
```

If the LSP-PE-1-P-5-metric 100 is shutdown, then the LSP-PE-1-P-5-metric 200 will become active.

```
*A:PE-1# configure router mpls
          lsp "LSP-PE-1-P-5-metric100"
          shutdown
          exit

*A:PE-1# show router ldp bindings active prefixes prefix 192.0.2.5/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
          (IPv6 LSR ID ::[0])
=====
```

LDP over RSVP Using OSPF as IGP

```

Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
        (S) - Static (M) - Multi-homed Secondary Support
        (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                               Op           IngLbl    EgrLbl
EgrNextHop                         EgrIf/LspId
-----
192.0.2.5/32                         Push         --        131055
192.0.2.5                          LspId 4
192.0.2.5/32                         Swap         131068    131055
192.0.2.5                          LspId 4
-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:PE-1#
**A:PE-1# show router mpls lsp
=====
MPLS LSPs (Originating)
=====
LSP Name                            To           Tun       Fastfail  Adm  Opr
                                Id          Config
-----
LSP-PE-1-P-5                        192.0.2.5    1         No        Up   Up
LSP-PE-1-P-8                        192.0.2.8    2         No        Up   Up
LSP-PE-1-P-5-metric100              192.0.2.5    3         No        Dwn  Dwn
LSP-PE-1-P-5-metric200              192.0.2.5    4         No        Up   Up
-----
LSPs : 4
=====
*A:PE-1#

```

If LSP-PE-1-P-5-metric 200 is shutdown, iLDP resumes.

```

*A:PE-1# configure router mpls
        lsp "LSP-PE-1-P-5-metric200"
        shutdown
        exit

*A:PE-1# show router ldp bindings active prefixes prefix 192.0.2.5/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
        (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
        (S) - Static (M) - Multi-homed Secondary Support
        (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                               Op           IngLbl    EgrLbl
EgrNextHop                         EgrIf/LspId
-----

```

Additional Topics

192.0.2.5/32	Push	--	131055
192.168.15.2	1/1/1		
192.0.2.5/32	Swap	131068	131055
192.168.15.2	1/1/1		

No. of IPv4 Prefix Active Bindings: 2
=====

Intra-PE Connectivity Will Change LDPoRSVP Behavior

Refer to [Figure 95](#). In the two metro areas, both of the intra PE's are physically connected with each other. Compared with the previous figures, PE-1 node is directly connected to PE-4 and PE-2 node is directly connected to PE-3 (up to the OSPF level).

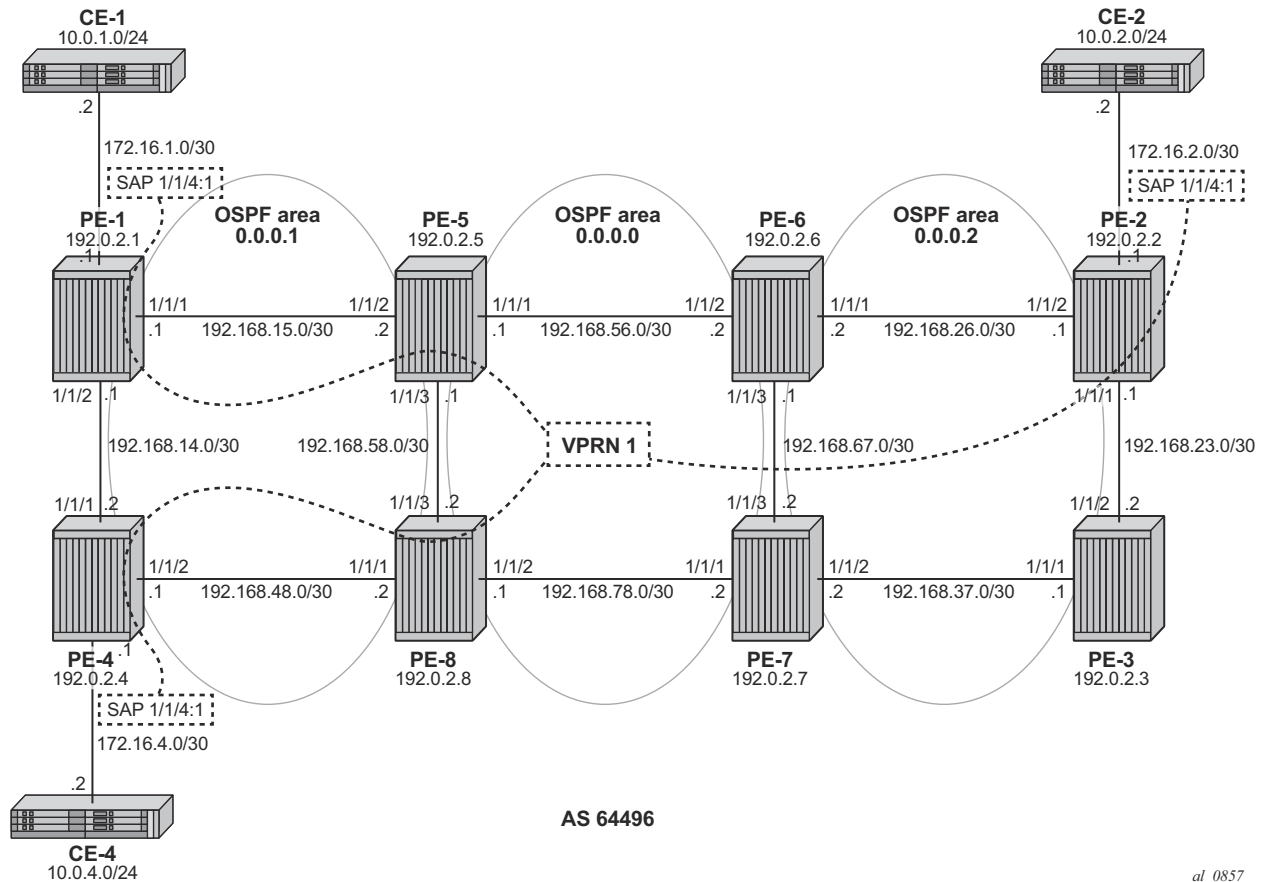


Figure 95: VPRN 1 with LDPoRSVP and Intra-Area PE Connectivity

The SPF path calculation on PE-1 towards destination (PE-4) will not point anymore to P-5 node (as was seen before) but will point directly to PE-4 (shortest, least IGP metric). As a conclusion it can be said that when possible intra-area endpoint node(s) are not part of the calculated SPF path, LDPoRSVP will be not be preferred anymore. For this situation it is advisable to configure iLDP on the intra-PE interfaces to have a fall back mechanism.

Translated into configuration commands on PE-1/PE-4 node, this becomes:

```
*A:PE-1# configure router
      interface "int-PE-1-PE-4"
```

Additional Topics

```
        address 192.168.14.1/30
        port 1/1/2
    exit

*A:PE-4# configure router
    interface "int-PE-4-PE-1"
        address 192.168.14.2/30
        port 1/1/1
    exit

*A:PE-1# configure router ospf
    area 0.0.0.1
        interface "int-PE-1-PE-4"
            interface-type point-to-point
        exit
    exit

*A:PE-4# configure router ospf
    area 0.0.0.1
        interface "int-PE-4-PE-1"
            interface-type point-to-point
        exit
    exit
```

From the moment iLDP is configured, an LDP LSP is setup. Intra-area PE traffic will flow over this LDP LSP.

```
*A:PE-1# configure router ldp
    interface-parameters
        interface "int-PE-1-PE-4"
    exit
exit

*A:PE-4# configure router ldp
    interface-parameters
        interface "int-PE-4-PE-1"
    exit
exit
```

```
A:PE-1# show router tunnel-table 192.0.2.4/32
=====
Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId  Pref    Nexthop      Metric
-----
192.0.2.4/32     ldp        MPLS  65544      9        192.168.14.2  10
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
A:PE-1#
```

If user traffic is monitored, between PE-1 (ingress LER) and PE-4 (egress LER) only two labels are seen. The outer one is the transport label (distributed using LDP protocol), the inner one is the

service label (distributed using MP-BGP). No LDPoRSVP label is present anymore. Translated into show commands for traffic ingressing port 1/1/1 on PE-4 node (PE-1<=>PE-4 link):

Transport label 131069 is added as the top LDP label on each user packet.

```
*A:PE-1# show router ldp bindings active prefixes prefix 192.0.2.4/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
              (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use,  N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
        (S) - Static          (M) - Multi-homed Secondary Support
        (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                               Op           IngLbl  EgrLbl
EgrNextHop                           EgrIf/LspId
-----
192.0.2.4/32                         Push         --      131069
192.168.14.2                        1/1/2
192.0.2.4/32                         Swap         131067  131069
192.168.14.2                        1/1/2
-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:PE-1#
```

Service label 131061 is added as the inner MP-BGP label on each user packet.

```
*A:PE-1# show router bgp neighbor 192.0.2.5 received-routes vpn-ipv4
=====
BGP Router ID:192.0.2.1      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP VPN-IPv4 Routes
=====
Flag  Network                               LocalPref  MED
      Nexthop (Router)                     Path-Id    Label
      As-Path
-----
u*>i  64496:1:10.0.2.0/24                   100        None
      192.0.2.2                           None       131061
      No As-Path
u*>i  64496:1:172.16.2.0/30                 100        None
      192.0.2.2                           None       131061
      No As-Path

<snipped>
```

Routes : 6
=====

Conclusion

LDPPoRSVP allows tunneling of user packets towards an LDP far-end destination inside an RSVP LSP (with the benefits of RSVP LSPs, fast-reroute (FRR) and traffic engineering (TE)). The main application of this feature is for deployment of MPLS based services, for example, VPRN, VLL and VPLS services, in large networks where a full mesh of LSPs reaches the limits of scalability.

MPLS LDP FRR using ISIS as IGP

In This Chapter

This section describes MPLS LDP FRR using ISIS as the IGP.

Topics in this section include:

- [Applicability on page 564](#)
- [Summary on page 565](#)
- [Configuration on page 566](#)
- [Conclusion on page 588](#)

Applicability

MPLS Label Distribution Protocol Fast Re-Route (LDP FRR) is supported on all 7x50 platforms including the 7750 SR c-4/12. This feature is supported on all IOM/IMMs and MDA/CMA types that support network interfaces from 9.0.R4 and higher. This feature was tested on release 13.0.R1. There are no pre-requisites for this configuration.

Summary

LDP FRR improves convergence in case of a single link or single node failure in the network. Convergence times will be in the order of 10s of milliseconds. This is important to some application services (like VoIP) which are sensitive to traffic loss when running over the MPLS network.

Without FRR, link and/or node failures inside an MPLS LDP network result in traffic loss in the order of 100s of milliseconds. The reason for that is that LDP depends on the convergence of the underlying IGP (IS-IS sending LSPs in this case). After IGP convergence, LDP itself needs to compute new primary next-hop Label Forwarding Entries (NHLFEs) for all affected Forwarding Equivalence Classes (FECs). Finally, the different Label Forwarding Information Bases (LFIBs) are updated.

When FRR is configured on a node, the node pre-computes primary NHLFEs for all FECs and in addition it will pre-compute backup NHLFEs for all FECs. The backup NHLFE corresponds to the label received for the same FEC from a Loop-Free Alternate (LFA) next-hop (see also RFC 5286, *Basic Specification for IP Fast Reroute: Loop-Free Alternates*). Both primary NHLFEs and backup NHLFEs are programmed in the IOM/IMM which makes it possible to converge very quickly.

Implementation

The 7x50 software has implemented Inequality 1 (link criterion) and Inequality 3 (node criterion) of RFC 5286. Similar to the Shortest Path Tree (SPT) computation that is part of standard link-state routing functionality, also the LFA next-hop computation is based on the IGP metric.

The underlying LFA formulas appear in the following format:

Inequality 1: $[SP(\text{backup NHR}, D) < \{SP(\text{backup NHR}, S) + SP(S, D)\}]$

Inequality 3: $[SP(\text{backup NHR}, D) < \{SP(\text{backup NHR}, PN) + SP(PN, D)\}]$

With 'SP' = 'shortest IGP metric path', 'NHR' = 'next-hop router', 'D' = 'destination', 'S' = 'source node or upstream node doing the actual LFA next-hop computation' and 'PN' = 'protected node'. The inequality 3 rule is stricter than the inequality 1 rule. See [Additional Topics on page 582](#) for a practical example on these formulas.

Configuration

This section provides information to configure:

- [Configuring the IP/MPLS network. on page 566](#)
- [Enabling LDP FRR and verify with show commands. on page 569](#)
- [Enable a synchronization timer between IGP and LDP protocol. on page 575](#)
- [Data path verification using a Layer 2/VLL service. on page 575](#)

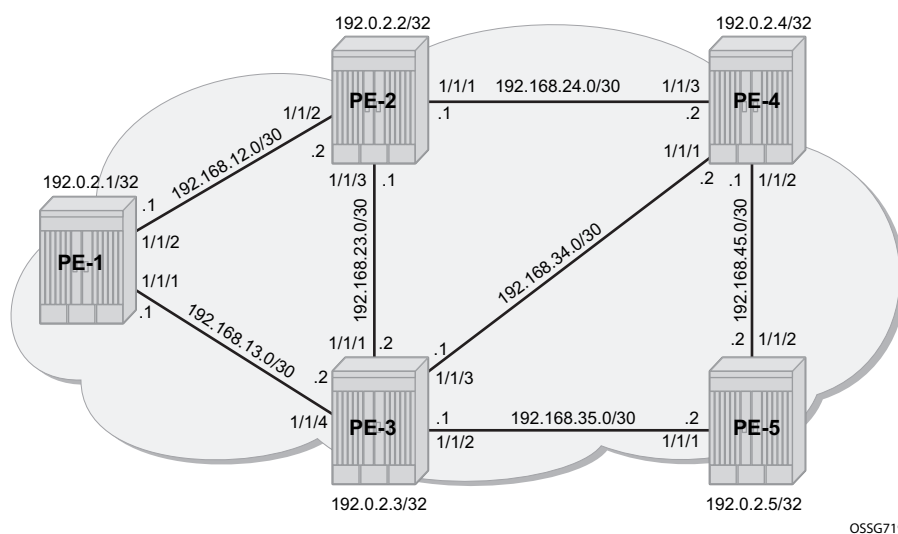


Figure 96: Initial Topology

Step 1. Configuring the IP/MPLS network.

The system addresses and IP interface addresses are configured according to [Figure 96](#). An interior gateway protocol (IGP) is needed to distribute routing information on all PEs. In our case, the IGP is IS-IS where each PE is acting as a Level 2 router. A configuration example is shown for PE-1. Similar configurations can be derived for the other PEs.

```
*A:PE-1# configure router isis
      level-capability level-2
      level 2
        wide-metrics-only
      exit
      interface "system"
      exit
      interface "int-PE-1-PE-2"
```

```

        interface-type point-to-point
    exit
    interface "int-PE-1-PE-3"
        interface-type point-to-point
    exit
    no shutdown

```

IS-IS interfaces are setup as type point-to-point to improve convergence since no DR/BDR election process is done. To verify that IS-IS adjacencies are up, **show router isis adjacency** is performed. To check if IP interface addresses/subnets are known on all PEs, **show router route-table** or **show router fib slot-number** will display the content of the forwarding information base (FIB).

```

*A:PE-1# show router isis adjacency
=====
Router Base ISIS Instance 0 Adjacency
=====
System ID              Usage State Hold Interface          MT-ID
-----
PE-2                   L2    Up    26    int-PE-1-PE-2          0
PE-3                   L2    Up    20    int-PE-1-PE-3          0
-----
Adjacencies : 2
=====
*A:PE-1#
*A:PE-1# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]      Type   Proto   Age      Pref
Next Hop[Interface Name] Metric
-----
192.0.2.1/32            Local  Local   07d23h34m  0
    system
192.0.2.2/32            Remote  ISIS    07d23h34m  18
    192.168.12.2
192.0.2.3/32            Remote  ISIS    07d23h17m  18
    192.168.13.2
192.0.2.4/32            Remote  ISIS    07d22h58m  18
    192.168.12.2
192.0.2.5/32            Remote  ISIS    07d23h17m  18
    192.168.13.2
192.168.12.0/30          Local  Local   07d23h34m  0
    int-PE-1-PE-2
192.168.13.0/30          Local  Local   07d23h34m  0
    int-PE-1-PE-3
192.168.23.0/30          Remote  ISIS    07d23h16m  18
    192.168.12.2
192.168.24.0/30          Remote  ISIS    07d23h34m  18
    192.168.12.2
192.168.34.0/30          Remote  ISIS    07d23h17m  18
    192.168.13.2
192.168.35.0/30          Remote  ISIS    07d23h17m  18
    192.168.13.2
192.168.45.0/30          Remote  ISIS    03h59m10s  18
    192.168.12.2
-----

```

Configuration

```
No. of Routes: 12
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====
*A:PE-1#
*A:PE-1# show router fib 1
=====
FIB Display
=====
Prefix [Flags]                                Protocol
NextHop
-----
192.0.2.1/32                                LOCAL
    192.0.2.1 (system)
192.0.2.2/32                                ISIS
    192.168.12.2 (int-PE-1-PE-2)
192.0.2.3/32                                ISIS
    192.168.13.2 (int-PE-1-PE-3)
192.0.2.4/32                                ISIS
    192.168.12.2 (int-PE-1-PE-2)
192.0.2.5/32                                ISIS
    192.168.13.2 (int-PE-1-PE-3)
192.168.12.0/30                             LOCAL
    192.168.12.0 (int-PE-1-PE-2)
192.168.13.0/30                             LOCAL
    192.168.13.0 (int-PE-1-PE-3)
192.168.23.0/30                             ISIS
    192.168.12.2 (int-PE-1-PE-2)
192.168.24.0/30                             ISIS
    192.168.12.2 (int-PE-1-PE-2)
192.168.34.0/30                             ISIS
    192.168.13.2 (int-PE-1-PE-3)
192.168.35.0/30                             ISIS
    192.168.13.2 (int-PE-1-PE-3)
192.168.45.0/30                             ISIS
    192.168.12.2 (int-PE-1-PE-2)
-----
Total Entries : 12
-----
=====
*A:PE-1#
```

Initially, the default IS-IS Level 2 metric is applied on all interfaces (value 10).

```
*A:PE-1# show router isis status | match "L2 Default Metric"
L2 Default Metric      : 10
*A:PE-1#
```

The next step in the process of setting up the IP/MPLS network is setting up interface-LDP sessions on all interfaces. If the keyword dual-stack and ipv4 no shutdown isn't included in the command, it will be added automatically.

```
*A:PE-1# configure router ldp
      interface-parameters
        interface "int-PE-1-PE-2"
        exit
        interface "int-PE-1-PE-3" dual-stack
        ipv4
        no shutdown
        exit
      exit
    exit
  targeted-session
  exit
  no shutdown
exit all
*A:PE-1#
```

There is now a full mesh of LDP LSPs setup between all PE's system interfaces. As an example, the tunnel-table on PE-1 will look like this:

```
*A:PE-1# show router tunnel-table
=====
Tunnel Table (Router: Base)
=====
Destination          Owner Encap TunnelId  Pref    Nexthop      Metric
-----
192.0.2.2/32         ldp   MPLS   -         9       192.168.12.2  10
192.0.2.3/32         ldp   MPLS   -         9       192.168.13.2  10
192.0.2.4/32         ldp   MPLS   -         9       192.168.12.2  20
192.0.2.5/32         ldp   MPLS   -         9       192.168.13.2  20
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
*A:PE-1#
```

Note that the LDP LSP metric follows the IGP cost. Optionally, LSP metrics can be applied but this is out of the scope for this configuration note.

Step 2. Enabling LDP FRR and verify with show commands.

Since LDP FRR is using LFA next-hop pre-computation by the IGP (as described in RFC 5286), the IGP CLI configuration will look like this:

```
*A:PE-1# configure router isis loopfree-alternate
*A:PE-1# show router isis status | match Loopfree
Loopfree-Alternate    : Enabled
*A:PE-1#
```

After enabling LFA inside the IGP context, FRR needs to be enabled within the LDP context:

```
*A:PE-1# configure router ldp fast-reroute
*A:PE-1# show router ldp status | match FRR
FRR                      : Enabled                Mcast Upstream FRR    : Disabled
*A:PE-1#
```

After these two CLI settings, the software pre-computes for each LDP FEC in the network both a primary and a backup NHLFE and uploads it to the IOM/IMM. The primary NHLFE corresponds to the label of the FEC received from the primary next-hop as per standard LDP resolution of the FEC prefix in the Routing Table Manager (RTM). The backup NHLFE corresponds to the label received for the same FEC from an LFA next-hop.

Note: For point-to-point interfaces, when multiple LFA next-hops are found for a given primary next-hop, the following selection algorithms are used:

- It will pick the node-protect type in favor of the link-protect type.
- If there is more than one LFA next-hop within the selected type, then it will pick one based on the least cost.
- If more than one LFA next-hop with the same cost, SPF will select the first one. This is not a deterministic selection and will vary following each SPF calculation.

Several show commands are possible to display LFA information:

- The **show router isis statistics** command displays the number of LFA runs on a specific node.

```
*A:PE-1# show router isis statistics
=====
Router Base ISIS Instance 0 Statistics
=====

<snipped>

LFA Statistics
LFA Runs      : 22
  Last runTimeStamp: 03/13/2015 13:48:06
Partial LFA Runs : 2
  Last runTimeStamp: 03/13/2015 13:23:47
<snipped>
=====
*A:PE-1#
```

- The **show router isis lfa-coverage** command performs a mathematical calculation between the number of nodes and IPv4/IPv6 routes in the network versus present LFA next-hop protections. In our network (see [Figure 96](#)), all IS-IS links have a default Level 2 metric of 10. This results in all four nodes and all IS-IS routes learned by PE1 being 100% LFA protected (link or node). Refer to the following output.

```
*A:PE-1# show router isis lfa-coverage
=====
```

```

LFA Coverage
=====
Topology          Level   Node           IPv4             IPv6
-----
IPV4 Unicast      L1      0/0 (0%)       9/9 (100%)      0/0 (0%)
IPV6 Unicast      L1      0/0 (0%)       0/0 (0%)        0/0 (0%)
IPV4 Multicast    L1      0/0 (0%)       0/0 (0%)        0/0 (0%)
IPV6 Multicast    L1      0/0 (0%)       0/0 (0%)        0/0 (0%)
IPV4 Unicast      L2      4/4 (100%)     9/9 (100%)      0/0 (0%)
IPV6 Unicast      L2      0/0 (0%)       0/0 (0%)        0/0 (0%)
IPV4 Multicast    L2      0/0 (0%)       0/0 (0%)        0/0 (0%)
IPV6 Multicast    L2      0/0 (0%)       0/0 (0%)        0/0 (0%)
=====
*A:PE-1#

```

- The **show router isis topology lfa detail** command shows the LFA protection type (link or node).

```

*A:PE-1# show router isis topology lfa detail
=====
Router Base ISIS Instance 0 Topology Table
=====
IS-IS IP paths (MT-ID 0), Level 2
-----
Node       : PE-2.00                      Metric    : 10
Interface  : int-PE-1-PE-2                SNPA      : none
Nexthop    : PE-2

LFA intf   : int-PE-1-PE-3                LFA Metric : 20
LFA nh     : PE-3                         LFA type   : linkProtection

Node       : PE-3.00                      Metric    : 10
Interface  : int-PE-1-PE-3                SNPA      : none
Nexthop    : PE-3

LFA intf   : int-PE-1-PE-2                LFA Metric : 20
LFA nh     : PE-2                         LFA type   : linkProtection

Node       : PE-4.00                      Metric    : 20
Interface  : int-PE-1-PE-2                SNPA      : none
Nexthop    : PE-2

LFA intf   : int-PE-1-PE-3                LFA Metric : 20
LFA nh     : PE-3                         LFA type   : nodeProtection

Node       : PE-5.00                      Metric    : 20
Interface  : int-PE-1-PE-3                SNPA      : none
Nexthop    : PE-3

LFA intf   : int-PE-1-PE-2                LFA Metric : 30
LFA nh     : PE-2                         LFA type   : linkProtection
=====
*A:PE-1#

```

- The **show router route-table** command adds an 'L' flag as reference that the associated prefix is having also an LFA next-hop available. For detailed interface address information used by the LFA calculation use the **show router route-table alternative** or **show router isis alternative** command. The output on PE-1 for PE-4 will look like this:

```
*A:PE-1# show router route-table 192.0.2.4
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type    Proto    Age          Pref
Next Hop[Interface Name]                          Metric
-----
192.0.2.4/32 [L]                                  Remote  ISIS     07d23h01m    18
192.168.12.2                                      20
-----

No. of Routes: 1
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====

*A:PE-1#
*A:PE-1# show router route-table alternative 192.0.2.4
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type    Proto    Age          Pref
Next Hop[Interface Name]                          Metric
Alt-NextHop                                         Alt-
Metric
-----
192.0.2.4/32                                       Remote  ISIS     07d23h02m    18
192.168.12.2                                      20
192.168.13.2 (LFA)                                20
-----

No. of Routes: 1
Flags: n = Number of times nexthop is repeated
      Backup = BGP backup route
      LFA = Loop-Free Alternate nexthop
      S = Sticky ECMP requested
=====

*A:PE-1#
*A:PE-1# show router isis routes 192.0.2.4 alternative
=====
Route Table
=====
Prefix[Flags]                                Metric    Lvl/Typ    Ver.  SysID/Hostname
NextHop                                      MT        AdminTag
Alt-Nexthop                                Alt-      Alt-Type
Metric
-----
192.0.2.4/32                                20        2/Int.     28    PE-2
192.168.12.2                                0         0          0     0
192.168.13.2 (L)                            20        NP         20    NP
-----

No. of Routes: 1
Flags: L = Loop-Free Alternate nexthop
Legend: LP = linkProtection, NP = nodeProtection
```



```
=====
*A:PE-1#
```

On PE-1, PE-4 (192.0.2.4/32) has a primary SPF next-hop pointing towards PE-2 (192.168.12.2) and an LFA next-hop pointing towards PE-3 (192.168.13.2).

Using the Inequality 3 formula on PE-1 for prefix 192.0.2.4/32, this becomes:

Inequality 3:

$$[SP(\text{backup NHR}, D) < \{SP(\text{backup NHR}, PN) + SP(PN, D)\}] \text{ or } [SP(PE-3, PE-4) < \{SP(PE-3, PE-2) + SP(PE-2, PE-4)\}] \text{ or } [10 < \{10 + 10\}]$$

This means that Inequality 3 is met. The calculated LFA next-hop for prefix 192.0.2.4/32 on PE-1 is node-protecting PE-2, see [Figure 97](#) for a graphical representation.

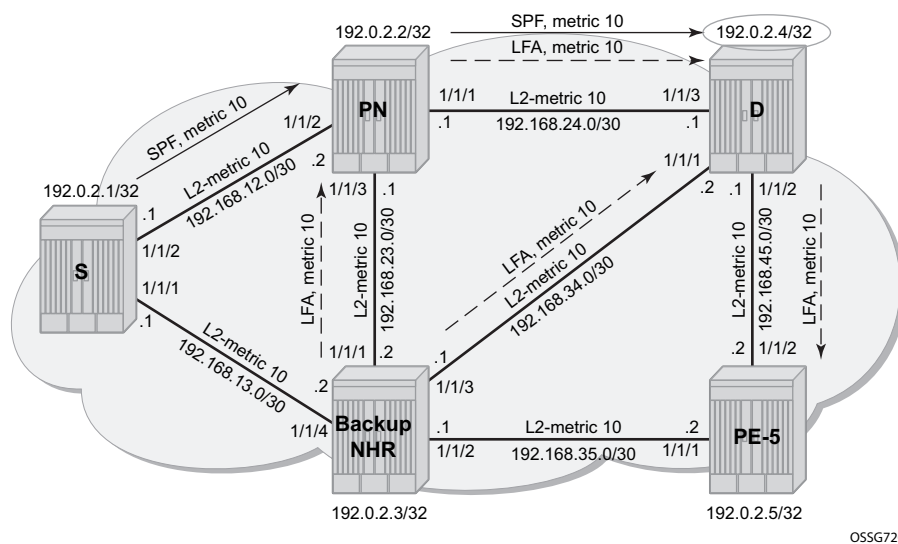


Figure 97: LFA Computation, Inequality 3 for Prefix PE-4 (D) on PE1 (S)

- The **show router ldp bindings** command displays the Label Information Base (LIB). A BU flag is present in case the associated label is used as backup NHLFE for the given prefix¹. As an example, a display on PE-1 for prefix PE-4 will look like this:

```
*A:PE-1# show router ldp bindings prefixes prefix 192.0.2.4/32
```

```
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
(IPv6 LSR ID ::[0])
```

```
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
```

- This is only possible because the 7x50 LDP implementation is using liberal retention mode which means that every label mapping received by a peer is retained regardless of whether the LSR is the next-hop for the advertised mapping.

```

WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
=====
LDP IPv4 Prefix Bindings
=====
Prefix                               IngLbl                               EgrLbl
Peer                                EgrIntf/LspId
EgrNextHop
-----
192.0.2.4/32                         131068N                             131068
192.0.2.2:0                         1/1/2
192.168.12.2

192.0.2.4/32                         131068U                             131068BU
192.0.2.3:0                         1/1/1
192.168.13.2
-----
No. of IPv4 Prefix Bindings: 2
=====
*A:PE-1#

```

- The **show router ldp bindings active** command displays the Label Forwarding Information Base (LFIB). Also the BU flag is present and in addition a reference to the label action itself: **pop** for eLER, **push** for iLER and **swap** for LSR.

```

*A:PE-1# show router ldp bindings active prefixes prefix 192.0.2.4/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
(IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
(S) - Static (M) - Multi-homed Secondary Support
(B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                               Op                               IngLbl                               EgrLbl
EgrNextHop                          EgrIf/LspId
-----
192.0.2.4/32                         Push                             --                                131068
192.168.12.2                         1/1/2

192.0.2.4/32                         Push                             --                                131068BU
192.168.13.2                         1/1/1

192.0.2.4/32                         Swap                             131068                            131068
192.168.12.2                         1/1/2

192.0.2.4/32                         Swap                             131068                            131068BU
192.168.13.2                         1/1/1
-----
No. of IPv4 Prefix Active Bindings: 4
=====
*A:PE-1#

```

Step 3. Enable a synchronization timer between IGP and LDP protocol.

Within an MPLS network using LDP it is common practice to enable a synchronization timer between LDP and the IGP. Also when LDP FRR is enabled, a situation can occur in which a synchronization timer between IGP and LDP will help: the revert scenario. When the interface for the previous primary next-hop is restored, IGP may re-converge before LDP completed the FEC exchange with its neighbor over that interface. This may cause LDP to de-program the LFA next-hop from the FEC and blackhole traffic.

In order to avoid these situations, it is recommended to first enable IGP-LDP synchronization on the LDP interface. The time is expressed in seconds and can have a value between 1 and 1800 seconds. Translated into configuration commands, this becomes:

```
*A:PE-1# configure router interface "int-PE-1-PE-2" ldp-sync-timer 10
*A:PE-1# configure router interface "int-PE-1-PE-3" ldp-sync-timer 10
```

When this timer is enabled, it means that when an interface is restored again, the IGP will advertise this link in the network with an infinite metric. The **ldp-sync-timer** is started, LDP adjacencies are brought up together with a label exchange. After the **ldp-sync-timer** expires, the normal metric is advertised in the network again.

Step 4. Data path verification using a Layer 2/VLL service.

Traffic generator ports are connected towards PE-1 and PE-5 for data verification, thus an Epipe service is created using an MPLS LDP based SDP on both PE-1 and PE-5.

```
*A:PE-1# configure service sdp 5
      far-end 192.0.2.5
      ldp
      keep-alive
      shutdown
      exit
      no shutdown

*A:PE-1# configure service epipe 1
      service-mtu 1450
      sap 1/1/3:1 create
      exit
      spoke-sdp 5:1 create
      no shutdown
      exit
      no shutdown
```

A similar configuration is configured on PE-5.

The IS-IS Level 2 metric value on the interface between PE-4 and PE-5 is decreased to 5, see [Figure 98](#).

```
*A:PE-4# configure router isis interface int-PE-4-PE-5 level 2 metric 5
*A:PE-5# configure router isis interface int-PE-5-PE-4 level 2 metric 5
```

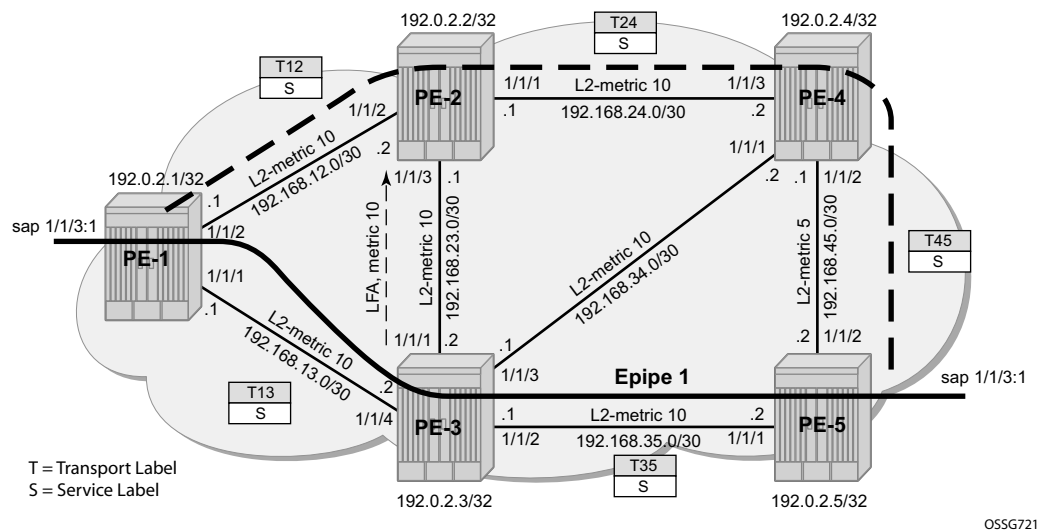


Figure 98: Data Verification, Direction PE-1 => PE-5 Using VLL Service

In this setup, PE-3 is node-protected for PE-5 prefix on PE-1:

```
*A:PE-1# show router isis topology lfa detail
=====
Router Base ISIS Instance 0 Topology Table
=====
IS-IS IP paths (MT-ID 0),   Level 2
-----
<snipped>
Node       : PE-5.00           Metric      : 20
Interface  : int-PE-1-PE-3     SNPA        : none
Nexthop    : PE-3

LFA intf   : int-PE-1-PE-2     LFA Metric   : 25
LFA nh     : PE-2              LFA type     : nodeProtection
=====
*A:PE-1#
*A:PE-1# show router isis routes alternative 192.0.2.5
=====
Route Table
=====
Prefix[Flags]          Metric   Lvl/Typ    Ver.  SysID/Hostname
NextHop                MT      AdminTag
Alt-Nexthop            Alt-   Alt-Type
Metric
```

```

192.0.2.5/32                20          2/Int.        20      PE-3
    192.168.13.2            0              0
    192.168.12.2 (L)        25          NP
-----
No. of Routes: 1
Flags: L = Loop-Free Alternate nexthop
Legend: LP = linkProtection, NP = nodeProtection
=====
*A:PE-1#

```

In normal conditions, MPLS traffic from PE-1 towards PE-5 over Epipe 1 will have two MPLS labels: 1) outer (transport) label given by LDP protocol, swapped on each intermediate LSR and 2) inner (service) label given by T-LDP, the same end-to-end. Refer to the following show commands.

The T-LDP service label is S (131066):

```

*A:PE-1# show router ldp bindings services service-id 1
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
      (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
       S - Status Signaled Up,  D - Status Signaled Down
       E - Epipe Service, V - VPLS Service, M - Mirror Service
       A - Apipe Service, F - Fpipe Service, I - IES Service, R - VPRN service
       P - Ipipe Service, WP - Label Withdraw Pending, C - Cpipe Service
       BU - Alternate For Fast Re-Route, TLV - (Type, Length: Value)
=====
LDP Service FEC 128 Bindings
=====
Type          VCId      SDPId      IngLbl  LMTU
Peer          SvcId      EgrLbl  RMTU
-----
E-Eth                1          5        131066U 1436
192.0.2.5:0          1                131066S 1436
-----
No. of VC Labels: 1
=====
LDP Service FEC 129 Bindings
=====
SAII          AGII          IngLbl  LMTU
TAII          Type          EgrLbl  RMTU
Peer          SvcId      SDPId
-----
No Matching Entries Found
=====
*A:PE-1#

```

The transport LDP label between PE-1 and PE-3 for prefix 192.0.2.5/32 is T13 (131067):

```

*A:PE-1# show router ldp bindings active prefixes prefix 192.0.2.5/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)

```

Configuration

```
(IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
        (S) - Static          (M) - Multi-homed Secondary Support
        (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                               Op           IngLbl      EgrLbl
EgrNextHop                           EgrIf/LspId
-----
192.0.2.5/32                         Push         --         131067
192.168.13.2                        1/1/1
192.0.2.5/32                         Push         --         131067BU
192.168.12.2                        1/1/2
192.0.2.5/32                         Swap         131067     131067
192.168.13.2                        1/1/1
192.0.2.5/32                         Swap         131067     131067BU
192.168.12.2                        1/1/2

-----
No. of IPv4 Prefix Active Bindings: 4
=====
*A:PE-1#
```

The transport LDP label between PE-3 and PE-5 for prefix 192.0.2.5/32 is T35 (131071):

```
*A:PE-3# show router ldp bindings active prefixes prefix 192.0.2.5/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.3:0)
(IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
        (S) - Static          (M) - Multi-homed Secondary Support
        (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                               Op           IngLbl      EgrLbl
EgrNextHop                           EgrIf/LspId
-----
192.0.2.5/32                         Push         --         131071
192.168.35.2                        1/1/2
192.0.2.5/32                         Push         --         131067BU
192.168.34.2                        1/1/3
192.0.2.5/32                         Swap         131067     131071
192.168.35.2                        1/1/2
192.0.2.5/32                         Swap         131067     131067BU
192.168.34.2                        1/1/3

-----
```

No. of IPv4 Prefix Active Bindings: 4

*A:PE-3#

When PE-3 reboots, PE-1 performs an immediate swap to LFA next-hop for prefix 192.0.2.5/32 bypassing PE-3. The service label remains the same, only the transport labels can change on the network ports PE-1 <=> PE-2, PE-2 <=> PE-4 and PE-4 <=> PE-5². Refer to the following show commands.

The T-LDP service label is S (131066):

```
*A:PE-1# show router ldp bindings services service-id 1
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
(IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
        S - Status Signaled Up, D - Status Signaled Down
        E - Epipe Service, V - VPLS Service, M - Mirror Service
        A - Apipe Service, F - Fpipe Service, I - IES Service, R - VPRN service
        P - Ipipe Service, WP - Label Withdraw Pending, C - Cpipe Service
        BU - Alternate For Fast Re-Route, TLV - (Type, Length: Value)
=====
LDP Service FEC 128 Bindings
=====
Type          VCId      SDPId      IngLbl  LMTU
Peer          SvcId      EgrLbl  RMTU
-----
E-Eth                1          5        131066U 1436
192.0.2.5:0         1          1        131066S 1436
-----
No. of VC Labels: 1
=====
LDP Service FEC 129 Bindings
=====
SAII          AGII          IngLbl  LMTU
TAII          Type          EgrLbl  RMTU
Peer          SvcId      SDPId
-----
No Matching Entries Found
=====
*A:PE-1#
```

The transport LDP label value between PE-1 and PE-2 for prefix 192.0.2.5/32 is the same label (previously tagged as BU) as before the node failure event: T12 (131067):

```
*A:PE-1# show router ldp bindings active prefixes prefix 192.0.2.5/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
(IPv6 LSR ID ::[0])
=====
```

-
2. LDP FRR MPLS label stack will never contain more than two labels. This is different when compared to RSVP-TE FRR facility mode which uses a three-label MPLS stack.

Configuration

```
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
       WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
       (S) - Static          (M) - Multi-homed Secondary Support
       (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                               Op           IngLbl      EgrLbl
EgrNextHop                          EgrIf/LspId
-----
192.0.2.5/32                         Push         --         131067
192.168.12.2                        1/1/2
192.0.2.5/32                         Swap         131067     131067
192.168.12.2                        1/1/2
-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:PE-1#
```

The transport LDP label between PE-2 and PE-4 for prefix 192.0.2.5/32 is T24 (131067):

```
*A:PE-2# show router ldp bindings active prefixes prefix 192.0.2.5/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.2:0)
      (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
       WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
       (S) - Static          (M) - Multi-homed Secondary Support
       (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                               Op           IngLbl      EgrLbl
EgrNextHop                          EgrIf/LspId
-----
192.0.2.5/32                         Push         --         131067
192.168.24.2                        1/1/1
192.0.2.5/32                         Swap         131067     131067
192.168.24.2                        1/1/1
-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:PE-2#
```

The transport LDP label between PE-4 and PE-5 for prefix 192.0.2.5/32 is T45 (131071):

```
*A:PE-4# show router ldp bindings active prefixes prefix 192.0.2.5/32
=====
LDP Bindings (IPv4 LSR ID 192.0.2.4:0)
      (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
```



```

WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
(S) - Static          (M) - Multi-homed Secondary Support
(B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                               Op           IngLbl  EgrLbl
EgrNextHop                          EgrIf/LspId
-----
192.0.2.5/32                         Push          --      131071
192.168.45.2                        1/1/2
192.0.2.5/32                         Swap          131067  131071
192.168.45.2                        1/1/2
-----
No. of IPv4 Prefix Active Bindings: 2
=====
*A:PE-4#

```

Additional Topics

Metric Change

Ensure the network is back to its initial topology with all the level 2 metrics back to their default value (10) before applying the changes referenced below:

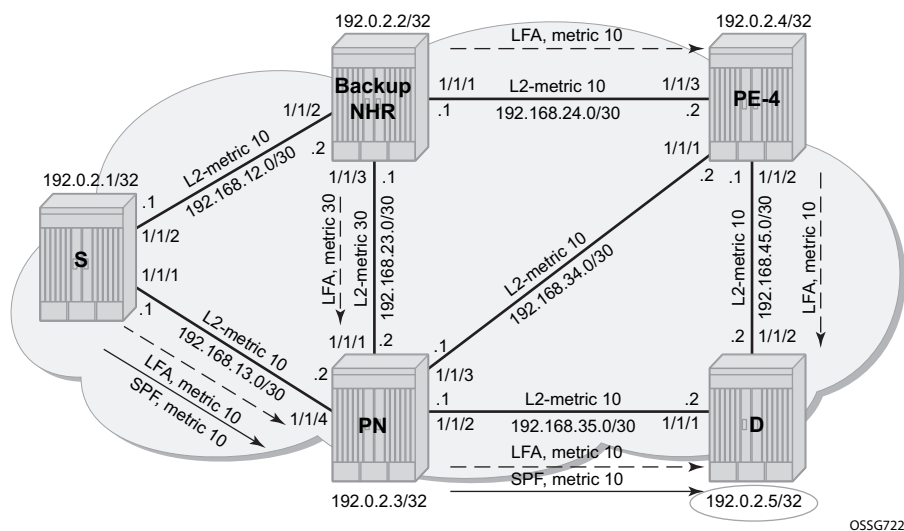
```
*A:PE-4# configure router isis interface "int-PE-4-PE-5" level 2 no metric
*A:PE-5# configure router isis interface "int-PE-5-PE-4" level 2 no metric
```

Suppose that the IS-IS Level 2 metric between PE-2 and PE-3 changes to 30, then 100% LFA coverage is no longer possible. Translated into configuration commands, this becomes:

```
*A:PE-3# configure router isis interface "int-PE-3-PE-2" level 2 metric 30
*A:PE-2# configure router isis interface "int-PE-2-PE-3" level 2 metric 30
```

On PE-1, Inequality 3 formula will find LFA next-hop coverages for prefix PE-4 and PE-5. Inequality formula 1 will find LFA next-hop coverages for prefix PE-4, PE-5 and the subnet between PE-4 and PE-5.

Both inequality formulas are visualized in [Figure 99](#) and [Figure 100](#) for prefix 192.0.2.5/32 (= PE-5) on PE-1 acting as the source node for LFA next-hop computation.



OSSG722

Figure 99: LFA Computation, Inequality 3 for Prefix PE-5 (D) on PE-1 (S)

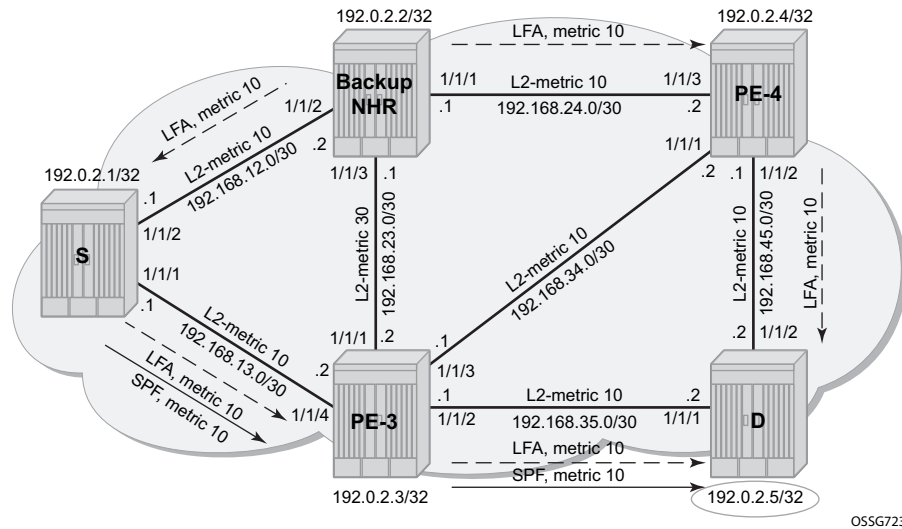


Figure 100: LFA Computation, Inequality 1 for Prefix PE-5 (D) on PE-1 (S)

Inequality 3 formula: $[SP(\text{backup NHR}, D) < \{SP(\text{backup NHR}, PN) + SP(PN, D)\}]$

For a node LFA next-hop calculation of prefix 192.0.2.5/32 (D) on PE-1, the above formula translated into text becomes:

[shortest path from backup next-hop router (PE-2) towards destination (PE-5) must be smaller than {shortest path from backup next-hop router (PE-2) towards protected node (PE-3) + shortest path from protected node (PE-3) to destination (PE-5)}].

The shortest path from backup next-hop router (PE-2) towards destination (PE-5) is going over PE-4, using IS-IS Level 2 metric 10 for interface PE-2 <=> PE-4 and IS-IS Level 2 metric 10 for interface PE-4 <=> PE-5. The shortest path from backup next-hop router (PE-2) towards protected node (PE-3) uses IS-IS Level 2 metric 30 for interface PE-2 <=> PE-3. The shortest path from protected node (PE-3) to destination (PE-5) uses IS-IS Level 2 metric 10 for interface PE-3 <=> PE-5. A concise example is displayed below.

Prefix 192.0.2.5/32: SP (PE-2, PE-5) < SP (PE-2, PE-3) + SP (PE-3, PE-5)
 $10 + 10 < 30 + 10 \Rightarrow \text{OK}$

Inequality 1 formula: $[SP(\text{backup NHR}, D) < \{SP(\text{backup NHR}, S) + SP(S, D)\}]$

For a link LFA next-hop calculation of prefix 192.0.2.5/32 (D) on PE-1, the formula translated displayed above into text becomes:

[shortest path from backup next-hop router (PE-2) towards destination (PE-5) must be smaller than {shortest path from backup next-hop router (PE-2) towards source (PE-1) + shortest path from source (PE-1) to destination (PE-5)}].

The shortest path from backup next-hop router (PE-2) towards destination (PE-5) is going over PE-4, using IS-IS Level 2 metric 10 for interface PE-2 <=> PE-4 and IS-IS Level 2 metric 10 for interface PE-4 <=> PE-5. The shortest path from backup next-hop router (PE-2) towards source (PE-1) uses IS-IS Level 2 metric 10 for interface PE-2 <=> PE-1. The shortest path from source (PE-1) to destination (PE-5) follows the normal SPF calculation, going over PE-3, using IS-IS Level 2 metric 10 for interface PE-1 <=> PE-3 and IS-IS Level 2 metric 10 for interface PE-3 <=> PE-5. Written more concisely:

```
Prefix 192.0.2.5/32 :  SP(PE-2,PE-5) < SP(PE-2,PE-3) + SP(PE-3,PE-5)
                      10 + 10      <    30 + 10                                => OK
```

For completion, all the other Inequality 1 calculations on PE-1 are given:

```
Prefix 192.0.2.2/32 :  SP(PE-3,PE-2) < SP(PE-3,PE-1) + SP(PE-1,PE-2)
                      30      <    10 + 10                                => NOK
Prefix 192.0.2.3/32 :  SP(PE-2,PE-3) < SP(PE-2,PE-1) + SP(PE-1,PE-3)
                      30      <    10 + 10                                => NOK
Prefix 192.0.2.4/32 :  SP(PE-3,PE-4) < SP(PE-3,PE-1) + SP(PE-1,PE-2)
                      10      <    10 + 10                                => OK
Prefix 192.168.23.0/30 : SP(PE-3,D) < SP(PE-3,PE-1) + SP(PE-1,D)
                      30      <    10 + 10                                => NOK
Prefix 192.168.24.0/30 : SP(PE-3,D) < SP(PE-3,PE-1) + SP(PE-1,D)
                      30 + 10 <    10 + (10 + 10)                        => NOK
Prefix 192.168.34.0/30 : SP(PE-2,D) < SP(PE-2,PE-1) + SP(PE-1,D)
                      30 + 10 <    10 + (10 + 10)                        => NOK
Prefix 192.168.35.0/30 : SP(PE-2,D) < SP(PE-2,PE-1) + SP(PE-1,D)
                      30 + 10 <    10 + (10 + 10)                        => NOK
Prefix 192.168.45.0/30 : SP(PE-3,D) < SP(PE-3,PE-1) + SP(PE-1,D)
                      10 + 10 <    10 + (10 + 10 + 10)                  => OK
```

As shown, only three are valid. On the 7x50, a summary command exists for LFA coverage on the router:

```
*A:PE-1# show router isis lfa-coverage
=====
LFA Coverage
=====
Topology      Level  Node      IPv4      IPv6
-----
IPV4 Unicast  L1     0/0 (0%)  3/9 (33%) 0/0 (0%)
IPV6 Unicast  L1     0/0 (0%)  0/0 (0%)  0/0 (0%)
IPV4 Multicast L1     0/0 (0%)  0/0 (0%)  0/0 (0%)
IPV6 Multicast L1     0/0 (0%)  0/0 (0%)  0/0 (0%)
IPV4 Unicast  L2     2/4 (50%) 3/9 (33%) 0/0 (0%)
IPV6 Unicast  L2     0/0 (0%)  0/0 (0%)  0/0 (0%)
IPV4 Multicast L2     0/0 (0%)  0/0 (0%)  0/0 (0%)
IPV6 Multicast L2     0/0 (0%)  0/0 (0%)  0/0 (0%)
=====
*A:PE-1#
```

IS-IS Overload Bit

Ensure the network is back to its initial topology with all the level 2 metrics back to 10 before applying the changes referenced below.

As stated in RFC 3137, *OSPF Stub Router Advertisement*, sometimes it is desirable not to have a router used as a transit node. For those cases, it is also desirable not to have that router used as transit node during the LFA next-hop computation. Within IS-IS protocol this is achieved by setting the overload bit. When other routers detect that this bit is set, they will only use this router for packets destined to the overloaded router's directly connected networks and IP prefixes.

As an example, setting of the IS-IS overload condition for a specific time on PE-2 provides following result on PE-1:

```
*A:PE-2# configure router isis overload
- no overload
- overload [timeout <seconds>] [max-metric]

<seconds>          : [60..1800]

*A:PE-1# show router isis lfa-coverage
=====
LFA Coverage
=====
Topology          Level   Node          IPv4           IPv6
-----
IPV4 Unicast      L1      0/0 (0%)      3/9 (33%)      0/0 (0%)
IPV6 Unicast      L1      0/0 (0%)      0/0 (0%)      0/0 (0%)
IPV4 Multicast    L1      0/0 (0%)      0/0 (0%)      0/0 (0%)
IPV6 Multicast    L1      0/0 (0%)      0/0 (0%)      0/0 (0%)
IPV4 Unicast      L2      1/4 (25%)      3/9 (33%)      0/0 (0%)
IPV6 Unicast      L2      0/0 (0%)      0/0 (0%)      0/0 (0%)
IPV4 Multicast    L2      0/0 (0%)      0/0 (0%)      0/0 (0%)
IPV6 Multicast    L2      0/0 (0%)      0/0 (0%)      0/0 (0%)
=====

*A:PE-1#
*A:PE-1# show router route-table alternative
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]          Type   Proto   Age           Pref
Next Hop[Interface Name]    Metric
Alt-NextHop                 Alt-
Metric
-----
192.0.2.1/32                Local  Local   07d23h49m    0
system                      0
192.0.2.2/32                Remote ISIS   07d23h48m    18
192.168.12.2                10
192.168.13.2 (LFA)         20
192.0.2.3/32                Remote ISIS   00h01m59s    18
192.168.13.2                10
192.0.2.4/32                Remote ISIS   00h00m40s    18
192.168.13.2                20
```

Additional Topics

```

192.0.2.5/32                               Remote  ISIS      00h01m59s  18
      192.168.13.2                          20
192.168.12.0/30                             Local   Local    07d23h48m  0
      int-PE-1-PE-2                          0
192.168.13.0/30                             Local   Local    07d23h48m  0
      int-PE-1-PE-3                          0
192.168.23.0/30                             Remote  ISIS      00h01m11s  18
      192.168.12.2                             20
      192.168.13.2 (LFA)                         30
192.168.24.0/30                             Remote  ISIS      07d23h48m  18
      192.168.12.2                             20
      192.168.13.2 (LFA)                         30
192.168.34.0/30                             Remote  ISIS      00h01m59s  18
      192.168.13.2                          20
192.168.35.0/30                             Remote  ISIS      00h01m59s  18
      192.168.13.2                          20
192.168.45.0/30                             Remote  ISIS      00h00m40s  18
      192.168.13.2                          25
-----
No. of Routes: 12
Flags: n = Number of times nexthop is repeated
      Backup = BGP backup route
      LFA = Loop-Free Alternate nexthop
      S = Sticky ECMP requested
=====
*A:PE-1#

*A:PE-1# show router isis routes alternative
=====
Route Table
=====
Prefix[Flags]           Metric    Lvl/Typ    Ver.  SysID/Hostname
NextHop                MT          AdminTag
Alt-Nexthop            Alt-      Alt-Type
                        Metric
-----
192.0.2.1/32             0          2/Int.     3     PE-1
      0.0.0.0             0          0          0
192.0.2.2/32           10        2/Int.    5     PE-2
      192.168.12.2           0          0
      192.168.13.2 (L)       20        LP
192.0.2.3/32            10         2/Int.     43    PE-3
      192.168.13.2          0          0
192.0.2.4/32            20         2/Int.     47    PE-3
      192.168.13.2          0          0
192.0.2.5/32            20         2/Int.     43    PE-3
      192.168.13.2          0          0
192.168.12.0/30         10         2/Int.     4     PE-1
      0.0.0.0             0          0
192.168.13.0/30         10         2/Int.     43    PE-1
      0.0.0.0             0          0
192.168.23.0/30       20        2/Int.    45    PE-2
      192.168.12.2           0          0
      192.168.13.2 (L)       30        LP
192.168.24.0/30       20        2/Int.    5     PE-2
      192.168.12.2           0          0
      192.168.13.2 (L)       30        LP
192.168.34.0/30         20         2/Int.     43    PE-3

```

```

192.168.13.2                                0      0
192.168.35.0/30                             20      2/Int.  43    PE-3
192.168.13.2                                0      0
192.168.45.0/30                             25      2/Int.  47    PE-3
192.168.13.2                                0      0
-----
No. of Routes: 12
Flags: L = Loop-Free Alternate nexthop
Legend: LP = linkProtection, NP = nodeProtection
=====
*A:PE-1#

```

On PE-1, only three Inequality 1 calculations are possible. Refer to the previous show commands.

$$\begin{aligned} \text{SP}(\text{backup NHR}, D) &< \{ \text{SP}(\text{backup NHR}, S) + \text{SP}(S, D) \} \\ \text{SP}(\text{PE-3}, D) &< \text{SP}(\text{PE-3}, \text{PE-1}) + \text{SP}(\text{PE-1}, D) \\ 10 + 10 &< 10 + (10 + 10) \quad \Rightarrow \text{OK} \end{aligned}$$

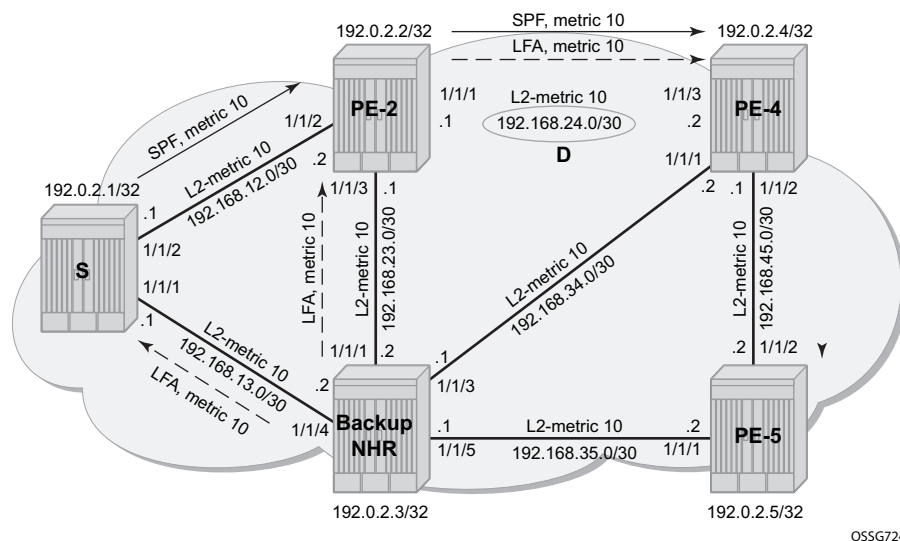


Figure 101: IS-IS Overload on PE-2, Inequality 1 for 192.168.24.0/30 (D) on PE-1 (S)

Conclusion

In production MPLS networks where FRR needs to be deployed, a trade off must be made between RSVP-TE FRR versus LDP FRR. The two main advantages of using LDP FRR compared to RSVP FRR are simple configuration and LFA next-hop calculation is a local decision, which means no interoperability issues when working in a multi-vendor environment. The main disadvantage of using LDP FRR is that LFA next-hop calculation has to deal with source-route paradigm (inequality formulas exclude a path going over the original source router).

MPLS Transport Profile

In This Chapter

This section provides information about Multiprotocol Label Switching Transport Profile (MPLS-TP).

Topics in this section include:

- [Applicability on page 590](#)
- [Summary on page 591](#)
- [Overview on page 592](#)
- [Configuration on page 594](#)
- [Conclusion on page 616](#)

Applicability

This example is applicable to the 7950, 7750 and 7450 series and was tested on release 13.0.R2. Multiprotocol Label Switching Transport Profile (MPLS-TP) requires a minimum of FP2 (Flex Path 2) or higher based hardware. A Control Processing Module (CPM) 3 or higher is required for the highest Bi-directional Forwarding Detection (BFD) scale using 10ms control packet timers.

This example assumes that the reader is familiar with the configuration of IP/MPLS and Virtual Leased Line (VLL) services on the 7x50.

MPLS-TP was first introduced in SR OS release 11.0.R4 and further enhancements were added in subsequent releases.

Summary

MPLS-TP is intended to allow MPLS to be operated in a similar manner to existing transport technologies, with static configuration of transport paths (particularly with no requirement for a dynamic control plane), in-band proactive and on-demand operations and maintenance (OAM), and protection mechanisms that do not rely on a control plane (for example, Resource Reservation Protocol with Traffic Engineering (RSVP-TE)) to operate. The 7x50 can operate both as a Label Edge Router (LER) and Label Switching Router (LSR) for MPLS-TP LSPs, and as a Terminating Provider Edge (T-PE) and Switching Provider Edge (S-PE) for Pseudowires (PWs) with MPLS-TP OAM. The 7x50 can therefore act as a node within an MPLS-TP network, or as a gateway between MPLS-TP and IP/MPLS domains.

Overview

MPLS can provide a network layer with packet transport services. In some operational environments it is desirable that the operation and maintenance of such an MPLS based packet transport network follows the operational models typically used in traditional optical transport networks (for example with SONET, SDH) while providing additional OAM, survivability and other maintenance functions targeted at that environment.

MPLS-TP defines a profile of MPLS targeted at transport applications. This profile defines the specific MPLS characteristics and extensions required to meet transport requirements, while retaining compliance with the standard IETF MPLS architecture and label-switching paradigm. The basic architecture and requirements for MPLS-TP are described by the IETF in RFC 5654, RFC 5921 and RFC 5960, in order to meet two objectives:

- To enable MPLS to be deployed in a transport network and operated in a similar manner to existing transport technologies.
- To enable MPLS to support packet transport services with a similar degree of predictability to that found in existing transport networks.

In order to meet these objectives, MPLS-TP has a number of high-level characteristics:

- MPLS-TP, including resilience and protection, operates in the absence of an IP control plane and IP. MPLS-TP does not modify the MPLS forwarding architecture, which is based on existing pseudowire and LSP constructs. Point-to-point LSPs may be unidirectional or bi-directional. Bi-directional LSPs must be congruent (i.e. co-routed and follow the same path in each direction) and are the only supported type on the 7x50. MPLS-TP is only supported on static LSPs and pseudowires (PWs). Also, there is no LSP merging.
- LSP and pseudowire monitoring is achieved using in-band OAM and does not rely on control plane or IP routing functions to determine the health of a path, for example, LDP hello failures do not trigger protection.

The system supports MPLS-TP on LSPs and PWs with static labels. MPLS-TP is not supported on dynamically signaled LSPs and PWs, although switching a static MPLS-TP PW to a targeted LDP (T-LDP) signaled PW is supported. MPLS-TP is supported for Epipe, Apipe and Cpipe VLLs, and Epipe spoke SDP termination on IES, VPRN and VPLS. Static PWs may use SDPs on top of either static MPLS-TP LSPs or RSVP-TE LSPs.

The following MPLS-TP OAM and protection mechanisms defined by the IETF are supported:

- MPLS-TP Generic Associated Channel for LSPs and PWs (RFC 5586)
- MPLS-TP Identifiers (RFC 6370)
- Proactive Continuity Check (CC), Connectivity Verification (CV), and Remote Defect Indicator (RDI) using Bi-directional Forwarding Detection (BFD) for LSPs (RFC 6428)

- On-Demand CV for LSPs and PWs using LSP Ping and LSP Trace (RFC 6426)
- 1-for-1 Linear protection for LSPs (RFC 6378)
- Static PW Status Signaling (RFC 6478)

The system can play the role of an LER and an LSR for static MPLS-TP LSPs, and a PE/T-PE and an S-PE for static MPLS-TP PWs. It can also act as an S-PE for MPLS-TP segments between an MPLS network that strictly follows the transport profile and an MPLS network that supports both MPLS-TP and dynamic IP/MPLS.

Configuration

This section details the configuration steps for a set of simple MPLS-TP examples.

The following reference network is used (Figure 102). It consists of four nodes and two Epipe VLL services. One service is transported across a network domain consisting of only static MPLS-TP LSPs (Epipe 10) from PE-1 to PE-2. The other Epipe (Epipe 20) is used to transport traffic from PE-1 in the MPLS-TP domain to a VPLS service on PE-4 in an IP/MPLS domain. A static MPLS-TP LSP exists between PE-1 and PE-2, while a dynamic RSVP-TE LSP exists between PE-2 and PE-4.

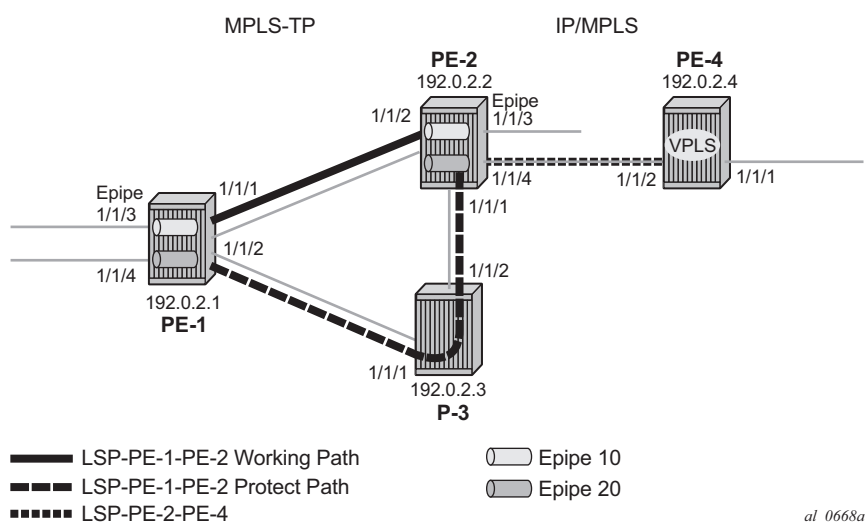


Figure 102: MPLS-TP Example Network Showing LSPs

shows further details of the logical architecture of the services in the example network. The Epipe spoke-sdps use the static MPLS-TP transport LSP between PE-1 and PE-2, and the dynamically signaled RSVP-TE LSP between PE-2 and PE-4. The MPLS-TP LSP is protected using 1:1 linear protection, with a working path from PE-1 to PE-2, and a protect path from PE-1, through LSR P-3, to PE-2. The Ethernet PW for Epipe 10 connects an Ethernet SAP on port 1/1/3 on PE-1 to an Ethernet SAP on port 1/1/3 on PE-2. The PW for Epipe 20 connects an Ethernet SAP on port 1/1/4 on PE-1 to the VPLS on PE-4 and is switched between a static MPLS-TP segment and a dynamic targeted LDP (T-LDP) segment at PE-2. PE-2 thus acts as a gateway between the MPLS-TP domain and the IP/MPLS domain.

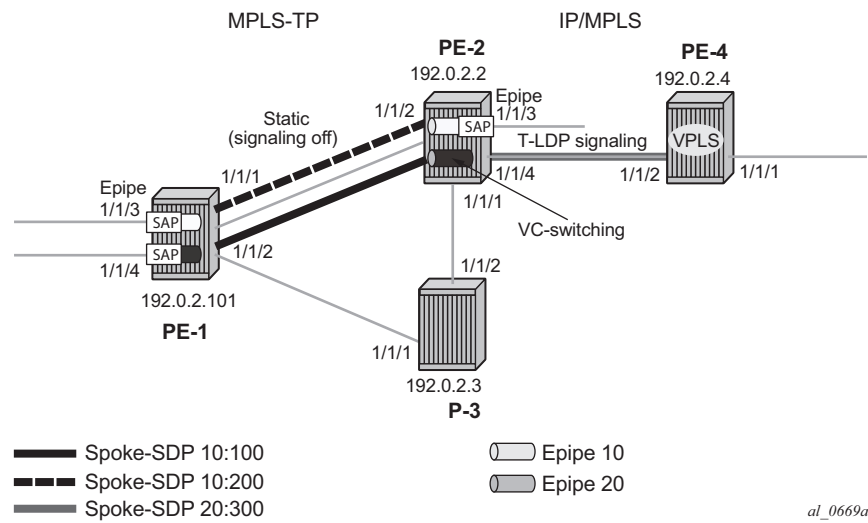


Figure 103: MPLS-TP Example Network Showing Services Detail

Figure 104 shows the configuration process to be followed when setting up MPLS-TP.

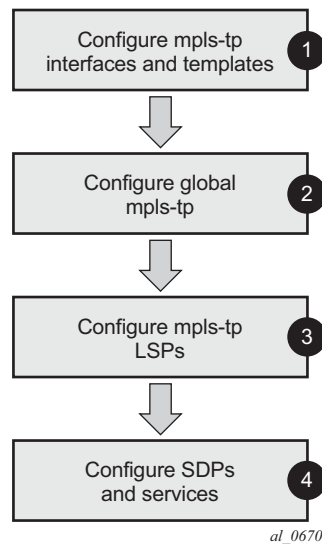


Figure 104: MPLS-TP Configuration Steps

Step 1. Configuration of MPLS-TP interfaces and templates.

MPLS-TP LSPs can use either numbered or unnumbered network IP interfaces, or unnumbered network interfaces that have been configured to operate without relying on IP routing. This non-IP interface type does not have an IP address associated with it and may be configured to have either a unicast, broadcast or multicast MAC address. The intent of using a broadcast or multicast MAC address is to enable a standard set of MAC addresses to be configured for a network without requiring any changes to the configuration of neighboring router interfaces each time an interface to which a router is connected is changed. Note that if a broadcast or multicast MAC address is used, then the operator should take care that only a point-to-point link is connected to the Ethernet port used by the interface. Otherwise, MPLS-TP packets may be replicated to each remote port to which the link is connected.

The non-IP network interface type is known as an unnumbered-mpls-tp interface. Only MPLS-TP can use this interface type. That is, other IP protocols are blocked from using it. Furthermore, ARP is not used for next hop resolution. This example uses unnumbered-mpls-tp interfaces.

Unnumbered MPLS-TP interfaces are configured on each network-facing interface for the nodes in the MPLS-TP domain, as shown below. This is done using the **unnumbered-mpls-tp** keyword at create time. In addition, the **static-arp unnumbered** command is used to set the next-hop unicast, broadcast or multicast MAC address of the interface. The system interface should also be configured. Numbered IP Network interfaces, bound to port 1/1/4 of PE-2 and port 1/1/2 of PE-4 are used for the IP/MPLS portion of the network in [Figure 102](#).

```
*A:PE-1# configure router
      interface "int-PE-1-P-3" unnumbered-mpls-tp
        port 1/1/2
        static-arp unnumbered 01:00:5e:90:00:00
        no shutdown
      exit
      interface "int-PE-1-PE-2" unnumbered-mpls-tp
        port 1/1/1
        static-arp unnumbered 01:00:5e:90:00:00
        no shutdown
      exit
      interface "system"
        address 192.0.2.1/32
      exit
      autonomous-system 64511

*A:PE-2# configure router
      interface "int-PE-2-P-3" unnumbered-mpls-tp
        port 1/1/1
        static-arp unnumbered 01:00:5e:90:00:00
        no shutdown
      exit
      interface "int-PE-2-PE-1" unnumbered-mpls-tp
        port 1/1/2
        static-arp unnumbered 01:00:5e:90:00:00
        no shutdown
      exit
      interface "int-PE-2-PE-4"
```



```

        address 192.168.24.1/30
        port 1/1/4
    exit
    interface "system"
        address 192.0.2.2/32
    exit
    autonomous-system 65535
    static-route 192.0.2.4/32 next-hop 192.168.24.2

*A:P-3# configure router
    interface "int-P-3-PE-1" unnumbered-mpls-tp
        port 1/1/1
        static-arp unnumbered 01:00:5e:90:00:00
        no shutdown
    exit
    interface "int-P-3-PE-2" unnumbered-mpls-tp
        port 1/1/2
        static-arp unnumbered 01:00:5e:90:00:00
        no shutdown
    exit
    interface "system"
        address 192.0.2.3/32
    exit
    autonomous-system 65535

*A:PE-4# configure router
    interface "int-PE-4-PE-2"
        address 192.168.24.2/30
        port 1/1/2
    exit
    interface "system"
        address 192.0.2.4/32
    exit
    autonomous-system 65535
    static-route 192.0.2.2/32 next-hop 192.168.24.1

```

Next, MPLS should be configured on each of the interfaces to be used by MPLS-TP. As an example, only PE-1 configuration is shown although a similar configuration is provisioned on PE-2 and P-3.

```

*A:PE-1# configure router
    mpls
        mpls-tp
        exit
        interface "system"
        exit
        interface "int-PE-1-PE-2"
        exit
        interface "int-PE-1-P-3"
        exit
        no shutdown
    exit

```

PE-4 is an IP/MPLS only node so there is no MPLS TP configuration

```
*A:PE-4# configure router
      mpls
        interface "system"
        exit
        interface "int-PE-4-PE-2"
        exit
        no shutdown
      exit
```

Note that the **mpls** context must be in the **no shutdown** state to enable MPLS-TP.

Static labels are used by MPLS-TP LSPs and PWs. SR OS requires that a user reserves a range from the global label space for static labels. This prevents the labels being used by signaling protocols, such as RSVP. Static labels are reserved as shown by the following CLI command. As an example, the lower 200 labels (from 32, onwards) are reserved for static allocation to LSPs or PWs. This configuration should be repeated for every node that is an LER or LSR for MPLS-TP LSPs, although only the configuration for PE-1 is displayed.

```
*A:PE-1# configure router
      mpls-labels
        static-label-range 200
      exit
```

Next, one or more Bidirectional Forwarding Detection (BFD) templates are configured on the LERs. These templates are used to define BFD state machine parameters used for BFD Continuity Check (CC) on an LSP, including the transmit and receive timer intervals (in milliseconds). CPM network processor BFD is required if timer intervals as short as 10ms are used, but depending on the platform, 100ms BFD may use CPU based BFD (as shown in the example here).

```
*A:PE-1# configure router bfd
- bfd

      abort          - Discard the changes that have been made to bfd template during a
                      session
      begin          - Switch to edit mode for bfd template - use commit to save or abort
                      to discard the changes made in a session
[no] bfd-template    + Configure a bfd template
      commit         - Save the changes made to bfd template during a session
```

```
*A:PE-1# configure router bfd bfd-template
- bfd-template <[32 chars max]>
- no bfd-template <[32 chars max]>
```

```
[no] echo-receive    - Configure echo receive interval
[no] multiplier      - Configure multiplier
[no] receive-interv* - Configure receive interval
[no] transmit-inter* - Configure transmit interval
[no] type            - Configure the bfd session endpoint type
```

```
*A:PE-1#
```

A subset of these parameters is used by MPLS-TP BFD sessions, as follows:

- **transmit-interval** *transmit-interval* and the **receive-interval** *receive-interval* — These are the transmit and receive timers for BFD packets. For MPLS-TP, these are the timers used by BFD CC packets. Values are in milliseconds: 10ms to 100,000ms, with 1ms granularity. Default 10ms for CPM3 or higher, 1 sec for other hardware. The minimum interval that can be supported is hardware dependent. For MPLS-TP BFD Connectivity Verification (CV) packets, a transmit interval of 1 sec is always used.
- **multiplier** *multiplier* — Integer 3 – 20. Default: 3. The configured parameter is used for MPLS-TP CC BFD sessions. It is ignored for MPLS-TP combined CC/CV BFD sessions, and the default of 3 is used.
- **type cpm-np** — This selects the CPM network processor as the local termination point for the BFD session. This is used by default for MPLS-TP. The CPM-NP type is needed to configure a transmit interval down to 10ms.

The following CLI illustrates the BFD template configuration at PE-1. Since default parameters are sufficient, only the bfd-template name is configured. Note that BFD templates use a begin/commit model for configuration. Create or modify a template with the **begin** statement. Changes to an existing template or the creation of a new template is not effected until the **commit** statement is entered.

```
*A:PE-1# configure router
      bfd
        begin
        bfd-template "tp-bfd"
        exit
        commit
      exit
```

The following **info detail** command shows the values that are assigned by default.

```
*A:PE-1>config>router>bfd# info detail
-----
      bfd-template "tp-bfd"
        no type
        transmit-interval 100
        receive-interval 100
        multiplier 3
        echo-receive 100
      exit
-----
```

Step 2. Configuration of Global MPLS-TP Parameters

MPLS-TP global parameters are configured under **config>router>mpls>mpls-tp**. These include the MPLS-TP identifiers for the node and the range of tunnel identifiers that should be reserved for MPLS-TP LSPs.

Node identifiers include the Global ID and the Node ID. The Node ID may be defined as an unsigned integer or use dotted quad notation (a.b.c.d), but the Node ID does not have to be a routable IP address.

The CLI tree for configuring the MPLS-TP identifiers for a node is as follows:

```
*A:PE-1# configure router
      mpls
        mpls-tp
          - mpls-tp
          - no mpls-tp

[no] global-id      - Global id for MPLS-TP
[no] node-id        - Node id for MPLS-TP local router
[no] oam-template   + Configure a MPLS-TP OAM Template
[no] protection-tem* + Configure a MPLS-TP Protection Template
[no] shutdown       - Administratively enable/disable the MPLS-TP
[no] tp-tunnel-id-r* - Configure MPLS-TP tunnel id range on the ingress router
[no] transit-path    + Configure a MPLS-TP Transit Path
```

The default value for the global-id is 0. This is used if the global-id is not configured. If an operator expects that inter-domain LSPs will be configured, then it is recommended to set the global ID to the local autonomous system number (ASN) of the node, as configured under **config>router**, to ensure that the combination of global-id and node-id is globally unique. If two-byte ASNs are used, then the two most significant bytes of the global-id are padded with zeros.

The default value of the **node-id** is the system interface IPv4 address. The MPLS-TP context cannot be administratively enabled unless at least a system interface IPv4 address is configured because MPLS requires that this value be configured.

In order to change the values, **config>router>mpls>mpls-tp** must be in the shutdown state. This will bring down all of the MPLS-TP LSPs on the node. New values are propagated to the system when a **no shutdown** is performed.

The following CLI shows the MPLS-TP node identifier configuration for PE-1. A similar configuration is implemented in all routers in this example, except that the node-ids must be different (PE-2 is 10.0.0. 2 and P-3 is 10.0.0. 3). In this example the global-id for PE-2 and P-3 equals 65535.

```
*A:PE-1# configure router
      mpls
        mpls-tp
          global-id 64511
          node-id 10.0.0.1
```

```
*A:PE-2# configure router
      mpls
        mpls-tp
          global-id 65535
          node-id 10.0.0.2
```

Next, protection and OAM templates should be configured at the MPLS-TP LERs. A protection template defines the parameters for the linear protection state coordination mechanism. MPLS-TP Linear Protection is specified in RFC6378. It provides protection for an LSP using a working and a protect path. A Protection State Coordination (PSC) protocol is used by the LERs at each end of the protected LSP to coordinate whether the working or protect path is used for forwarding. BFD is run on both the working and protect paths.

The linear protection parameters include revertive or non-revertive behavior, the wait-to-restore timer, the rapid-psc-timer and the slow-psc-timer. The wait-to-restore timer (in seconds) defines the time to wait before reverting to the working path if, on restoration of connectivity, the revertive behavior is selected.

The following CLI tree is used to configure the protection template:

```
*A:PE-1# configure router
      mpls
        mpls-tp
          protection-template
            - no protection-template <[32 chars max]>
            - protection-template <[32 chars max]>

            rapid-psc-timer - Configure the rapid Protection Switch Coordination (PSC) timer
            [no] revertive   - Enable/Disable the template's revertive mode
            slow-psc-timer  - Configure the slow Protection Switch Coordination (PSC) timer
            [no] wait-to-restore - Configure the WTR timer for the template
```

Refer to the CLI command descriptions in the MPLS User Guide for further details of these commands.

The OAM template defines generic proactive OAM parameters, such as BFD hold down and hold up timer values (which can be used to introduce some hysteresis if BFD bounces) and the BFD template to use.

The following CLI tree is used to configure the OAM template:

```
*A:PE-1# configure router
      mpls
        mpls-tp
          oam-template
            - no oam-template <template-name>
            - oam-template <template-name>

            <template-name>      : [32 chars max]
```

Configuration

```
[no] bfd-template      - Configure the Bidirectional Forwarding Detection (BFD) template
[no] hold-time-down    - Configure hold-down dampening timer
[no] hold-time-up      - Configure the hold-up dampening timer
```

Refer to the CLI command descriptions in the MPLS User Guide for further details of these commands.

MPLS-TP requires the reservation of a tunnel ID range, dedicated for the use of MPLS-TP LSPs. This range is reserved using the following CLI tree:

```
*A:PE-1# configure router
      mpls
        mpls-tp
          tp-tunnel-id-range
            - tp-tunnel-id-range <min> <max>
            - no tp-tunnel-id-range

<min>          : [1..61440]
<max>          : [1..61440]
```

The default parameter values are used as shown below, where PE-1 and PE-2 have the same configuration:

```
*A:PE-1# configure router mpls
      mpls-tp
        tp-tunnel-id-range 100 1000
        protection-template "tp-protect"
        exit
        oam-template "tp-oam"
          bfd-template "tp-bfd"
        exit
        no shutdown
      exit
```

Step 3. Configuration of MPLS-TP LSPs

Once the global MPLS-TP parameters have been configured, the system is ready to configure MPLS-TP LSPs. An MPLS-TP LSP is configured under the **config>router>mpls>lsp** context.

Note that because LSP labels are statically configured, both ends of the LSP must be explicitly configured. The LSP paths must also be explicitly configured in the LSR nodes. MPLS-TP LSPs must use the **mpls-tp** keyword including a source tunnel number at creation time.

The following commands are used to configure an MPLS-TP LSP at an LER:

```
configure
router
  mpls
    lsp <lsp-name> mpls-tp <src-tunnel-num>
      to node-id {<a.b.c.d> | <1.. .4,294,967,295>}
      dest-global-id <global-id>
      dest-tunnel-number <tunnel-num>
      [no] working-tp-path
        lsp-num <lsp-num>
        in-label <in-label>
        out-label <out-label> out-link <if-name> [next-hop <ipv4-address>]
        [no] mep
          [no] oam-template <name>
          [no] bfd-enable [cc | cc-cv]
          [no] shutdown
        exit
      [no] shutdown
    exit
    [no] protect-tp-path
      lsp-num <lsp-num>
      in-label <in-label>
      out-label <out-label> out-link <if-name> [next-hop <ipv4-address> ]
      [no] mep
        [no] protection-template <name>
        [no] oam-template <name>
        [no] bfd-enable [cc | cc-cv]
        [no] shutdown
      exit
    [no] shutdown
  exit
```

Refer to the CLI command descriptions in the MPLS User Guide for further details of these commands.

A working path and a protect path for LSP LSP-PE-1-P-2 must be configured between PE-1 and PE-2. Each LSP is configured with the full set of MPLS-TP identifiers required to build the LSP ID. Each working path and protect path must have an incoming label, outgoing label and outgoing link configured.

Each working path and protect path also includes a Maintenance Entity Group Endpoint (MEP) configuration, under which the applicable OAM template is configured. BFD is also enabled under the MEP context for the path. In this example, BFD operating in CC mode is enabled on the

working and protect paths. Note that the Protection Template, containing parameters for linear protection, is only applied under the protect path context.

Figure 105 shows the LSP working and protect path label values configured at PE-1, PE-2 and P-3. Note that at each node the outgoing label must match the incoming label on the next hop for a given direction. At the LERs (PE-1 and PE-2), the incoming and outgoing label values for each LSP path are configured together. At the LSR (P-3), the label values for the label mapping between ingress and egress for each direction of the path (that is, forward and reverse) are configured together.

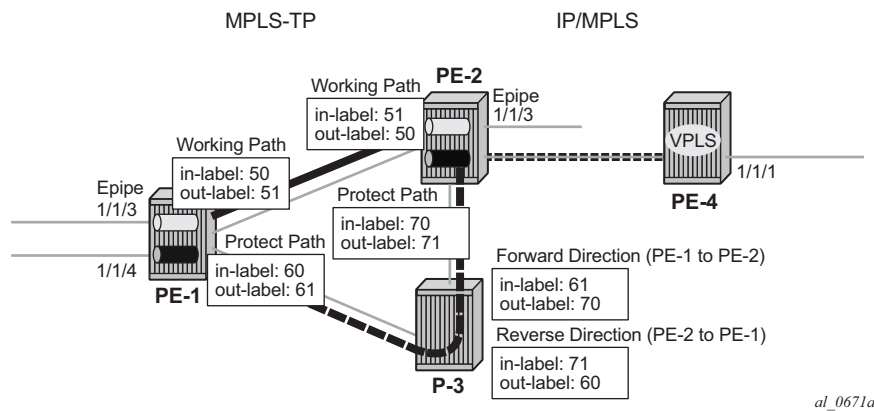


Figure 105: LSP Path Label Value Configurations

The following shows the LER LSP configuration of PE-1 and PE-2.

```
*A:PE-1# configure router
mpls
  lsp "LSP-PE-1-PE-2" mpls-tp 100
    to node-id 10.0.0.2
    dest-global-id 65535
    dest-tunnel-number 100
    working-tp-path
      in-label 50
      out-label 51 out-link "int-PE-1-PE-2"
      mep
        oam-template "tp-oam"
        bfd-enable cc
        no shutdown
      exit
    no shutdown
  exit
  protect-tp-path
    in-label 60
    out-label 61 out-link "int-PE-1-P-3"
    mep
      protection-template "tp-protect"
```



```

        oam-template "tp-oam"
        bfd-enable cc
        no shutdown
    exit
    no shutdown
exit
no shutdown
exit
no shutdown
exit
no shutdown
exit

*A:PE-2# configure router
mpls
  lsp "LSP-PE-1-PE-2" mpls-tp 100
    to node-id 10.0.0.1
    dest-global-id 64511
    dest-tunnel-number 100
    working-tp-path
      in-label 51
      out-label 50 out-link "int-PE-2-PE-1"
      mep
        oam-template "tp-oam"
        bfd-enable cc
        no shutdown
      exit
      no shutdown
    exit
  protect-tp-path
    in-label 70
    out-label 71 out-link "int-PE-2-P-3"
    mep
      protection-template "tp-protect"
      oam-template "tp-oam"
      bfd-enable cc
      no shutdown
    exit
    no shutdown
  exit
  no shutdown
exit
no shutdown
exit

```

Since this example requires a protect path to be switched via P-3, a transit path must be configured in P-3. The CLI tree for configuring MPLS-TP transit paths is as follows:

```

config
  router
    mpls
      mpls-tp
        transit-path <path-name>
          [no] path-id {lsp-num <lsp-num>|working-path|protect-path

              [src-global-id <global-id>]
              src-node-id {<ipv4address> | <1.. .4,294,967,295>}
              src-tunnel-num <tunnel-num>

```

Configuration

```
[dest-global-id <global-id>]
dest-node-id {<ipv4address> | <1.. .4,294,967,295>}
[dest-tunnel-num <tunnel-num>]}

forward-path
  in-label <in-label> out-label <out-label>
  out-link <if-name> [next-hop <ipv4-next-hop>]
  exit
reverse-path
  in-label <in-label> out-label <out-label>
  [out-link <if-name> [next-hop <ipv4-next-hop>]]
  exit
[no] shutdown
```

Refer to the CLI command descriptions in the MPLS User Guide for further details of these commands.

The CLI configuration for the forward and reverse directions of the transit path (that is, the protect path of the LSP) at P-3 is as follows:

```
*A:P-3# configure router
mpls
  mpls-tp
    transit-path "LSP-PE-1-PE-2"
      forward-path
        in-label 61 out-label 70 out-link "int-P-3-PE-2"
      exit
      reverse-path
        in-label 71 out-label 60 out-link "int-P-3-PE-1"
      exit
      path-id src-global-id 64511 src-node-id 10.0.0.1 src-tunnel-num 100
      dest-global-id 65535 dest-node-id 10.0.0.2 dest-tunnel-num 100 lsp-num
      2
      no shutdown
    exit
  no shutdown
exit
no shutdown
exit
```

The example also requires an LSP across the IP/MPLS network to backhaul traffic from PE-2 at the edge of the MPLS-TP network to the VPLS service hosted in PE-4. An RSVP LSP is configured at PE-2 for this purpose, as follows:

```
*A:PE-2# configure router
mpls
  path "loose"
    no shutdown
  exit
  lsp "LSP-PE-2-PE-4"
    to 192.0.2.4
    primary "loose"
    exit
    no shutdown
  exit
```

```
*A:PE-2# configure router rsvp no shutdown
```

Create a T-LDP session towards PE-4. LDP over RSVP is preferred (prefer-tunnel-in-tunnel).

```
*A:PE-2# configure router
      ldp
        prefer-tunnel-in-tunnel
        interface-parameters
        exit
        targeted-session
          peer 192.0.2.4
          exit
        exit
      exit
```

A similar configuration is implemented in PE-4.

At this point in the configuration process, it is recommended to check the MPLS-TP LSP configuration and operation of BFD and linear protection.

First, check that the BFD sessions on both the working and protect paths are up:

```
*A:PE-1# show router bfd session
=====
Legend:  wp = Working path    pp = Protecting path
=====
BFD Session
=====
If/Lsp Name/Svc-Id          State          Tx Intvl  Rx Intvl  Multipl
  Rem Addr/Info/SdpId:VcId  Protocols      Tx Pkts   Rx Pkts   Type
  LAG port                  LAG ID
-----
wp::LSP-PE-1-PE-2          Up              100        100        3
   65535::10.0.0.2          mplstP         92177      92138     central
pp::LSP-PE-1-PE-2          Up              100        100        3
   65535::10.0.0.2          mplstP         92217      92179     central
-----
No. of BFD sessions: 2
=====
*A:PE-1#
```

Next, check the currently active path. This can be done using the `oam lsp-trace` command. Note that the static option must be specified for MPLS-TP LSPs.

```
*A:PE-1# oam lsp-trace static "LSP-PE-1-PE-2"
lsp-trace to LSP-PE-1-PE-2: 0 hops min, 0 hops max, 100 byte packets
1 GlobalId 65535 NodeId 10.0.0.2
   rtt=0.564ms rc=3(EgressRtr)
*A:PE-1#
```

This shows that data packets currently follow the working path of the LSP (no transit node is shown).

Configuration

In order to test the operation of linear protection, the port used by the working path can be shutdown, and the BFD session state checked again:

```
*A:PE-1# configure port 1/1/1 shutdown

*A:PE-1# show router bfd session
=====
Legend:  wp = Working path   pp = Protecting path
=====
BFD Session
=====
If/Lsp Name/Svc-Id          State          Tx Intvl  Rx Intvl  Multipl
  Rem Addr/Info/SdpId:VcId  Protocols      Tx Pkts   Rx Pkts   Type
  LAG port                  LAG ID
-----
wp::LSP-PE-1-PE-2          Down           1000      100       3
  65535::10.0.0.2          mplsTp        93952     93907     central
pp::LSP-PE-1-PE-2          Up             100       100       3
  65535::10.0.0.2          mplsTp        94043     94005     central
-----
No. of BFD sessions: 2
=====
*A:PE-1#
```

Execute LSP trace again to check that the LSP has failed over to use the protect path:

```
*A:PE-1# oam lsp-trace static "LSP-PE-1-PE-2"
lsp-trace to LSP-PE-1-PE-2: 0 hops min, 0 hops max, 100 byte packets
1 GlobalId 65535 NodeId 10.0.0.3
  rtt=0.585ms rc=8(DSRtrMatchLabel)
2 GlobalId 65535 NodeId 10.0.0.2
  rtt=1.07ms rc=3(EgressRtr)
*A:PE-1#
```

This shows that packets are now forwarded via the protect path through P-3, which has Node ID 10.0.0.3.

Finally bring the LSP back to the working path by bringing port 1/1/1 up, and either waiting for the LSP to revert to the working path or forcing it onto the working path and clearing the revert timer by executing a tools command as follows:

```
*A:PE-1# configure port 1/1/1 no shutdown

*A:PE-1# tools perform router mpls tp-tunnel force "LSP-PE-1-PE-2"

*A:PE-1# tools perform router mpls tp-tunnel clear "LSP-PE-1-PE-2"

*A:PE-1# oam lsp-trace static "LSP-PE-1-PE-2"
lsp-trace to LSP-PE-1-PE-2: 0 hops min, 0 hops max, 100 byte packets
1 GlobalId 65535 NodeId 10.0.0.2
  rtt=0.582ms rc=3(EgressRtr)
*A:PE-1#
```

Step 4. Step 4: Configuration of SDPs and Services

Services can be configured to use MPLS-TP LSPs once the LSP configuration is completed. SDPs and services are configured in a similar manner to those using static-labelled pseudowires without MPLS-TP.

Distributed services are configured to use MPLS-TP with the following steps:

- Configure an SDP with signaling off. With signaling off, the SDP far-end may then be configured as an MPLS-TP node-id or an IPv4 address. SDP keep-alive should be disabled.
- Configure the service, including the spoke-sdp using the SDP. To use MPLS-TP, the spoke-sdp must have statically assigned ingress and egress labels, the control-word must be enabled, and it must have an MPLS-TP identifier for the PW (the PW Path ID) configured. This is comprised of two parts, a Source Attachment Individual Identifier (SAII) and a Target Attachment Individual Identifier (TAII), both of which must be configured. Control channel status signaling may also be configured to support PW status signaling on the static MPLS-TP PW.

In this example, an SDP is configured to use the MPLS-TP LSP from PE-1 to PE-2, which will act as a transport for the static MPLS-TP PWs corresponding to Epipe 10 and Epipe 20. Another SDP is configured for the targeted LDP (T-LDP) PW segment corresponding to Epipe 20 between PE-2 and PE-4.

The following CLI shows the SDP between PE-1 and PE-2 and the SDP between PE-2 and PE-4:

```
*A:PE-1# configure service
      sdp 1 mpls create
        signaling off
        far-end node-id 10.0.0.2 global-id 65535
        lsp "LSP-PE-1-PE-2"
        no shutdown
      exit

*A:PE-2# configure service
      sdp 1 mpls create
        signaling off
        far-end node-id 10.0.0.1 global-id 64511
        lsp "LSP-PE-1-PE-2"
        no shutdown
      exit
      sdp 2 mpls create
        far-end 192.0.2.4
        lsp "LSP-PE-2-PE-4"
        no shutdown
      exit

*A:PE-4# configure service
      sdp 2 mpls create
        far-end 192.0.2.2
```

Configuration

```
lsp "LSP-PE-4-PE-2"  
no shutdown  
exit
```

Next, configure the services that will use the MPLS-TP LSPs.

The service configuration CLI tree for an Epipe service using MPLS-TP is as follows:

```
configure  
  service  
    epipe  
      [no] spoke-sdp sdp-id[:vc-id]  
        [no] hash-label  
        [no] standby-signaling-slave  
      [no] spoke-sdp sdp-id[:vc-id] [vc-type {ether|vlan}]  
        [create] [vc-switching] [no-endpoint | {endpoint [icb]}]  
        egress  
          vc-label <out-label>  
        ingress  
          vc-label <in-label>  
        [no] control-word  
        [no] pw-path-id  
        agi <agi>  
        sai-type2 <global-id:node-id:ac-id>  
        taii-type2 <global-id:node-id:ac-id>  
        exit  
      control-channel-status  
        [no] acknowledgment  
        [no] refresh-timer <value>  
        [no] request-timer <value> retry-timer <value> [timeout-multiplier <value>]  
        [no] shutdown  
        exit
```

Refer to the CLI command descriptions in the user guides for further details of these commands.

The following CLI examples show the Epipe service configuration at PE-1, PE-2, and the VPLS spoke-sdp termination point at PE-4.

Note that Epipe 10 belongs to customer 1, and Epipe 20 belongs to customer 2 in this example.

```
*A:PE-1# configure service  
  epipe 10 customer 1 create  
    sap 1/1/3 create  
    exit  
    spoke-sdp 1:10 create  
      ingress  
        vc-label 150  
      exit  
      egress  
        vc-label 151  
      exit  
      control-word  
      pw-path-id  
        sai-type2 64511:10.0.0.1:1  
        taii-type2 65535:10.0.0.2:1
```

```

        exit
        control-channel-status
            no shutdown
        exit
        no shutdown
    exit
    no shutdown
exit
epipe 20 customer 2 create
    sap 1/1/4 create
    exit
    spoke-sdp 1:20 create
        ingress
            vc-label 200
        exit
        egress
            vc-label 201
        exit
        control-word
        pw-path-id
            sai-type2 64511:10.0.0.1:2
            taii-type2 65535:10.0.0.2:2
        exit
        control-channel-status
            no shutdown
        exit
        no shutdown
    exit
    no shutdown
exit

```

At PE-2, Epipe 10 terminates on a SAP on port 1/1/3, while Epipe 20 is switched between a static MPLS-TP PW segment (spoke-sdp 1:20) and a T-LDP signaled PW segment (spoke-sdp 2:1) for backhaul to the remote PE-4 containing the VPLS service.

```

*A:PE-2# configure service
    epipe 10 customer 1 create
        sap 1/1/3 create
        exit
        spoke-sdp 1:10 create
            ingress
                vc-label 151
            exit
            egress
                vc-label 150
            exit
            control-word
            pw-path-id
                sai-type2 65535:10.0.0.2:1
                taii-type2 64511:10.0.0.1:1
            exit
            control-channel-status
                no shutdown
            exit
            no shutdown
        exit
        no shutdown
    exit
    no shutdown

```

Configuration

```
exit
epipe 20 customer 2 vc-switching create
    spoke-sdp 1:20 create
        ingress
            vc-label 201
        exit
        egress
            vc-label 200
        exit
        control-word
        pw-path-id
            sai-type2 65535:10.0.0.2:2
            taii-type2 64511:10.0.0.1:2
        exit
        control-channel-status
            no shutdown
        exit
        no shutdown
    exit
    spoke-sdp 2:1 create
        control-word
        no shutdown
    exit
    no shutdown
exit
```

At PE-4, the T-LDP signaled PW segment for Epipe 20 is terminated on a VPLS service:

```
*A:PE-4# configure service
    vpls 1 customer 2 create
        stp
            shutdown
        exit
        sap 1/1/1 create
        exit
        spoke-sdp 2:1 create
            control-word
            no shutdown
        exit
        no shutdown
    exit
```

Epipe 10 uses a static MPLS-TP PW from end to end, which can be tested using the Virtual Circuit Connectivity Verification vccv-ping command at PE-1, as follows:

```
*A:PE-1# oam vccv-ping static 1:10
VCCV-PING 1:10 84 bytes MPLS payload
Seq=1, send from intf int-PE-1-PE-2
    send from lsp LSP-PE-1-PE-2
    reply via Control Channel
    src id tlv received: GlobalId 65535 NodeId 10.0.0.2
    cv-data-len=44 rtt=0.597ms rc=3 (EgressRtr)

---- VCCV PING 1:10 Statistics ----
1 packets sent, 1 packets received, 0.00% packet loss
round-trip min = 0.597ms, avg = 0.597ms, max = 0.597ms, stddev = 0.000ms
```


*A:PE-1#

The operation of control channel status signaling can also be tested for this Epipe, as follows:

Shutdown the port the SAP on PE-2 is using:

*A:PE-2# configure port 1/1/3 shutdown

The PW peer status bits for the spoke-sdp for Epipe 10, signaled using control channel status signaling, can be displayed at node PE-1 using the following command (note that some of the show command output has been removed for brevity). The peer PW status bits are shown in bold in the output below.

```
*A:PE-1# show service id 10 sdp detail
=====
Services: Service Destination Points Details
=====
-----
Sdp Id 1:10 - (10.0.0.2:65535)
-----
Description      : (Not Specified)
SDP Id           : 1:10                               Type           : Spoke
Spoke Descr      : (Not Specified)
VC Type          : Ether                               VC Tag          : n/a
Admin Path MTU   : 0                                   Oper Path MTU   : 1556
Delivery         : MPLS
Far End          : 10.0.0.2:65535
Tunnel Far End   : n/a                               LSP Types       : MPLSTP
Hash Label       : Disabled                           Hash Lbl Sig Cap : Disabled
Oper Hash Label  : Disabled

Admin State      : Up                                Oper State      : Up
Acct. Pol        : None                              Collect Stats   : Disabled
Ingress Label    : 150                               Egress Label    : 151
Ingr Mac Fltr-Id : n/a                               Egr Mac Fltr-Id : n/a
Ingr IP Fltr-Id  : n/a                               Egr IP Fltr-Id  : n/a
Ingr IPv6 Fltr-Id : n/a                             Egr IPv6 Fltr-Id : n/a
Admin ControlWord : Preferred                         Oper ControlWord : True
Admin BW(Kbps)   : 0                                  Oper BW(Kbps)   : 0
BFD Template     : None
BFD-Enabled      : no                                BFD-Encap       : ipv4
Last Status Change : 05/20/2015 11:57:25             Signaling       : None
Last Mgmt Change  : 05/20/2015 11:17:29
Endpoint         : N/A                               Precedence      : 4
PW Status Sig     : Enabled
Force Vlan-Vc     : Disabled                         Force Qinq-Vc   : Disabled
Class Fwding State : Down
Flags             : None
Local Pw Bits     : None
Peer Pw Bits      : lacIngressFault lacEgressFault
Peer Fault Ip     : None
Peer Vccv CV Bits : None
Peer Vccv CC Bits : None
---snipped---
```

Epipe 20 uses a static MPLS-TP PW from PE-1 to PE-2, identified by a static PW Forwarding Equivalence Class (FEC), and a T-LDP segment with FEC128 from PE-2 to PE-4. Therefore the target FEC used for a vccv-ping command from PE-1 to PE-4 is different from the local FEC for the PW at PE-1. VCCV-trace provides a useful tool to test the resulting multi-segment PW (MS-PW). Note that the same associated channel type must be used for both segments. This is the IPv4 channel.

```
*A:PE-1# oam vccv-trace static 1:20 assoc-channel ipv4 detail
VCCV-TRACE 1:20 with 116 bytes of MPLS payload
1 192.0.2.2 GlobalId 65535 NodeId 10.0.0.2
   rtt=0.599ms rc=8(DSRtrMatchLabel)
   Next segment: VcId=1 VcType=Ether Source=192.0.2.2 Remote=192.0.2.4
2 192.0.2.4 rtt=1.06ms rc=3(EgressRtr)

*A:PE-1#
```

The system supports the interworking of control channel status on a static MPLS-TP PW segment with T-LDP-signaled PW status on a T-LDP PW segment. This can be tested as follows.

Shutdown the port the spoke SDP on PE-4 is using:

```
*A:PE-4# configure port 1/1/2 shutdown
```

The PW peer status bits for the spoke-sdp for Epipe 20 can then be displayed at node PE-1 using the following command (note that some of the show command output has been removed for brevity). The peer PW status bits are shown in bold in the output below.

```
*A:PE-1# show service id 20 sdp detail

=====
Services: Service Destination Points Details
=====
-----
Sdp Id 1:20 -(10.0.0.2:65535)
-----
```

Description	: (Not Specified)		
SDP Id	: 1:20	Type	: Spoke
Spoke Descr	: (Not Specified)		
VC Type	: Ether	VC Tag	: n/a
Admin Path MTU	: 0	Oper Path MTU	: 1556
Delivery	: MPLS		
Far End	: 10.0.0.2:65535		
Tunnel Far End	: n/a	LSP Types	: MPLSTP
Hash Label	: Disabled	Hash Lbl Sig Cap	: Disabled
Oper Hash Label	: Disabled		
Admin State	: Up	Oper State	: Up
Acct. Pol	: None	Collect Stats	: Disabled
Ingress Label	: 200	Egress Label	: 201
Ingr Mac Fltr-Id	: n/a	Egr Mac Fltr-Id	: n/a
Ingr IP Fltr-Id	: n/a	Egr IP Fltr-Id	: n/a
Ingr IPv6 Fltr-Id	: n/a	Egr IPv6 Fltr-Id	: n/a
Admin ControlWord	: Preferred	Oper ControlWord	: True
Admin BW(Kbps)	: 0	Oper BW(Kbps)	: 0

MPLS Transport Profile

```
BFD Template      : None
BFD-Enabled       : no
Last Status Change : 05/20/2015 11:57:25
Last Mgmt Change  : 05/20/2015 11:17:29
Endpoint          : N/A
PW Status Sig     : Enabled
Force Vlan-Vc     : Disabled
Class Fwding State : Down
Flags             : None
Local Pw Bits     : None
Peer Pw Bits      : psnIngressFault psnEgressFault
Peer Fault Ip     : None
Peer Vccv CV Bits : None
Peer Vccv CC Bits : None
---snipped---
```

Conclusion

Release 11.0.R4 of SR OS introduced extensive MPLS Transport Profile (MPLS-TP) capabilities. MPLS-TP is intended to allow MPLS to be operated in a similar manner to existing transport technologies, with in-band proactive and on-demand operations and maintenance (OAM), and protection mechanisms that do not rely on a control plane to operate. The 7x50 can operate both as an LER and LSR for MPLS-TP LSPs, and as a T-PE and S-PE for PWs with MPLS-TP OAM. The 7x50 can therefore act as a node within an MPLS-TP network, or as a gateway between MPLS-TP and IP/MPLS domains.

This example has illustrated a simple configuration, demonstrating the role of the 7x50 as an LER and LSR for MPLS-TP LSPs, and how its already extensive multi-service capabilities can be extended over an MPLS-TP network and between MPLS-TP and IP/MPLS networks.

Point-to-Point LSPs

In This Chapter

This section provides information about point-to-point LSPs (static, LDP and RSVP-TE).

Topics in this section include:

- [Applicability on page 618](#)
- [Overview on page 619](#)
- [Configuration on page 623](#)
- [Conclusion on page 650](#)

Applicability

This chapter is applicable to all of the 7x50 platform. It was tested on release 13.0.R1. There are no pre-requisites or conditions on the hardware for this configuration.

Overview

Due to the connectionless nature of the network layer protocol IP, packets travel through the network on a hop-by-hop basis with routing decisions made at each node. As a result, hyper aggregation of data on certain links may occur and it may impact the provider's ability to provide guaranteed service levels across the network end-to-end. To address these shortcomings, MPLS (Multiprotocol Label Switching) was developed. The technology provides the capability to establish connection oriented paths, called Label Switched Paths (LSPs), over a connectionless (IP) network. The LSP offers a mechanism to engineer network traffic independently from the underlying network routing protocol (mostly IP) to improve the network resiliency and recovery options and to permit delivery of new services that are not readily supported by conventional IP routing techniques (Layer 2 IP VPNs). These benefits are essential for today's communication network explaining the wide deployment base of the MPLS technology.

RFC 3031, *Multiprotocol Label Switching Architecture*, specifies the MPLS architecture while this document describes the configuration and troubleshooting of point-to-point LSPs on Alcatel-Lucent SR and ESS series routers.

Packet Forwarding

As a packet of a connectionless network layer protocol travels from one router to the next, each router in the network makes an independent forwarding decision by performing the following basic tasks: first analyzing the packet's header, then referencing the local routing table to find the longest match based on the destination address in the IP header, and finally sending out the packet on the selected interface. In other terms, the first function partitions the entire set of possible packets into a set of Forwarding Equivalence Classes (FECs). All packets associated to a particular FEC will be forwarded along the same logical path to the same destination. The second function maps each FEC to a next hop destination router. Each router along the packet's path performs these actions.

On the other hand, in MPLS the assignment of a particular packet to a particular FEC is done just once, as the packet enters the network. In turn the FEC is mapped to an LSP, which is pre-signaled prior to any data flowing. An MPLS label, representing the FEC to which the packet is assigned, is attached to the packet (push operation) and once labeled the packet is forwarded to the next hop router along that LSP path. At subsequent hops, there is no further analysis of the packet's network layer header. Instead, the label is used as an index into a table which specifies the next hop and a new label. The old label is replaced with the new label (swap operation), and the packet is forwarded to its next hop. At the MPLS network egress, the label is removed from the packet (pop operation). If this router is the final destination (based on the remaining packet), the packet is handed to the receiving application (such as a VPLS domain). If this router is not the final destination of the packet, the packet will be sent into a new MPLS tunnel or forwarded by conventional IP forwarding towards the Layer 3 destination.

Terminology

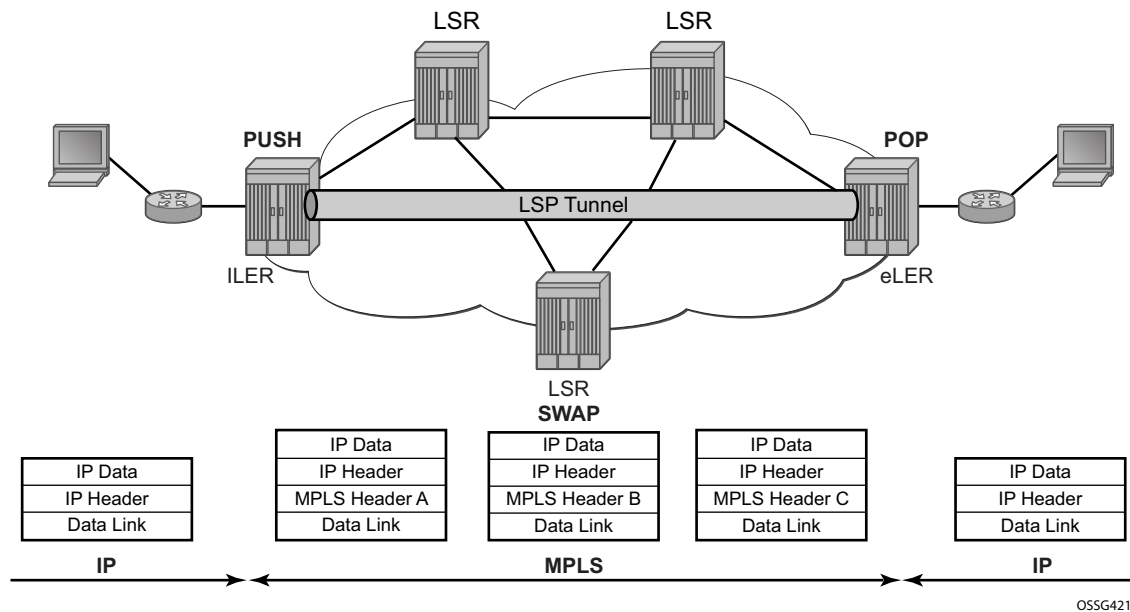


Figure 106: Generic MPLS Network, MPLS Label Operations

Figure 106 depicts a general network topology clarifying the MPLS-related terms. A Label Edge Router (LER) is a device at the edge of an MPLS network, with at least one interface outside the MPLS domain. A router is usually defined as an LER based on its position relative to a particular LSP. The MPLS router at the head-end of an LSP is called the ingress Label Edge Router (iLER). The MPLS router at the tail-end of an LSP is called the egress Label Edge Router (eLER). The iLER receives unlabeled packets from outside the MPLS domain then applies MPLS labels to the packets and forwards the labeled packets into the MPLS domain. The eLER receives labeled packets from the MPLS domain then removes the labels and forwards unlabeled packets outside the MPLS domain. The eLER can signal an implicit-null label (numeric value 3). This informs the previous hop to send MPLS packets without an outer label and so is known as Penultimate Hop Popping (PHP). This is also available when using static LSPs.

A Label Switching Router (LSR) is a device internal to an MPLS network, with all interfaces inside the MPLS domain. These devices switch labeled packets inside the MPLS domain. In the core of the network, LSR ignore the packet's network layer (IP) header and simply forward the packet using the MPLS label swapping mechanism.

A single LSP is uni-directional. In common practice, because the bi-directional nature of most traffic flows is implied, the term LSP often is used to define the pair of LSPs that enable the bidirectional flow. For ease of terminology and discussion however, the LSP in this chapter is referred to as a single entity.

LSP Establishment

Prior to packet forwarding, the LSP must be established. In order to do so, labels need to be distributed for the path. Labels are usually distributed by a downstream router in the upstream direction (relative to the data flow). There are a number of ways used for label distribution.

- The label distribution can be done manually by the network administrator by configuring static LSPs. Although a high control level of the labels in use is achieved, the LSP cannot enjoy the resilience and recovery functionality the dynamic label signaling protocols can offer.
- LDP (RFC 5036, *LDP Specification*) can be considered as an extension to the network IGP. As routers become aware of new destination networks, they advertise labels in the upstream direction that will allow upstream routers to reach the destination.
- RSVP-TE (RFC 3209, *RSVP-TE: Extensions to RSVP for LSP Tunnels*) can also be used to signal LSPs across the network. RSVP-TE is used for traffic engineering when the ingress router wishes to create an LSP with specific constraints beyond the best route chosen by the IGP. RSVP-TE identifies the specific path desired for the LSP and may include resource requirements for the path.

The most important benefit of the label swapping mechanism RSVP-TE is its ability to map any type of user traffic to a LSP that has been specifically engineered to satisfy user traffic requirements. Customized LSPs may be created based on hop count or bandwidth requirements. They can even be routed through specific network links or nodes, as specified by the ingress node. This offers service providers precise control over the flow of traffic in their networks and results in a network that operates more efficiently and provides more predictable and scalable services.

Testbed Topology

The network topology is displayed in [Figure 107](#). The setup consists of six 7750 nodes located in a single autonomous system.

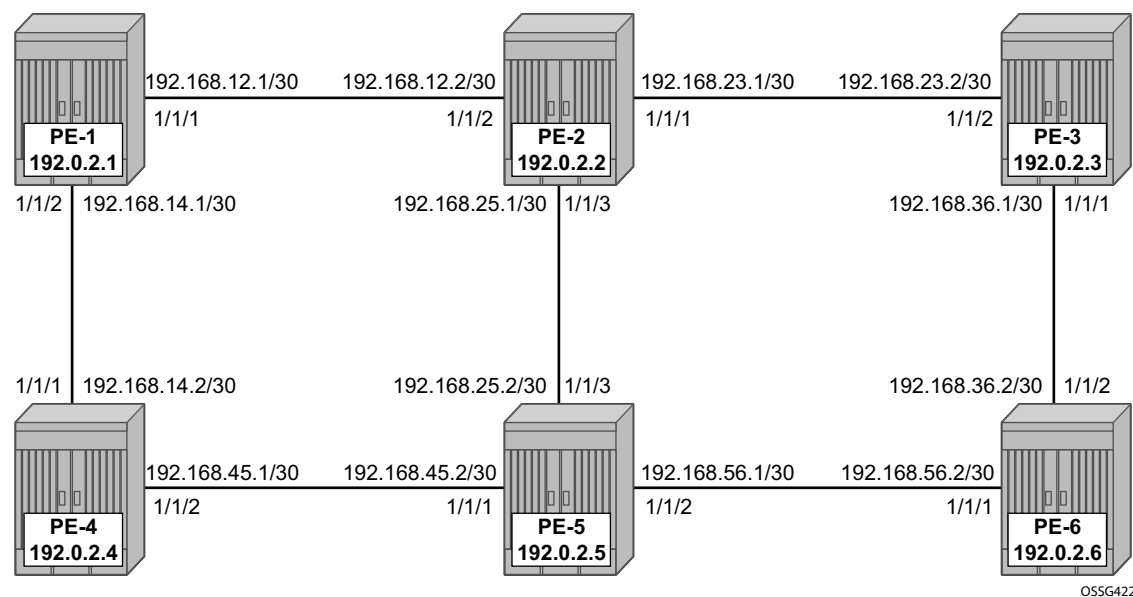


Figure 107: MPLS Testbed Topology

Configuration

As a general pre-requisite for the configuration of MPLS LSPs, a correctly working IGP is required¹; either OSPF or IS-IS can be used for this purpose.

For LSPs that are set up manually or using RSVP-TE, the first step is to enable MPLS on:

- All network interfaces that will be used to carry LSPs
- System IP address

For manually configured LSPs, any interface used by the static LSP must be added into the MPLS protocol instance, even though RSVP is not actually used to signal labels. For router PE-1 this results in the following configuration:

```
A:PE-1# configure router mpls
*A:PE-1>config>router>mpls$ interface "system"
*A:PE-1>config>router>mpls>if$ exit
*A:PE-1>config>router>mpls$ interface "int-PE-1-PE-2"
*A:PE-1>config>router>mpls>if$ exit
*A:PE-1>config>router>mpls$ interface "int-PE-1-PE-4"
*A:PE-1>config>router>mpls>if$ exit
*A:PE-1>config>router>mpls$ no shutdown
*A:PE-1>config>router>mpls$
```

1. Static LSPs do not need an IGP.

Manually Configured LSPs

As an example, a static LSP will be created starting from PE-1, running over PE-2 and PE-5, then terminating on PE-6 as depicted in [Figure 108](#).

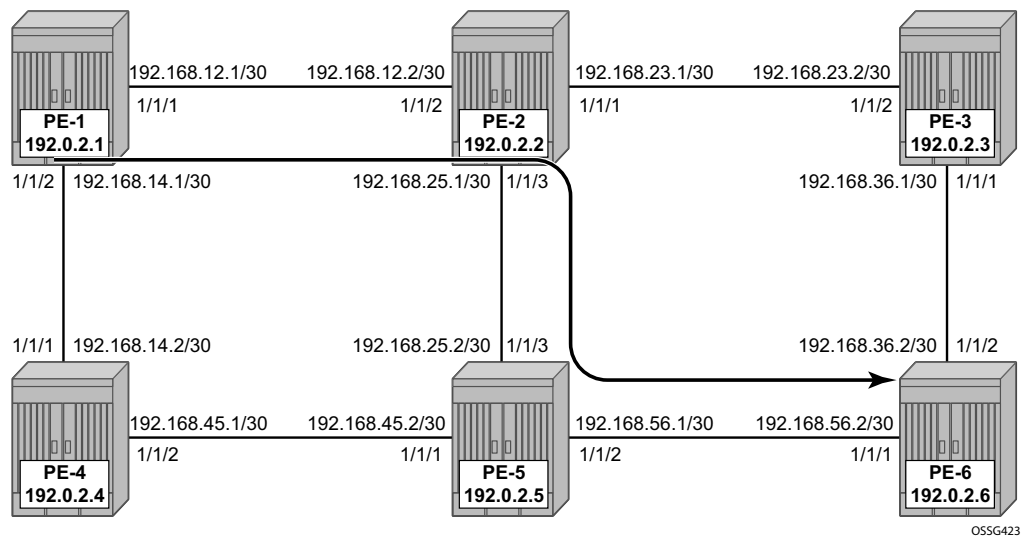


Figure 108: Static LSP Running Over PE-1 PE-2 PE-5 PE-6

For each node verify the available labels prior to beginning the configuration and verify the acceptable label range for use with static configurations.

```
*A:PE-1# show router mpls-labels label-range
=====
Label Ranges
=====
Label Type      Start Label End Label   Aging      Available  Total
-----
Static          32          18431      -          18400     18400
Dynamic         18432       131071     0          112640    112640
  Seg-Route      0           0          -           0         112640
=====
*A:PE-1#
```

The label range for static LSPs extends from the value 32 to 18431. To ensure the labels have not yet been allocated to another configuration, use the command:

```
*A:PE-1# show router mpls-labels label 32 18431 in-use
=====
MPLS Labels from 32 to 18431 (In-use)
=====
Label                Label Type          Label Owner
-----
In-use labels (Owner: All) in specified range    : 0
In-use labels in entire range                    : 0
=====
*A:PE-1#
```

As no LSPs are currently originating, passing through, or terminating in the node, no labels are in use and any label from the range 32 to 18431 is available for the static LSP. For the originating router PE-1, the label 100 will be used for the push operation on the interface towards PE-2.

Static LSPs are configured within the MPLS configuration context, but do not rely on dynamic label signaling.

The configuration of the MPLS static LSP head-end PE-1 contains:

- The system IP address of the destination router PE-6 (to).
- A push operation of the label 100.
- The interface address facing the current node of the next hop along the static path, which is PE-2 (nexthop).

```
*A:PE-1# configure router mpls static-lsp LSP-PE-1-PE-6
*A:PE-1>config>router>mpls>static-lsp$ to 192.0.2.6
*A:PE-1>config>router>mpls>static-lsp$ push 100 nexthop 192.168.12.2
*A:PE-1>config>router>mpls>static-lsp$ no shutdown
*A:PE-1>config>router>mpls>static-lsp$ exit all
```

The transit LSRs (PE-2 and PE-5) perform swap operations and forward the packet to the manually defined next-hop. On the LSR under the context of the interface on which the incoming LSP arrives, the correct label is selected (label-map) and in this context a swap operation with a new label and the new next hop (nexthop) is entered.

```
*A:PE-2# configure router mpls interface "int-PE-2-PE-1"
*A:PE-2>config>router>mpls>if# label-map 100
*A:PE-2>config>router>mpls>if>label-map$ swap 150 nexthop 192.168.25.2
*A:PE-2>config>router>mpls>if>label-map$ no shutdown
*A:PE-2>config>router>mpls>if>label-map$ exit all

*A:PE-5# configure router mpls interface "int-PE-5-PE-2"
*A:PE-5>config>router>mpls>if# label-map 150
*A:PE-5>config>router>mpls>if>label-map$ swap 200 nexthop 192.168.56.2
```

Manually Configured LSPs

```
*A:PE-5>config>router>mpls>if>label-map$ no shutdown
*A:PE-5>config>router>mpls>if>label-map$ exit all
```

The terminating router PE-6 performs a pop operation and forwards the now unlabeled packets external to the MPLS domain.

```
*A:PE-6# configure router mpls interface "int-PE-6-PE-5"
*A:PE-6>config>router>mpls>if# label-map 200
*A:PE-6>config>router>mpls>if>label-map$ pop
*A:PE-6>config>router>mpls>if>label-map$ no shutdown
*A:PE-6>config>router>mpls>if>label-map$ exit all
```

To verify the operational status of the static LSP configuration the **show router mpls static-lsp** command is used on the iLER. A static LSP is considered to be operationally up when only its next-hop is reachable. Since there is no check whether the end-to-end LSP path is up (the LSP connectivity to the eLER is never verified), it can be the static LSP path is broken while the iLER displays an operational enabled LSP.

```
*A:PE-1# show router mpls static-lsp
=====
MPLS Static LSPs (Originating)
=====
LSP Name      To          Next Hop      Out Label Up/Down Time  Adm  Opr
  ID                                     Out Port
-----
LSP-PE-1-PE- 192.0.2.6    192.168.12.2  100       0d 00:00:57   Up   Up
-6
  1                                     1/1/1
-----
LSPs : 1
=====
*A:PE-1#
```

On the LSR the **transit** keyword is added to the command.

```
*A:PE-2# show router mpls static-lsp transit
=====
MPLS Static LSPs (Transit)
=====
In Label      In Port      Out Label      Out Port      Next Hop      Adm  Opr
-----
100           1/1/2        150           1/1/3         192.168.25.2   Up   Up
-----
LSPs : 1
=====
*A:PE-2#
*A:PE-5# show router mpls static-lsp transit
=====
MPLS Static LSPs (Transit)
=====
In Label      In Port      Out Label      Out Port      Next Hop      Adm  Opr
-----
150           1/1/3        200           1/1/2         192.168.56.2   Up   Up
-----
```

```
-----
LSPs : 1
=====
*A:PE-5#
```

On the terminating router (eLER), the keyword **terminate** is added.

```
*A:PE-6# show router mpls static-lsp terminate
=====
MPLS Static LSPs (Terminate)
=====
In Label      In Port      Out Label     Out Port      Next Hop      Adm   Opr
-----
200           1/1/1        n/a           n/a           n/a           Up    Up
-----
LSPs : 1
=====
*A:PE-6#
```

To track the label action associated with the static LSP configuration, the **show router mpls interface label-map** command can be used on all LSRs and eLERs (not iLER).

```
*A:PE-2# show router mpls interface label-map
=====
MPLS Interfaces (Label-Map)
=====
In Label   In I/F      Out Label  Out I/F      Next Hop      Type      Adm  Opr
-----
100        1/1/2       150        1/1/3        192.168.25.2  Static    Up   Up
-----
Interfaces : 1
=====
*A:PE-2#
*A:PE-6# show router mpls interface label-map
=====
MPLS Interfaces (Label-Map)
=====
In Label   In I/F      Out Label  Out I/F      Next Hop      Type      Adm  Opr
-----
200        1/1/1       n/a        n/a          n/a           Static    Up   Up
-----
Interfaces : 1
=====
*A:PE-6#
```

The **show router mpls status** command is used to verify each of the LSP types, the number of configured LSPs and whether they originate on, transit through or terminate on the router.

```
*A:PE-1# show router mpls status
=====
MPLS Status
=====
Admin Status      : Up      Oper Status      : Up
Oper Down Reason  : n/a
FRR Object        : Enabled  Resignal Timer   : Disabled
Hold Timer        : 1 seconds Next Resignal    : N/A
Srlg Frr          : Disabled  Srlg Frr Strict  : Disabled
Admin Group Frr   : Disabled
Dynamic Bypass    : Enabled  User Srlg Database : Disabled
BypassResignalTimer : Disabled BypassNextResignal : N/A
LeastFill Min Thd : 5 percent LeastFill Reopti Thd : 10 percent
Local TTL Prop    : Enabled  Transit TTL Prop  : Enabled
AB Sample Multiplier : 1      AB Adjust Multiplier : 288
Exp Backoff Retry  : Disabled CSPF On Loose Hop   : Disabled
Lsp Init RetryTimeout : 30 seconds MBB Pref Current Hops : Disabled
Logger Event Bundling : Disabled
RetryIgpOverload   : Disabled

P2mp Resignal Timer : Disabled  P2mp Next Resignal : N/A
Sec FastRetryTimer   : Disabled  Static LSP FR Timer : 30 seconds
P2P Max Bypass Association: 1000
P2PActPathFastRetry  : Disabled  P2MP S2L Fast Retry : Disabled
In Maintenance Mode  : No
MplsTp               : Disabled
```



```

LSP Counts          Originate          Transit          Terminate
-----
Static LSPs         1              0              0
Dynamic LSPs        0              0              0
Detour LSPs         0              0              0
P2MP S2Ls           0              0              0
MPLS-TP LSPs        0              0              0
Mesh-P2P LSPs       0              0              0
One Hop-P2P LSPs    0              0              0
=====
*A:PE-1#

```

PHP can be used with static LSPs. This is achieved by configuring the penultimate LER to swap the incoming label to implicit-null instead of a specific label value (the label-map must be shutdown to add the **swap** command).

```

*A:PE-5# configure router mpls interface "int-PE-5-PE-2"
*A:PE-5>config>router>mpls>if# label-map 150
*A:PE-5>config>router>mpls>if>label-map# shutdown
*A:PE-5>config>router>mpls>if>label-map# swap implicit-null-label nexthop 192.168.56.2
*A:PE-5>config>router>mpls>if>label-map# no shutdown
*A:PE-5>config>router>mpls>if>label-map# exit all

```

The previous configuration will cause PE-5 to pop the top label from the incoming labeled frame received from PE-2 and send it to PE-6 without adding another outer label. The result can be seen from the following command (note that label 3 is never actually pushed onto a frame).

```

*A:PE-5# show router mpls static-lsp transit
=====
MPLS Static LSPs (Transit)
=====
In Label   In Port   Out Label  Out Port   Next Hop           Adm   Opr
-----
150        1/1/3     3          1/1/2      192.168.56.2       Up    Up
-----
LSPs : 1
=====
*A:PE-5#

```

If the traffic arriving at PE-5 was IP with a single label then it would arrive at PE-6 as unlabeled IP traffic.

If the static LSP spans a single hop (PE-1 to PE-2) the ingress LER would push the implicit-null instead of pushing a label.

```

*A:PE-1# configure router mpls static-lsp "LSP-PE-1-PE-2"
*A:PE-1>config>router>mpls>static-lsp$ to 192.0.2.2
*A:PE-1>config>router>mpls>static-lsp$ push implicit-null-label nexthop 192.168.12.2
*A:PE-1>config>router>mpls>static-lsp$ no shutdown
*A:PE-1>config>router>mpls>static-lsp$ exit all

```

In this case, no MPLS action (swap or pop) is required for this LSP on PE-2.

LDP

LDP is a simple label distribution protocol with basic MPLS functionality (no traffic engineering). Fast Reroute is supported, but that feature is beyond the scope of this chapter. LDP relies on the underlying routing information provided by an IGP in order to forward labeled packets. Each LDP configured LSR will originate a label for its system address and a label for each FEC for which it has a next-hop that is external to the MPLS domain, without the explicit need to create the LSPs. When deviations from this default behavior are desired, import and export policies can be applied.

The configuration is as simple as enabling the LDP protocol instance and adding all network interfaces, for each node. As an example the configuration on node PE-1 is displayed below; similar configurations apply on the other nodes.

```
A:PE-1# configure router ldp
*A:PE-1>config>router>ldp$ interface-parameters
*A:PE-1>config>router>ldp>if-params$ interface "int-PE-1-PE-2"
*A:PE-1>config>router>ldp>if-params>if$ exit
*A:PE-1>config>router>ldp>if-params$ interface "int-PE-1-PE-4"
*A:PE-1>config>router>ldp>if-params$ exit all
A:PE-1#
```

The **show router ldp discovery** and **show router ldp session** commands can be used to verify the LDP hello adjacencies and sessions. The adjacency type (AdjType) needs to be **Link** while the state should be **Established**.

```
*A:PE-1# show router ldp discovery
=====
LDP IPv4 Hello Adjacencies
=====
Interface Name          Local Addr          State
AdjType                 Peer Addr
-----
int-PE-1-PE-2           192.0.2.1:0        Estab
link                    192.0.2.2:0
int-PE-1-PE-4           192.0.2.1:0        Estab
link                    192.0.2.4:0

-----
No. of IPv4 Hello Adjacencies: 2
=====
LDP IPv6 Hello Adjacencies
=====
Interface Name          Local Addr          State
AdjType                 Peer Addr
-----
No Matching Entries Found
=====
*A:PE-1#
*A:PE-1# show router ldp session
=====
LDP IPv4 Sessions
```

```

=====
Peer LDP Id          Adj Type  State          Msg Sent  Msg Recv  Up Time
-----
192.0.2.2:0          Link      Established    35         37         0d 00:01:19
192.0.2.4:0          Link      Established    29         31         0d 00:00:58
-----
No. of IPv4 Sessions: 2
=====
LDP IPv6 Sessions
=====
Peer LDP Id
Adj Type          State          Msg Sent      Msg Recv      Up Time
-----
No Matching Entries Found
=====
*A:PE-1#

```

The show **router ldp bindings** command displays the contents of the LIB (Label Information Base) and should contain all labels locally generated (IngLbl) and those received from any LDP neighbors (EgrLbl), whether they are in use or not.

```

*A:PE-1# show router ldp bindings
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
(IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
        S - Status Signaled Up, D - Status Signaled Down
        E - Epipe Service, V - VPLS Service, M - Mirror Service
        A - Apipe Service, F - Fpipe Service, I - IES Service, R - VPRN service
        P - Ipipe Service, WP - Label Withdraw Pending, C - Cpipe Service
        BU - Alternate For Fast Re-Route, TLV - (Type, Length: Value)
=====
LDP IPv4 Prefix Bindings
=====
Prefix          IngLbl          EgrLbl
Peer            EgrIntf/LspId
EgrNextHop
-----
192.0.2.1/32          131071U          --
192.0.2.2:0          --
--

192.0.2.1/32          131071U          --
192.0.2.4:0          --
--

192.0.2.2/32          --              131071
192.0.2.2:0          1/1/1
192.168.12.2

192.0.2.2/32          131070U          131069
192.0.2.4:0          --
--

192.0.2.3/32          131069N          131069
192.0.2.2:0          1/1/1

```

```
192.168.12.2

192.0.2.3/32          131069U          131068
192.0.2.4:0          --
--

192.0.2.4/32          131068U          131068
192.0.2.2:0          --
--

192.0.2.4/32          --          131071
192.0.2.4:0          1/1/2
192.168.14.2

192.0.2.5/32          131067N          131067
192.0.2.2:0          1/1/1
192.168.12.2

192.0.2.5/32          131067U          131067
192.0.2.4:0          --
--

192.0.2.6/32          131066N          131066
192.0.2.2:0          1/1/1
192.168.12.2

192.0.2.6/32          131066U          131066
192.0.2.4:0          --
--

-----
No. of IPv4 Prefix Bindings: 12
=====...

---snip---
```

The **show router ldp bindings active** command displays the content of the Label Forwarding Information Base (LFIB) and contains all active labels and the associated label actions used for label switching packets.

```
*A:PE-1# show router ldp bindings active
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
              (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use,  N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
        (S) - Static           (M) - Multi-homed Secondary Support
        (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix          Op          IngLbl    EgrLbl
EgrNextHop      EgrIf/LspId
-----
192.0.2.1/32    Pop          131071    --
--              --
```

192.0.2.2/32 192.168.12.2	Push 1/1/1	--	131071
192.0.2.2/32 192.168.12.2	Swap 1/1/1	131070	131071
192.0.2.3/32 192.168.12.2	Push 1/1/1	--	131069
192.0.2.3/32 192.168.12.2	Swap 1/1/1	131069	131069
192.0.2.4/32 192.168.14.2	Push 1/1/2	--	131071
192.0.2.4/32 192.168.14.2	Swap 1/1/2	131068	131071
192.0.2.5/32 192.168.12.2	Push 1/1/1	--	131067
192.0.2.5/32 192.168.12.2	Swap 1/1/1	131067	131067
192.0.2.6/32 192.168.12.2	Push 1/1/1	--	131066
192.0.2.6/32 192.168.12.2	Swap 1/1/1	131066	131066

No. of IPv4 Prefix Active Bindings: 11
=====

---snip---

In the tunnel table, there are LDP LSPs to all other nodes:

```
*A:PE-1# show router tunnel-table
=====
IPv4 Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId  Pref    Nexthop      Metric
-----
192.0.2.2/32     ldp        MPLS  65537      9        192.168.12.2  10
192.0.2.3/32     ldp        MPLS  65538      9        192.168.12.2  20
192.0.2.4/32     ldp        MPLS  65539      9        192.168.14.2  10
192.0.2.5/32     ldp        MPLS  65540      9        192.168.12.2  20
192.0.2.6/32     ldp        MPLS  65541      9        192.168.12.2  30
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
*A:PE-1#
```

LDP

In order to signal PHP with LDP, implicit-null must be configured on the eLER.

```
A:PE-6# configure router ldp implicit-null-label
```

The implicit-null is signaled immediately, all related labels are withdrawn and re-advertised with the label 3. The new label would show up on PE-5 as a swap from the ingress label to an egress label of 3, although label 3 is not pushed on to the frame.

Import and Export Policies

The default label handling behavior is to originate label bindings for the system address and to propagate all FECs received. If this is not the desired behavior, an import/export policy can be applied. An LDP import policy impacts inbound filtering; an LDP export policy impacts outbound filtering. An export policy may be configured to control the set of LDP label bindings advertised by the LER (sending to LDP peers). As such, export policies are used to include additional FECs rather than filtering FECs from those advertised. An import policy can be used to control for which FECs a router will generate labels (accepting from LDP peers). This functionality is not unique to LDP; it can be used for RSVP-TE, OSPF, and IS-IS as well as others.

The policy can be global or LDP peer FEC prefix filtering, both for import and export. LDP peer FEC prefix filtering uses a similar policy context as the LDP global policies and works in addition to these global policies.

```
*A:PE-1# tree flat detail | match import-pref
configure router ldp session-parameters peer import-prefixes <policy-name>
    [<policy-name>...(upto 5 max)]
configure router ldp session-parameters peer no import-prefixes
configure router ldp targeted-session import-prefixes <policy-name>
    [<policy-name>...(upto 5 max)]
configure router ldp targeted-session no import-prefixes
*A:PE-1#

*A:PE-1# tree flat detail | match export-pref
configure router ldp session-parameters peer export-prefixes <policy-name>
    [<policy-name>...(upto 5 max)]
configure router ldp session-parameters peer no export-prefixes
configure router ldp targeted-session export-prefixes <policy-name>
    [<policy-name>...(upto 5 max)]
configure router ldp targeted-session no export-prefixes
*A:PE-1#
```

By default no labels are generated for directly connected (local) interfaces. To change this behavior, an export policy is created and applied to the LDP instance. There is no configuration difference in defining an import and export policy.

A policy starts with the keyword `begin` contains a list of entries (of which each has a number), and ends with the keyword `commit`. An entry typically contains matching criteria (however, it is not required in such cases where everything matches) and a corresponding action. Entries without an action are considered incomplete and are rendered inactive. When executing the policy, the router executes the specified action on the first matching statement; it does not process any further matches. For this reason, entries must be sequenced correctly from most to least specific.

The configuration of the LDP export policy for local interfaces is given below.

```
A:PE-1# configure router policy-options
A:PE-1>config>router>policy-options# begin
A:PE-1>config>router>policy-options# policy-statement LDP-export
A:PE-1>config>router>policy-options>policy-statement$ entry 10
```

```

A:PE-1>config>router>policy-options>policy-statement>entry$ from protocol direct
A:PE-1>config>router>policy-options>policy-statement>entry# action accept
A:PE-1>config>router>policy-options>policy-statement>entry>action# back
A:PE-1>config>router>policy-options>policy-statement>entry# back
A:PE-1>config>router>policy-options>policy-statement# back
A:PE-1>config>router>policy-options# commit
A:PE-1>config>router>policy-options# exit all

```

There are 11 active LDP bindings before applying the export policy.

```

*A:PE-1# show router ldp bindings active
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
      (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use,  N - Label Not In Use, W - Label Withdrawn
      WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
      (S) - Static           (M) - Multi-homed Secondary Support
      (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
LDP IPv4 Prefix Bindings (Active)
=====
Prefix                                     Op           IngLbl      EgrLbl
EgrNextHop                               EgrIf/LspId
-----
192.0.2.1/32                             Pop           131071      --
  --                                     --
192.0.2.2/32                             Push          --         131071
192.168.12.2                             1/1/1
192.0.2.2/32                             Swap          131070      131071
192.168.12.2                             1/1/1
192.0.2.3/32                             Push          --         131069
192.168.12.2                             1/1/1
192.0.2.3/32                             Swap          131069      131069
192.168.12.2                             1/1/1
192.0.2.4/32                             Push          --         131071
192.168.14.2                             1/1/2
192.0.2.4/32                             Swap          131068      131071
192.168.14.2                             1/1/2
192.0.2.5/32                             Push          --         131067
192.168.12.2                             1/1/1
192.0.2.5/32                             Swap          131067      131067
192.168.12.2                             1/1/1
192.0.2.6/32                             Push          --         131066
192.168.12.2                             1/1/1
192.0.2.6/32                             Swap          131066      131066
192.168.12.2                             1/1/1

```



```
-----
No. of IPv4 Prefix Active Bindings: 11
=====
```

```
---snip---
```

The LDP export or import policy is applied to the LDP instance on the router, respectively, with the **export** or **import** keyword.

```
A:PE-1# configure router ldp export LDP-export
```

When the export policy is applied, the active LDP binding table has additional entries: the local interfaces of PE-x.

```
*A:PE-1# show router ldp bindings active
```

```
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
              (IPv6 LSR ID ::[0])
=====
```

```
Legend: U - Label In Use,  N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
        (S) - Static          (M) - Multi-homed Secondary Support
        (B) - BGP Next Hop (BU) - Alternate Next-hop for Fast Re-Route
=====
```

```
LDP IPv4 Prefix Bindings (Active)
=====
```

Prefix EgrNextHop	Op EgrIf/LspId	IngLbl	EgrLbl
192.0.2.1/32	Pop	131071	--
--	--		
192.0.2.2/32	Push	--	131071
192.168.12.2	1/1/1		
192.0.2.2/32	Swap	131070	131071
192.168.12.2	1/1/1		
192.0.2.3/32	Push	--	131069
192.168.12.2	1/1/1		
192.0.2.3/32	Swap	131069	131069
192.168.12.2	1/1/1		
192.0.2.4/32	Push	--	131071
192.168.14.2	1/1/2		
192.0.2.4/32	Swap	131068	131071
192.168.14.2	1/1/2		
192.0.2.5/32	Push	--	131067
192.168.12.2	1/1/1		
192.0.2.5/32	Swap	131067	131067

LDP

192.168.12.2	1/1/1		
192.0.2.6/32	Push	--	131066
192.168.12.2	1/1/1		
192.0.2.6/32	Swap	131066	131066
192.168.12.2	1/1/1		
192.168.12.0/30	Pop	131063	--
--	--		
192.168.14.0/30	Pop	131065	--
--	--		
192.168.23.0/30	Swap	131060	131062
192.168.12.2	1/1/1		
192.168.25.0/30	Swap	131062	131061
192.168.12.2	1/1/1		
192.168.36.0/30	Swap	131059	131059
192.168.12.2	1/1/1		
192.168.45.0/30	Swap	131064	131064
192.168.14.2	1/1/2		
192.168.56.0/30	Swap	131061	131060
192.168.12.2	1/1/1		

No. of IPv4 Prefix Active Bindings: 18

=====

---snip---

RSVP-TE

RSVP-TE, an extension of the original RSVP protocol, has two major benefits adding to the basic MPLS functionality. The first benefit is traffic engineering, which allows the ingress router to create an LSP with specific constraints beyond the best route chosen by the IGP. The second benefit is improved network resiliency when a link or node fails in the network.

In this section, an RSVP-TE based LSP is established from PE-1 to PE-6, starting with a simple LSP with no specific TE constraints. Although the Fast Reroute (FRR), admin groups, hop limit restriction, and bandwidth reservation features could be implemented, the usage of these features is beyond the scope of this document.

Like the configuration of static LSPs, the MPLS instance needs to be enabled on each router and all network interfaces facing the MPLS domain as well as the system interface. When adding interfaces to the MPLS instance they are automatically added to the RSVP instance as well, but the instance itself is still in an administrative shutdown state. The next step is to enable the RSVP instance on all routers in the MPLS network. As a result all interfaces facing the MPLS domain as well as the system interface are added to the MPLS and RSVP instance and both instances are in a no shutdown state. For PE-1 this comes to:

```
A:PE-1# configure router rsvp
*A:PE-1>config>router>rsvp# no shutdown
*A:PE-1>config>router>rsvp# info
-----
      interface "system"
          no shutdown
      exit
      interface "int-PE-1-PE-2"
          no shutdown
      exit
      interface "int-PE-1-PE-4"
          no shutdown
      exit
      no shutdown
-----
*A:PE-1>config>router>rsvp#
```

Strict or loose path

On the iLER first the definition of a path is required. A path is a sequence of MPLS routers (hops) through which the LSP -using that path- has to pass. It is not uniquely bound to a particular LSP; it can be used by any LSP originating in that node. A hop in a path can be strict or loose: strict or loose meaning that the LSP must take either a direct path from the previous hop router to this router (strict) or can traverse through other routers (loose). The hops missing in the loose path definition are created by calculating the IGP shortest path. A third possibility is an entirely empty path implying not a single node is required to be present in the LSP path and the shortest path from the IGP is used to define the LSP path. The use of other techniques, such as the use of admin groups or shared risk link groups, are not covered in this document. Three paths are configured below, respectively:

1. Only strict hops
2. Mixed strict and loose hops
3. Completely loose path

To find a valid path, the last hop in the path sequence needs to be the system IP or an interface address of the terminating router (eLER). The IP addresses in the hop command can be the node's system IP addresses or its interface addresses. However, it is recommended to use the system IP addresses with keyword loose as this allows more flexibility when finding new paths in failover scenarios (because the upstream node could use any of multiple paths to the system address, where specifying the interface address would restrict the upstream node to a single entry-point). The recommendation when using the keyword strict in the hop command context, is to use the physical link addresses.

```
*A:PE-1# configure router mpls

*A:PE-1>config>router>mpls# path "path-PE-1-PE-6-strict"
*A:PE-1>config>router>mpls>path$ hop 10 192.168.12.2 strict
*A:PE-1>config>router>mpls>path$ hop 20 192.168.25.2 strict
*A:PE-1>config>router>mpls>path$ hop 30 192.168.56.2 strict
*A:PE-1>config>router>mpls>path$ no shutdown
*A:PE-1>config>router>mpls>path$ exit

*A:PE-1>config>router>mpls# path "path-PE-1-PE-6-semiLoose"
*A:PE-1>config>router>mpls>path$ hop 10 192.0.2.5 loose
*A:PE-1>config>router>mpls>path$ hop 20 192.168.56.2 strict
*A:PE-1>config>router>mpls>path$ no shutdown
*A:PE-1>config>router>mpls>path$ exit

*A:PE-1>config>router>mpls# path "pathLoose"
*A:PE-1>config>router>mpls>path$ no shutdown
*A:PE-1>config>router>mpls>path$ exit all
```

The paths can be checked with the **show router mpls path** command.

```
*A:PE-1# show router mpls path
=====
MPLS Path:
=====
```

Path Name	Adm	Hop	Index	IP Address	Strict/Loose
path-PE-1-PE-6-strict	Up	10		192.168.12.2	Strict
		20		192.168.25.2	Strict
		30		192.168.56.2	Strict
path-PE-1-PE-6-semiLoose	Up	10		192.0.2.5	Loose
		20		192.168.56.2	Strict
pathLoose	Up	no hops		n/a	n/a

```
-----
Total Paths : 3
=====
*A:PE-1#
```

Simple RSVP LSP

The configuration of a simple LSP using RSVP signaling contains at least on the iLER:

- System IP address of the terminating node (to)
- Path the LSP will take to the eLER (primary)
- Administratively enabled (no shutdown)

```
*A:PE-1# configure router mpls
*A:PE-1>config>router>mpls# lsp LSP-PE-1-PE-6
*A:PE-1>config>router>mpls>lsp$ to 192.0.2.6
*A:PE-1>config>router>mpls>lsp$ primary "path-PE-1-PE-6-strict"
*A:PE-1>config>router>mpls>lsp>primary$ exit
*A:PE-1>config>router>mpls>lsp$ secondary "pathLoose"
*A:PE-1>config>router>mpls>lsp>secondary$ exit
*A:PE-1>config>router>mpls>lsp$ no shutdown
*A:PE-1>config>router>mpls>lsp$ exit all
```

The nodes through which the LSP will pass (LSRs and eLER) require no additional configuration: enabling MPLS (and automatically RSVP together with it) on their interfaces suffices.

An overview of all LSPs configured on a particular node is given by the **show router mpls lsp** command. More details about a particular LSP can be retrieved by adding the keyword **detail** to the previous command.

```

*A:PE-1# show router mpls lsp
=====
MPLS LSPs (Originating)
=====
LSP Name                               To           Tun      Fastfail  Adm  Opr
                                Id           Config
-----
LSP-PE-1-PE-6                        192.0.2.6    1        No        Up   Up
-----
LSPs : 1
=====
*A:PE-1#
*A:PE-1# show router mpls lsp "LSP-PE-1-PE-6" detail
=====
MPLS LSPs (Originating) (Detail)
=====
Type : Originating
-----
LSP Name      : LSP-PE-1-PE-6
LSP Type      : RegularLsp
From          : 192.0.2.1
Adm State     : Up
LSP Up Time   : 0d 00:00:19
Transitions   : 1
Retry Limit   : 0
Signaling     : RSVP
Hop Limit     : 255
Adaptive      : Enabled
FastReroute   : Disabled
CSPF          : Disabled
Metric        : N/A
Load Balanc*  : N/A
Include Grps  :
None
Least Fill    : Disabled

LSP Tunnel ID : 1
To            : 192.0.2.6
Oper State    : Up
LSP Down Time : 0d 00:00:00
Path Changes  : 1
Retry Timer   : 30 sec
Resv. Style   : SE
Negotiated MTU : 1564
ClassType     : 0
Oper FR       : Disabled
ADSPEC        : Disabled

Exclude Grps  :
None

Revert Timer: Disabled
Auto BW      : Disabled
LdpOverRsvp  : Enabled
IGP Shortcut: Enabled
IGP LFA      : Disabled
BGPTransTun  : Enabled
Oper Metric  : 16777215
Prop Adm Grp: Disabled

Next Revert In : N/A
VprnAutoBind   : Enabled
BGP Shortcut   : Enabled
IGP Rel Metric : Disabled

Primary(a)     : path-PE-1-PE-6-strict
Bandwidth      : 0 Mbps
Secondary      : pathLoose
Bandwidth      : 0 Mbps
Up Time        : 0d 00:00:19
Down Time      : 0d 00:00:19
=====
* indicates that the corresponding row element may have been truncated.
*A:PE-1#

```

In the tunnel table, there are two ways to go to PE-6: one LDP LSP and one RSVP LSP.

```
*A:PE-1# show router tunnel-table
=====
IPv4 Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId  Pref    Nexthop      Metric
-----
192.0.2.2/32      ldp        MPLS  65537      9        192.168.12.2   10
192.0.2.3/32      ldp        MPLS  65538      9        192.168.12.2   20
192.0.2.4/32      ldp        MPLS  65539      9        192.168.14.2   10
192.0.2.5/32      ldp        MPLS  65540      9        192.168.12.2   20
192.0.2.6/32      rsvp      MPLS  1      7      192.168.12.2 16777215
192.0.2.6/32      ldp      MPLS  65541    9      192.168.12.2 30
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
*A:PE-1#
```

In order to signal PHP with RSVP, implicit-null must be configured on the eLER (RSVP must be shutdown to perform this command).

```
*A:PE-6# configure router rsvp
*A:PE-6>config>router>rsvp# shutdown
*A:PE-6>config>router>rsvp# implicit-null-label
*A:PE-6>config>router>rsvp# no shutdown
*A:PE-6>config>router>rsvp# exit all
```

The implicit-null is signaled after re-enabling RSVP and would show up on PE-5 as an egress label of 3.

The use of implicit-null can also be enabled/disabled on a per interface basis (either RSVP, or the interface within RSVP, must be shutdown to perform this change).

```
A:PE-6>config>router>rsvp# interface "int-PE-6-PE-5"
A:PE-6>config>router>rsvp>if# implicit-null-label
  - implicit-null-label {<enable|disable>}
  - no implicit-null-label

<<enable|disable>> : keyword

A:PE-6>config>router>rsvp>if#
```

Manual Resignal

Instead of waiting for the resignal timer to expire, one can manually trigger the resignal process.

The command to resignal the path "path-PE-1-PE-6-strict" of LSP "LSP-PE-1-PE-6":

```
*A:PE-1# tools perform router mpls resignal lsp "LSP-PE-1-PE-6" path "path-PE-1-PE-6-strict"
```

The command to resignal all RSVP LSPs originating at node PE-1:

```
*A:PE-1# tools perform router mpls resignal delay 0
```

LSP OAM

The LSP diagnostics are modeled after ICMP echo request/reply which provides a mechanism to detect data plane failures in MPLS LSPs. For a given FEC, LSP ping verifies whether the packet reaches the egress label edge router (LER). While in LSP traceroute mode, the packet is sent to the control plane of each transit label switched router (LSR) which performs various checks to see if it is actually a transit LSR for the path.

```
*A:PE-1# oam lsp-ping "LSP-PE-1-PE-6"
LSP-PING LSP-PE-1-PE-6: 92 bytes MPLS payload
Seq=1, send from intf int-PE-1-PE-2, reply from 192.0.2.6
      udp-data-len=32 ttl=255 rtt=4.69ms rc=3 (EgressRtr)

---- LSP LSP-PE-1-PE-6 PING Statistics ----
1 packets sent, 1 packets received, 0.00% packet loss
round-trip min = 4.69ms, avg = 4.69ms, max = 4.69ms, stddev = 0.000ms
*A:PE-1#

*A:PE-1# oam lsp-trace "LSP-PE-1-PE-6"
lsp-trace to LSP-PE-1-PE-6: 0 hops min, 0 hops max, 116 byte packets
1  192.0.2.2  rtt=1.88ms rc=8(DSRtrMatchLabel) rsc=1
2  192.0.2.5  rtt=3.07ms rc=8(DSRtrMatchLabel) rsc=1
3  192.0.2.6  rtt=42.9ms rc=3(EgressRtr) rsc=1
*A:PE-1#
```

The same LSP OAM commands can be used for LDP based LSPs, for a specific destination FEC.

```
*A:PE-1# oam lsp-ping prefix 192.0.2.5/32
LSP-PING 192.0.2.5/32: 80 bytes MPLS payload
Seq=1, send from intf int-PE-1-PE-2, reply from 192.0.2.5
      udp-data-len=32 ttl=255 rtt=1.13ms rc=3 (EgressRtr)

---- LSP 192.0.2.5/32 PING Statistics ----
1 packets sent, 1 packets received, 0.00% packet loss
round-trip min = 1.13ms, avg = 1.13ms, max = 1.13ms, stddev = 0.000ms
*A:PE-1#
```



```
*A:PE-1# oam lsp-trace prefix 192.0.2.5/32
lsp-trace to 192.0.2.5/32: 0 hops min, 0 hops max, 104 byte packets
1 192.0.2.2 rtt=0.609ms rc=8(DSRtrMatchLabel) rsc=1
2 192.0.2.5 rtt=1.01ms rc=3(EgressRtr) rsc=1
*A:PE-1#
```

Debug Tools

A wide range of debug tools are available which can be tuned to the specific information of importance for a certain troubleshooting task. In the **debug router mpls** context, the LSP object to trace or monitor can be selected by the following parameters:

- LSP name
- Source address of the LSP (the **from** parameter in the LSP definition)
- Termination point of the LSP (the **to** parameter in the LSP definition)
- Tunnel ID of the LSP
- LSP ID

```
A:PE-1# debug router rsvp
- no rsvp
- rsvp [lsp <lsp-name>] [sender <sender-address>] [endpoint <endpoint-address>]
    [tunnel-id <tunnel-id>] [lsp-id <lsp-id>] [interface <ip-int-name>]

<lsp-name>          : [80 chars max]
<sender-address>    : a.b.c.d
<endpoint-address>  : a.b.c.d
<tunnel-id>         : [0..4294967295]
<lsp-id>            : [1..65535]
<ip-int-name>       : [32 chars max]

[no] event          + Enable/disable debugging for specific RSVP events
[no] packet         + Enable/disable debugging for specific RSVP packets
```

```
A:PE-1# debug router mpls
- mpls [lsp <lsp-name>] [sender <source-address>] [endpoint <endpoint-address>]
    [tunnel-id <tunnel-id>]
    [lsp-id <lsp-id>]
- no mpls

<lsp-name>          : [80 chars max]
<source-address>    : a.b.c.d
<endpoint-address>  : a.b.c.d
<tunnel-id>         : [0..4294967295]
<lsp-id>            : [1..65535]

[no] event          + Enable/disable debugging for specific MPLS events
```

In the **debug** command tree, the MPLS event type can be selected (tracing must be enabled):

```
A:PE-1# debug router mpls lsp "LSP-PE-1-PE-6" event
- event
- no event

[no] all            - Enable/disable debugging for MPLS all
```

```

[no] frf          - Enable/disable debugging for MPLS frf
[no] iom          - Enable/disable debugging for MPLS iom
[no] lsp-setup    - Enable/disable debugging for MPLS lsp setup
[no] mbb          - Enable/disable debugging for MPLS mbb
[no] misc         - Enable/disable debugging for MPLS misc
[no] xc           - Enable/disable debugging for MPLS xc

```

As an example, the **all** keyword is entered, logging all MPLS events related to the selected LSP:

```
A:PE-1# debug router mpls lsp "LSP-PE-1-PE-6" event all
```

The last step is to create a log container which will gather all MPLS debugging information according to the criteria set in the debug context. The **from debug-trace** parameter must be configured but there are several options where the different captured entries will be stored: console, a syslog server, SNMP, local file on the compact flash card, a temporary circular memory buffer, or the telnet/SSH session from which you are logged into the node.

The log container's ID is just a local number without any other significance.

```

A:PE-1# configure log log-id 2
A:PE-1>config>log>log-id# from debug-trace
A:PE-1>config>log>log-id# to
  - to console
  - to file <log-file-id>
  - to memory [<size>]
  - to session
  - to snmp [<size>]
  - to syslog <syslog-id>

<console>          : keyword - specifies console as destination
<syslog-id>        : [1..10]
<snmp>             : keyword - specifies SNMP as destination
<log-file-id>      : [1..99]
<memory>           : keyword - specifies memory as destination
<size>             : [50..3000]
<session>          : keyword - specifies telnet session as destination

```

For this example, the temporary buffer (with adjustable size) is chosen and the log container is enabled (no shutdown).

```

A:PE-1>config>log>log-id# to memory
A:PE-1>config>log>log-id# no shutdown

```

All MPLS events related to the selected LSP are stored in the location (memory) specified. The content of this log container can be viewed through the **show log log-id 2** command.

```
*A:PE-1# show log log-id 2
=====
Event Log 2
=====
Description : (Not Specified)
Memory Log contents [size=100 next event=33 (not wrapped)]

32 2015/02/10 12:59:21.67 UTC MINOR: DEBUG #2001 Base MPLS
"MPLS: RTM
Add tunnel table entry for TunnelId 1 Dest 192.0.2.6
Owner RSVP Pref 7 Metric 30 MTU 1560 LDPoRSVP Yes VprnAutoBind Yes PropTtl YesIgpSh-
cut Yes BgpShcut No BgpTransTunn No BW 0
NHLFE[1]: outIf (3) outLbl (131071) nhAddr 192.168.14.2 nhIndex 66"

31 2015/02/10 12:59:21.67 UTC MINOR: DEBUG #2001 Base MPLS
"MPLS: LSP
Set operational state for LSP LSP-PE-1-PE-6 to Up, previous state is Down"

30 2015/02/10 12:59:21.67 UTC MINOR: DEBUG #2001 Base MPLS
"MPLS: LTN
Add PUSH for LspKey P2P: Session(192.0.2.6, 1, 192.0.2.1) Sender(192.0.2.1, 27658)
PHOP(0.0.0.0)
Primary NHLFE - OutLabel 131071 OutIfIndex 3 Next-hop 192.168.14.2"

29 2015/02/10 12:59:21.67 UTC MINOR: DEBUG #2001 Base MPLS
"MPLS: LSP
Set LspPath LSP-PE-1-PE-6::pathLoose(LspId 27658) as active path for LSP LSP-PE-1-PE-6"

28 2015/02/10 12:59:21.67 UTC MINOR: DEBUG #2001 Base MPLS
"MPLS: LSP
LSP LSP-PE-1-PE-6 has no active path. Set path LSP-PE-1-PE-6::pathLoose(LspId 27658) as
active"

27 2015/02/10 12:59:21.67 UTC MINOR: DEBUG #2001 Base MPLS
"MPLS: LSP
Secondary path LSP-PE-1-PE-6::pathLoose(LspId 27658) is operationally up for LSP LSP-PE-1-
PE-6"

26 2015/02/10 12:59:21.67 UTC MINOR: DEBUG #2001 Base MPLS
"MPLS: LSP Path
Set operational MTU for LspPath LSP-PE-1-PE-6::pathLoose(LspId 27658) to 1564"

25 2015/02/10 12:59:21.67 UTC MINOR: DEBUG #2001 Base MPLS
"MPLS: LSP Path
Set operational metric for LspPath LSP-PE-1-PE-6::pathLoose(LspId 27658) to 30"

24 2015/02/10 12:59:21.67 UTC MINOR: DEBUG #2001 Base MPLS
"MPLS: LSP Path
Set operational state for LspPath LSP-PE-1-PE-6::pathLoose(LspId 27658) to Up, previous
state is Down"

23 2015/02/10 12:59:21.67 UTC MINOR: DEBUG #2001 Base MPLS
"MPLS: LSP Path
LspPath LSP-PE-1-PE-6::pathLoose(LspId 27658) setup successfully"
```

```

22 2015/02/10 12:59:21.67 UTC MINOR: DEBUG #2001 Base MPLS
"MPLS: XC
P2P: Session(192.0.2.6, 1, 192.0.2.1) Sender(192.0.2.1, 27658) PHOP(0.0.0.0)
Create OutSegment with Index 18390
Create XC with XCIndex 18390"

21 2015/02/10 12:59:21.67 UTC MINOR: DEBUG #2001 Base MPLS
"MPLS: Resv
P2P: Session(192.0.2.6, 1, 192.0.2.1) Sender(192.0.2.1, 27658) PHOP(0.0.0.0)
Received label mapping for label 131071"

20 2015/02/10 12:59:21.67 UTC MINOR: DEBUG #2001 Base MPLS
"MPLS: LSP Path
Set operational MTU for LspPath LSP-PE-1-PE-6::pathLoose(LspId 27658) to 1564"

---snip---

=====

```

Debugging for LDP can be enabled per LDP interface or per LDP peer:

```

A:PE-1# debug router ldp
- ldp
- no ldp

[no] interface      + Enable/disable and configure debugging for an LDP interface
[no] peer           + Enable/disable and configure debugging for an LDP peer

...

```

Conclusion

MPLS provides the capability to establish connection oriented paths over a connectionless network. The LSP offers a mechanism to engineer network traffic on constraint-based paths rather than the IGP shortest path. This can greatly improve network resiliency. In this section, the configuration of several LSP features is given together with the associated show output which can be used to verify and troubleshoot.

RSVP Signaled Point-to-Multipoint LSPs

In This Chapter

This section provides information about RSVP signaled point-to-multipoint LSPs.

Topics in this section include:

- [Applicability on page 652](#)
- [Overview on page 653](#)
- [Configuration on page 655](#)
- [Conclusion on page 700](#)

Applicability

This feature is applicable to all of the 7x50 chassis and tested on release 13.0.R2. On all nodes involved with the LSP, at least chassis mode C is required. This means that modular systems should be equipped with IOM-2 line cards or higher.

Overview

Point-to-MultiPoint (P2MP) MPLS label switched paths (LSP) allow the source of multicast traffic to forward packets to one or many multicast receivers over a network without requiring a multicast protocol, such as PIM, to be configured in the network. A P2MP LSP tree is established in the control plane which path consists of a head-end node, one or many branch and bud nodes, and the leaf nodes. A bud node combines the roles of branch node and leaf node (for different source-to-leaf LSPs). Packets injected by the head-end node are replicated in the data plane at the branching nodes before they are delivered to the leaf nodes.

Similar to point-to-point (P2P) LSPs, also P2MP LSPs are unidirectional, originating on a head-end node (the ingress LER) and terminating on one or more leaf node(s) (the egress LER(s)). Initially, RSVP is used as signaling protocol. A P2MP LSP is modeled as a set of root-to-leaf sub LSPs (Source-to-Leaf: S2L). Each S2L is modeled as a point-to-point LSP in the control plane. This means that each S2L has its own PATH/RESV messages. This is called the de-aggregated method.

The forwarding of multicast packets to the LSP tree is based on static multicast routes initially but will evolve to BGP based VPN routes in the future. Forwarding multicast packets is initially done over P2MP RSVP LSPs in the base router instance but will evolve to VPRNs.

RSVP signalled P2MP LSPs can have fast reroute (FRR) enabled, the facility method (one-to-many) with link protection is supported.

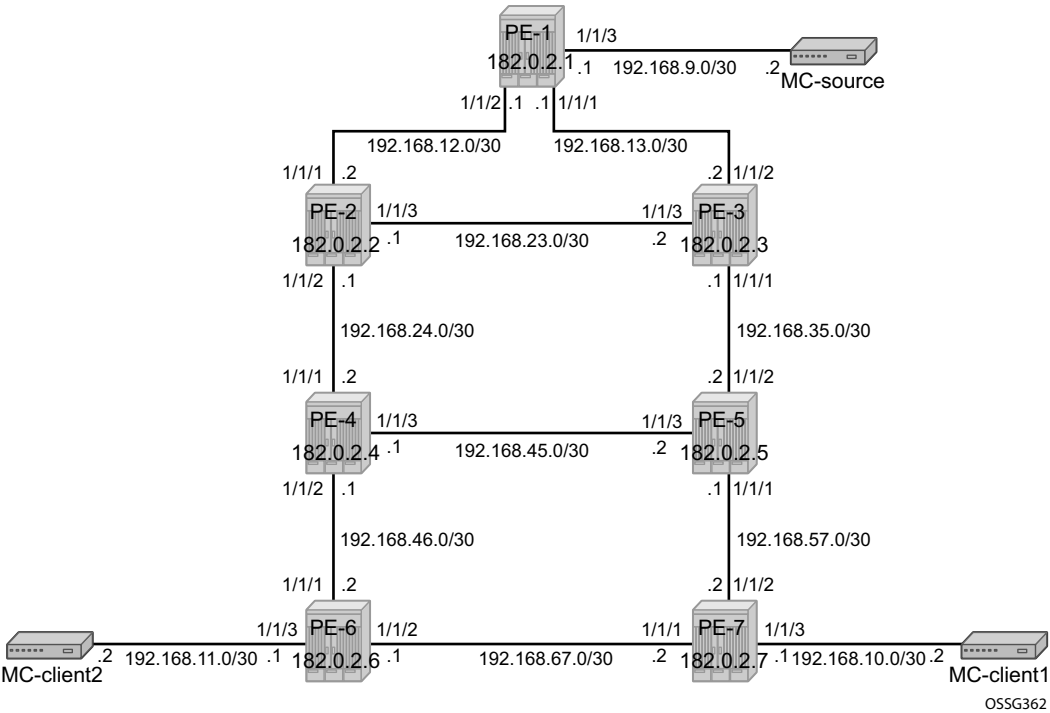


Figure 109: P2MP Network Topology

Configuration

The following sections describe the tasks you must perform to configure RSVP signaled point-to-multipoint LSPs.

Configuring the IP/MPLS Network

After configuring the cards and MDAs, the correct chassis-mode must be set (C or higher) on all MPLS nodes along the P2MP LSP.

```
A:PE-1# configure system chassis-mode
- chassis-mode <chassis-mode> [force]

<chassis-mode>      : a|b|c|d
                     : Chassis Mode corresponds to scaling and feature sets associated
                     : with a card.
                     : mode 'a' corresponds to the iom-20g.
                     : mode 'b' corresponds to the iom-20g-b.
                     : mode 'c' corresponds to the iom2-20g.
                     : mode 'd' corresponds to the iom3-xp.
<force>             : keyword - forces an upgrade from a lesser scaling and feature
                     : set to a greater one.

A:PE-1#

A:PE-1# configure system chassis-mode c

*A:PE-1# show chassis | match "chassis mode"
Admin chassis mode      : c
Oper chassis mode       : c
```

The system addresses and Layer 3 interface addresses are configured according to [Figure 109](#). An interior gateway protocol (IGP) is needed to distribute routing information to all PEs. In our case, the IGP is OSPF using the backbone area (area 0.0.0.0). A configuration example is shown for PE-1. A similar configuration is needed on all seven PEs.

```
*A:PE-1# configure router
interface "int-PE-1-PE-2"
  address 192.168.12.1/30
  port 1/1/2
exit
interface "int-PE-1-PE-3"
  address 192.168.13.1/30
  port 1/1/1
exit
interface "system"
  address 192.0.2.1/32
exit

*A:PE-1# configure router
ospf
```

Configuring the IP/MPLS Network

```
traffic-engineering
area 0.0.0.0
  interface "system"
  exit
  interface "int-PE-1-PE-2"
    interface-type point-to-point
  exit
  interface "int-PE-1-PE-3"
    interface-type point-to-point
  exit
exit
exit
```

Since fast reroute (FRR) will be enabled on the P2MP LSP, traffic engineering (TE) is needed on the IGP. By doing this, OSPF will generate opaque LSAs which are collected in a traffic engineering database (TED), separate from the traditional OSPF topology database. OSPF interfaces are setup as type '*point-to-point*' to improve convergence, no DR/BDR election process is done.¹

To verify that OSPF neighbors are up (state **Full**), **show router ospf neighbor** is performed. To check if Layer 3 interface addresses/subnets are known on all PEs, **show router route-table** or **show router fib iom-card-slot** will display the content of the forwarding information base (FIB).

```
*A:PE-1# show router ospf neighbor
```

```
=====
OSPF Neighbors
=====
Interface-Name          Rtr Id          State    Pri  RetxQ  TTL
-----
int-PE-1-PE-2           192.0.2.2       Full     1    0      31
int-PE-1-PE-3           192.0.2.3       Full     1    0      35
-----
No. of Neighbors: 2
=====
*A:PE-1#
```

```
*A:PE-1# show router route-table
```

```
=====
Route Table (Router: Base)
=====
Dest Prefix              Type    Proto    Age          Pref
Next Hop[Interface Name] Metric
-----
192.0.2.1/32             Local   Local    03h25m22s    0
    system               0
192.0.2.2/32             Remote  OSPF     02h08m41s    10
    192.168.12.2         10
192.0.2.3/32             Remote  OSPF     03h24m21s    10
-----
```

1. Convergence is out of the scope of this configuration note.

RSVP Signaled Point-to-Multipoint LSPs

192.168.13.2			10
192.0.2.4/32	Remote	OSPF	02h08m28s 10
192.168.12.2			20
192.0.2.5/32	Remote	OSPF	02h05m36s 10
192.168.12.2			30
192.0.2.6/32	Remote	OSPF	01h49m05s 10
192.168.12.2			30
192.0.2.7/32	Remote	OSPF	02h05m36s 10
192.168.12.2			40
192.168.12.0/30	Local	Local	03h25m22s 0
int-PE-1-PE-2			0
192.168.13.0/30	Local	Local	03h25m22s 0
int-PE-1-PE-3			0
192.168.23.0/30	Remote	OSPF	02h08m28s 10
192.168.12.2			20
192.168.24.0/30	Remote	OSPF	02h08m28s 10
192.168.12.2			20
192.168.35.0/30	Remote	OSPF	02h05m36s 10
192.168.12.2			40
192.168.45.0/30	Remote	OSPF	02h08m28s 10
192.168.12.2			30
192.168.46.0/30	Remote	OSPF	01h49m10s 10
192.168.12.2			30
192.168.57.0/30	Remote	OSPF	02h05m36s 10
192.168.12.2			40
192.168.67.0/30	Remote	OSPF	01h49m05s 10
192.168.12.2			40

No. of Routes: 16

=====

*A:PE-1#

*A:PE-1# show router fib 1

=====

FIB Display

=====

Prefix

Protocol

NextHop

192.0.2.1/32	LOCAL
192.0.2.1 (system)	
192.0.2.2/32	OSPF
192.168.12.2 (int-PE-1-PE-2)	
192.0.2.3/32	OSPF
192.168.13.2 (int-PE-1-PE-3)	
192.0.2.4/32	OSPF
192.168.12.2 (int-PE-1-PE-2)	
192.0.2.5/32	OSPF
192.168.12.2 (int-PE-1-PE-2)	
192.0.2.6/32	OSPF
192.168.12.2 (int-PE-1-PE-2)	
192.0.2.7/32	OSPF
192.168.12.2 (int-PE-1-PE-2)	
192.168.12.0/30	LOCAL
192.168.12.0 (int-PE-1-PE-2)	
192.168.13.0/30	LOCAL
192.168.13.0 (int-PE-1-PE-3)	
192.168.23.0/30	OSPF
192.168.12.2 (int-PE-1-PE-2)	

Configuring the IP/MPLS Network

```
192.168.24.0/30                                OSPF
    192.168.12.2 (int-PE-1-PE-2)
192.168.35.0/30                                OSPF
    192.168.12.2 (int-PE-1-PE-2)
192.168.45.0/30                                OSPF
    192.168.12.2 (int-PE-1-PE-2)
192.168.46.0/30                                OSPF
    192.168.12.2 (int-PE-1-PE-2)
192.168.57.0/30                                OSPF
    192.168.12.2 (int-PE-1-PE-2)
192.168.67.0/30                                OSPF
    192.168.12.2 (int-PE-1-PE-2)
-----
Total Entries : 16
=====
*A:PE-1#
```

In our example on PE-1, the interface towards the multicast source is configured in an IES service. This could have been on a router interface instead.

```
*A:PE-1# configure service
    ies 1 customer 1 create
        interface "int-PE-1-MC-source" create
            address 192.168.9.1/30
            sap 1/1/3 create
            exit
        exit
    no shutdown
exit
```

Similar IES services are configured on PE-7 and PE-6 for multicast client 1 and multicast client 2.

```
*A:PE-7# configure service
    ies 1 customer 1 create
        interface "int-PE-7-MC-client1" create
            address 192.168.10.1/30
            sap 1/1/3 create
            exit
        exit
    no shutdown
exit

*A:PE-6# configure service
    ies 1 customer 1 create
        interface "int-PE-6-MC-client2" create
            address 192.168.11.1/30
            sap 1/1/3 create
            exit
        exit
    no shutdown
exit
```

The next step in the process of setting up a P2MP LSP, is enabling our Layer 3 interfaces in the MPLS and RSVP context on all involved PE nodes (from PE-1 to PE-7). By default, the system

interface is put automatically within the MPLS/RSVP context. When an interface is put in the MPLS context, 7x50 copies it also in the RSVP context. Explicit enabling of MPLS and RSVP context is done by the **no shutdown** command. Below you can find the MPLS/RSVP configuration for PE-1.

```
*A:PE-1# configure router mpls no shutdown
*A:PE-1# configure router rsvp no shutdown
*A:PE-1#

*A:PE-1# configure router mpls
      interface "system"
      exit
      interface "int-PE-1-PE-2"
      exit
      interface "int-PE-1-PE-3"
      exit
```

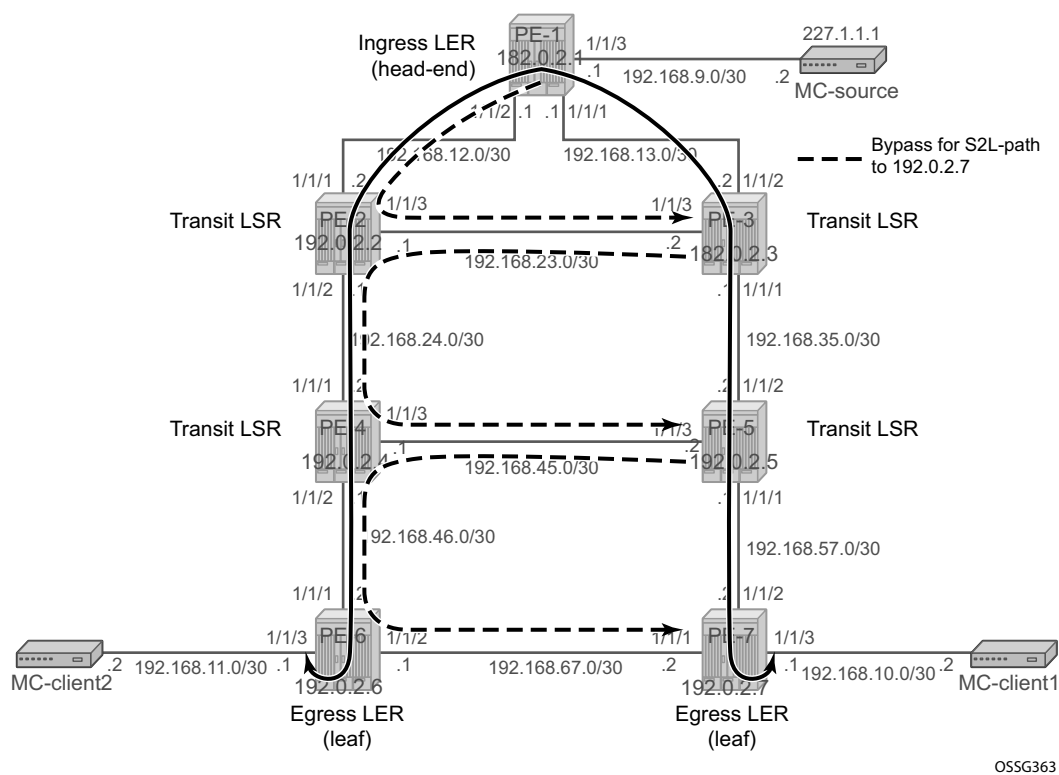


Figure 110: P2MP LSP LSP-p2mp-1

Configuring P2MP RSVP LSP

See [Figure 110](#) for P2MP LSP LSP-p2mp-1 with facility backup.

A P2MP LSP (LSP-p2mp-1) will be setup from PE-1 acting as head-end node and PE-6 and PE-7 acting as leaf nodes. Because FRR is enabled, Constrained Shortest Path First (CSPF) is enabled to do route calculations on the traffic engineering database (TED). FRR method **facility** is used without node protection, **facility** stands for one-to-many meaning that one bypass tunnel can protect a set of primary LSPs with similar backup constraints. When a link failure occurs on one of the active S2L paths, the Point of Local Repair (PLR) node will push an additional MPLS label on the incoming MPLS packet before sending it into the bypass tunnel downstream towards the merge point (MP) node.

In the first example, OSPF (our IGP) will do the path calculation to the two destinations (PE-6 and PE-7). The intermediate hops of the LSP are dynamically assigned by OSPF best route selection, thus S2L paths follow the IGP least cost path. Therefore, an MPLS path **loose** is configured without specifying any strict/loose hops.

```
*A:PE-1# configure router mpls
      path "loose"
      no shutdown
      exit
```

Creation of the P2MP LSP itself is done on the ingress LER or head-end node (PE-1 in our example) and can be seen in following CLI output. P2MP name is **LSP-p2mp-1**. A create time keyword **p2mp-lsp** is added in addition to the P2MP name to make a distinction in configuration between normal point-to-point LSPs and point-to-multipoint LSPs. A primary P2MP instance is initiated using the **primary-p2mp-instance** keyword accompanied with the P2MP instance name p-LSP-p2mp-1. Within this primary P2MP instance, the different S2Ls are defined using the **s2l-path** keyword. Be aware that the same MPLS path name can be used for different S2Ls as long as the destination is different (**to** command).

```
*A:PE-1# configure router mpls
      lsp "LSP-p2mp-1" p2mp-lsp
      cspf
      fast-reroute facility
      no node-protect
      exit
      primary-p2mp-instance "p-LSP-p2mp-1"
        s2l-path "loose" to 192.0.2.6
        exit
        s2l-path "loose" to 192.0.2.7
        exit
      exit
      no shutdown
      exit
```


On the head-end LER node of the P2MP LSP, several show commands can be used. A first set of show commands is used to verify the administrative and operational state of the P2MP LSP and its different S2L paths (including FRR bypass information). In this example, 'LSP-p2mp-1' P2MP LSP has two active S2L paths: one towards leaf node PE-6 and one to leaf node PE-7.

```
*A:PE-1# show router mpls p2mp-lsp
=====
MPLS P2MP LSPs (Originating)
=====
LSP Name                               Tun      Fastfail  Adm  Opr
                                   Id        Config
-----
LSP-p2mp-1                             1         Yes      Up   Up
-----
LSPs : 1
=====
*A:PE-1#
*A:PE-1# show router mpls p2mp-lsp "LSP-p2mp-1" detail
=====
MPLS P2MP LSPs (Originating) (Detail)
=====
Type : Originating
-----
LSP Name      : LSP-p2mp-1
LSP Type      : P2mpLsp                      LSP Tunnel ID : 1
From          : 192.0.2.1
Adm State     : Up                          Oper State    : Up2
LSP Up Time   : 0d 00:00:19                 LSP Down Time : 0d 00:00:00
Transitions   : 1                          Path Changes   : 1
Retry Limit   : 0                          Retry Timer    : 30 sec
Signaling     : RSVP                       Resv. Style    : SE
Hop Limit     : 255                       Negotiated MTU : n/a
Adaptive      : Enabled                    ClassType      : 0
FastReroute   : Enabled                    Oper FR        : Enabled
FR Method     : Facility                   FR Hop Limit   : 16
FR Node Pro*  : Disabled                   FR Prop Adm Grp: Disabled
FR Object     : Enabled                    ADSPEC         : Disabled
CSPF          : Enabled                    Use TE metric  : Disabled
Metric        : Disabled
Load Balanc*  : N/A
Include Grps  :                             Exclude Grps   :
None
Least Fill    : Disabled                    None

Revert Timer: Disabled                    Next Revert In : N/A
Auto BW      : Disabled
LdpOverRsvp  : Disabled                    VprnAutoBind   : Disabled
IGP Shortcut : Disabled                    BGP Shortcut    : Disabled
IGP LFA      : Disabled                    IGP Rel Metric  : Disabled
BGPTransTun  : Disabled
Oper Metric  : Disabled
Prop Adm Grp: Disabled
```

2.As long as one S2L path is operationally up (show router mpls p2mp-lsp lsp-name p2mp-instance instance-name) , the Oper State of the P2MP LSP is Up.

Configuring P2MP RSVP LSP

```

P2MPInstance: p-LSP-p2mp-1                P2MP-Inst-type : Primary
S2L Cfg Cou*: 2                            S2L Oper Count*: 2
S2l-Name   : loose                         To           : 192.0.2.6
S2l-Name   : loose                         To           : 192.0.2.7
=====
* indicates that the corresponding row element may have been truncated.
*A:PE-1#
*A:PE-1# show router mpls p2mp-info
=====
MPLS P2MP Cross Connect Information
=====
-----
S2L LSP-p2mp-1::loose
-----
Source IP Address   : 192.0.2.1           Tunnel ID       : 1
P2MP ID             : 0                   Lsp ID          : 31232
S2L Name            : LSP-p2mp-1::loose   To              : 192.0.2.6
Out Interface       : 1/1/2               Out Label       : 262143
Num. of S2ls        : 1
-----
S2L LSP-p2mp-1::loose
-----
Source IP Address   : 192.0.2.1           Tunnel ID       : 1
P2MP ID             : 0                   Lsp ID          : 31232
S2L Name            : LSP-p2mp-1::loose   To              : 192.0.2.7
Out Interface       : 1/1/1               Out Label       : 262143
Num. of S2ls        : 1
-----
P2MP Cross-connect instances : 2
=====
*A:PE-1#
*A:PE-1# show router mpls p2mp-lsp "LSP-p2mp-1" p2mp-instance "p-LSP-p2mp-1"
=====
MPLS P2MP Instance (Originating)
=====
-----
Type : Originating
-----
LSP Name      : LSP-p2mp-1
P2MP ID       : 0
Adm State     : Up
LSP Tunnel ID : 1
Oper State    : Up

P2MPInstance: p-LSP-p2mp-1                P2MP-Inst-type : Primary
P2MP Inst Id: 1                            P2MP Lsp Id     : 31232
Inst Admin    : Up                         Inst Oper       : Up
Inst Up Time: 0d 00:05:45                  Inst Dn Time    : 0d 00:00:00
Hop Limit     : 255                        Adaptive        : Enabled
Record Route: Record                       Record Label    : Record
Include Grps:                               Exclude Grps     :
None                                                  None
Bandwidth     : No Reservation              Oper Bw         : 0 Mbps
S2l-Name      : loose                       To              : 192.0.2.6
S2l Admin     : Up                         S2l Oper        : Up
S2l-Name      : loose                       To              : 192.0.2.7
S2l Admin     : Up                         S2l Oper        : Up
-----
P2MP instances : 1
=====
*A:PE-1#

```

FRR information can be displayed in detail for each S2L path. From this moment onward, the focus is on the S2L path towards PE-7. As you can see in the show command, link protection is present for links PE-1 <=> PE-3, PE-3 <=> PE-5 and PE-5 <=> PE-7 ('@'-reference inside show command).

```
*A:PE-1# show router mpls p2mp-lsp "LSP-p2mp-1" p2mp-instance "p-LSP-p2mp-1" s2l loose to
192.0.2.7 detail
=====
MPLS LSP LSP-p2mp-1 S2L loose (Detail)
=====
Legend :
    @ - Detour Available          # - Detour In Use
    b - Bandwidth Protected      n - Node Protected
    S - Strict                   L - Loose
    A - ABR
    s - Soft Preemption
=====
-----
LSP LSP-p2mp-1 S2L loose
-----
LSP Name       : LSP-p2mp-1          S2l LSP ID    : 31232
P2MP ID        : 0                  S2l Grp Id    : 2
Adm State      : Up                 Oper State     : Up
S2l State      : Active              :
S2L Name       : loose              To            : 192.0.2.7
S2l Admin      : Up                 S2l Oper      : Up
OutInterface    : 1/1/1              Out Label     : 262143
S2L Up Time    : 0d 00:08:07         S2L Dn Time   : 0d 00:00:00
RetryAttempt    : 0                  NextRetryIn   : 0 sec
S2L Trans       : 1                  CSPF Queries  : 1
Failure Code    : noError            Failure Node   : n/a
Inter-area      : False
ExplicitHops    :
    No Hops Specified
Actual Hops    :
    192.168.13.1 (192.0.2.1) @      Record Label   : N/A
    -> 192.168.13.2 (192.0.2.3) @    Record Label   : 262143
    -> 192.168.35.2 (192.0.2.5) @    Record Label   : 262143
    -> 192.168.57.2 (192.0.2.7)      Record Label   : 262143
ComputedHops    :
    192.168.13.1 (S)
    -> 192.168.13.2 (S)
    -> 192.168.35.2 (S)
    -> 192.168.57.2 (S)
LastResignal    : n/a
=====
*A:PE-1#
```

More in detail, **show router mpls bypass-tunnel** can be used. **Actual Hops** gives you the explicit hops of the bypass tunnel used to avoid link PE-1 <=> PE-3. On node PE-1 the MPLS path PE-1 <=> PE-2 <=> PE-3 is followed (see also [Figure 110](#)).

```
*A:PE-1# show router mpls bypass-tunnel detail
=====
MPLS Bypass Tunnels (Detail)
=====
```

Configuring P2MP RSVP LSP

```
-----
bypass-link192.168.13.2-61441
-----
To          : 192.168.23.2      State          : Up
Out I/F     : 1/1/2            Out Label     : 262141
Up Time    : 0d 00:09:51       Active Time    : n/a
Reserved BW : 0 Kbps           Protected LSP Count : 1
Type       : P2mp              Bypass Path Cost : 20
Setup Priority : 7              Hold Priority    : 0
Class Type  : 0
Exclude Node : None            Inter-Area      : False
Computed Hops :
    192.168.12.1 (S)           Egress Admin Groups : None
    -> 192.168.12.2 (S)         Egress Admin Groups : None
    -> 192.168.23.2 (S)         Egress Admin Groups : None
Actual Hops :
    192.168.12.1 (192.0.2.1)    Record Label     : N/A
    -> 192.168.12.2 (192.0.2.2) Record Label     : 262141
    -> 192.168.23.2 (192.0.2.3) Record Label     : 262142
-----
---snipped---
```

On node PE-3 the MPLS path PE-3 <=> PE-2 <=> PE-4 <=> PE-5 is followed (see [Figure 110 on page 659](#)) to avoid link PE-3 <=> PE-5.

```
*A:PE-3# show router mpls bypass-tunnel protected-lsp p2mp detail
=====
MPLS Bypass Tunnels (Detail)
=====
-----
bypass-link192.168.35.2-61441
-----
To          : 192.168.45.2      State          : Up
Out I/F     : 1/1/3            Out Label     : 262142
Up Time    : 0d 00:13:14       Active Time    : n/a
Reserved BW : 0 Kbps           Protected LSP Count : 1
Type       : P2mp              Bypass Path Cost : 30
Setup Priority : 7              Hold Priority    : 0
Class Type  : 0
Exclude Node : None            Inter-Area      : False
Computed Hops :
    192.168.23.2 (S)           Egress Admin Groups : None
    -> 192.168.23.1 (S)         Egress Admin Groups : None
    -> 192.168.24.2 (S)         Egress Admin Groups : None
    -> 192.168.45.2 (S)         Egress Admin Groups : None
Actual Hops :
    192.168.23.2 (192.0.2.3)    Record Label     : N/A
    -> 192.168.23.1 (192.0.2.2) Record Label     : 262142
    -> 192.168.24.2 (192.0.2.4) Record Label     : 262141
    -> 192.168.45.2 (192.0.2.5) Record Label     : 262142

Protected LSPs -
LSP Name    : LSP-p2mp-1::loose
From        : 192.0.2.1        To              : 192.0.2.7
Avoid Node/Hop : 192.168.35.2 Downstream Label    : 262143
Bandwidth   : 0 Kbps
```

```
=====
*A:PE-3#
```

A similar output can be seen on PE-5 node also. The MPLS path PE-5 <=> PE-4 <=> PE-6 <=> PE-7 is followed (see also [Figure 110](#)).

On the transit LSRs and egress LER/leaf node (see also [Figure 110](#)), the **show router mpls p2mp-info** command can be used. Attached is the show command on PE-3 node included for S2L path to 192.0.2.7. Similar outputs are possible for nodes PE-5 and PE-7.

```
*A:PE-3# show router mpls p2mp-info
  - p2mp-info [type {originate|transit|terminate}] [s2l-endpoint <ip-address>]

  <originate|transit*> : keywords
  <ip-address>         : [a.b.c.d]

*A:PE-3# show router mpls p2mp-info
=====
MPLS P2MP Cross Connect Information
=====
-----
S2L LSP-p2mp-1::loose
-----
-----
Source IP Address   : 192.0.2.1           Tunnel ID       : 1
P2MP ID             : 0                  Lsp ID         : 31232
S2L Name            : LSP-p2mp-1::loose   To             : 192.0.2.7
Out Interface       : 1/1/1              Out Label      : 262143
Num. of S2ls        : 1
-----
P2MP Cross-connect instances : 1
=====
*A:PE-3#
```

Mapping Multicast Traffic

To map multicast traffic into the LSP tree from the head-end node until leaf node, PIM and IGMP configurations are needed on the head-end node (PE-1) and leaf nodes (PE-6 and PE-7) of the P2MP RSVP LSP. The intermediate nodes (transit LSR, branch LSR and bud LSR) do not need any explicit configuration for that.

Head-end Node (Ingress LER)

PIM must be enabled on the interface towards the MC source and PIM must be enabled on the tunnel interface. A tunnel interface should be seen as an internal representation of a specific P2MP LSP. Creation is done within the PIM context using the **tunnel-interface rsvp-p2mp** command followed by the P2MP LSP name. Translated into configuration commands, this becomes:

```
*A:PE-1# configure router pim
      interface "int-PE-1-MC-source"
      exit
      tunnel-interface rsvp-p2mp "LSP-p2mp-1"
```

In the data path, when a multicast packet is received on an interface, a successful Reverse Path Forwarding (RPF) check must be done for the source address otherwise the packet will be dropped.

Besides enabling PIM on the tunnel interface also IGMP is enabled to do a static <S,G> or <*,G> join of a multicast group address (227.1.1.1 in our example) to the tunnel interface/P2MP LSP. Be aware that there is always a one-to-one mapping between <S,G> or <*,G> and a tunnel interface/P2MP LSP. In our example a < S,G > will be configured. A <*,G> join scenario is included in [Additional Topics on page 674](#).

```
*A:PE-1# configure router igmp      ...
      tunnel-interface rsvp-p2mp "LSP-p2mp-1"
      static
        group 227.1.1.1
        source 192.168.9.2
      exit
    exit
  exit
no shutdown
```

The **show router pim tunnel-interface** command shows you the admin state of the tunnel interface and an association to an internal local ifindex (**73728** in the example).

```
*A:PE-1# show router pim tunnel-interface
=====
PIM Interfaces ipv4
```

```

=====
Interface                      Originator Address  Adm  Opr  Transport Type
-----
mpls-if-73728                  N/A                Up   Up   Tx-IPMSI
-----
Interfaces : 1
=====
*A:PE-1#

```

With **show router igmp group** you see the configured <S,G> entry and outgoing interface (= tunnel interface), represented by mpls-if-73728.

```

*A:PE-1# show router igmp group 227.1.1.1
=====
IGMP Interface Groups
=====
(192.168.9.2,227.1.1.1)                                UpTime: 0d 00:11:10
  Fwd List   : mpls-if-73728
-----
Entries : 1
=====
IGMP Host Groups
=====
No Matching Entries
=====
IGMP SAP Groups
=====
No Matching Entries
=====
*A:PE-1#

```

At this moment in time, users can verify if multicast traffic is using P2MP LSP at the head-end node using the **show router pim group group-address detail** command.

```

*A:PE-1# show router pim group 227.1.1.1 detail
=====
PIM Source Group ipv4
=====
Group Address      : 227.1.1.1
Source Address     : 192.168.9.2
RP Address         : 0
Advt Router        : 192.0.2.1
Flags              :                               Type           : (S,G)
MRIB Next Hop      : 192.168.9.2
MRIB Src Flags     : direct
Keepalive Timer    : Not Running
Up Time            : 0d 00:02:42      Resolved By           : rtable-u

Up JP State        : Joined           Up JP Expiry          : 0d 00:00:00
Up JP Rpt          : Not Joined StarG  Up JP Rpt Override   : 0d 00:00:00

Register State     : No Info
Reg From Anycast RP: No

Rpf Neighbor       : 192.168.9.2
Incoming Intf      : int-PE-1-MC-source

```

Mapping Multicast Traffic

```
Outgoing Intf List : mpls-if-73728
```

```
Curr Fwding Rate   : 5357.3 kbps
Forwarded Packets  : 124213
Forwarded Octets   : 5713798
Spt threshold      : 0 kbps
Admin bandwidth    : 1 kbps
Discarded Packets  : 0
RPF Mismatches     : 0
ECMP opt threshold : 7
```

```
-----
Groups : 1
=====
```

```
*A:PE-1#
```

Leaf Node (Egress LER)

In the PIM context, the same tunnel interface must be created as the head-end node with in addition an explicit reference to the head-end system address, translated into the **sender systemIP_head-end_node** parameter.

```
*A:PE-7# configure router pim
      tunnel-interface rsvp-p2mp LSP-p2mp-1 sender 192.0.2.1
```

The **show router pim tunnel-interface** command shows you the admin state of the tunnel interface and an association to an internal local ifindex (73728 in this example, by coincidence the same ifindex as the one on the head-end node/PE-1).

```
*A:PE-7# show router pim tunnel-interface
```

```
=====
PIM Interfaces ipv4
```

```
=====
Interface                               Originator Address  Adm  Opr  Transport Type
-----
mpls-if-73728                           N/A                 Up   Up   Tx-IPMSI
-----
```

```
Interfaces : 1
=====
```

```
*A:PE-7#
```

The main goal on the leaf node(s) is to get traffic off the P2MP LSP/tunnel interface. This is done using a multicast information policy (*multicast-info-policy*). Inside this MC policy, a range of multicast group addresses must be defined under a bundle context (*bundle1*) in order to see traffic (*channel*). Also inside the bundle context, the P2MP LSP is presented by the tunnel interface (*primary-tunnel-interface*). Translated into configuration commands, this becomes:

```
*A:PE-7# configure mcast-management
      multicast-info-policy "p2mp-pol" create
      bundle "bundle1" create
      primary-tunnel-interface rsvp-p2mp LSP-p2mp-1 sender 192.0.2.1
      channel "227.1.1.1" "227.1.1.1" create
      exit
exit
```



```

        bundle "default" create
    exit
exit

```

Note: The **channel** command must be seen as a range command with a start-mc-group-address and an end-mc-group-address. In our example, only one MC group address, 227.1.1.1 is seen.

The configured multicast information policy must be applied to the base router instance.

```
*A:PE-7# configure router multicast-info-policy "p2mp-pol"
```

On the leaf node (PE-7/PE-6), MC clients are connected. IGMP is enabled on those MC clients with a static <S,G> join to redirect MC traffic downstream to the MC client. Translated into configuration commands, this becomes:

```

*A:PE-7# configure router igmp
    interface "int-PE-7-MC-client1"
        static
            group 227.1.1.1
            source 192.168.9.2
        exit
    exit
exit

```

With **show router igmp group** you see the configured <S,G> entry and outgoing interface (= int-PE-7-MC-client1).

```

*A:PE-7# show router igmp group 227.1.1.1
=====
IGMP Interface Groups
=====
(192.168.9.2,227.1.1.1)                               UpTime: 0d 00:09:02
  Fwd List   : int-PE-7-MC-client1
-----
Entries : 1
=====
IGMP Host Groups
=====
No Matching Entries
=====
IGMP SAP Groups
=====
No Matching Entries
=====
*A:PE-7#

```

Now, users can verify if multicast traffic is sent to the MC client using the **show router pim group group-address detail** command

```

*A:PE-7# show router pim group 227.1.1.1 detail
=====
PIM Source Group ipv4

```

Mapping Multicast Traffic

```
=====
Group Address      : 227.1.1.1
Source Address     : 192.168.9.2
RP Address         : 0
Advt Router       :
Flags             :                               Type           : (S,G)
MRIB Next Hop     :
MRIB Src Flags    : remote
Keepalive Timer   : Not Running
Up Time           : 0d 00:01:39      Resolved By          : unresolved

Up JP State       : Joined           Up JP Expiry          : 0d 00:00:21
Up JP Rpt         : Not Joined StarG Up JP Rpt Override   : 0d 00:00:00

Register State    : No Info
Reg From Anycast RP: No

Rpf Neighbor      :
Incoming Intf     : mpls-if-73728
Outgoing Intf List : int-PE-7-MC-client1

Curr Fwding Rate  : 4590.8 kbps
Forwarded Packets : 226349           Discarded Packets    : 0
Forwarded Octets  : 10412054        RPF Mismatches       : 0
Spt threshold     : 0 kbps           ECMP opt threshold   : 7
Admin bandwidth   : 1 kbps

-----
Groups : 1
=====
*A:PE-7#
```

OAM Tool

P2P LSP operation and maintenance (OAM) commands (**oam lsp-ping** and **oam lsp-trace**) are extended for P2MP LSP. The user can instruct the head-end node to generate an P2MP LSP ping or a P2MP LSP trace by entering the command **oam p2mp-lsp-ping** or **oam p2mp-lsp-trace**. The P2MP OAM extensions are defined in **draft-ietf-mppls-p2mp-lsp-ping**.

For P2MP LSP ping, the echo request is sent on the active P2MP instance and is replicated in the data path over all branches of the P2MP LSP instance. By default, all egress LER nodes which are leaves of the P2MP LSP instance will reply. Echo reply messages can be reduced by configuring the **s2l-dest-address** (a maximum of up to five egress nodes in a single run of the OAM command). Replies are sent by IP.

```
*A:PE-1# oam p2mp-lsp-ping
- p2mp-lsp-ping {<lsp-name> [p2mp-instance <instance-name> [s2l-dest-address
    <ipv4-address> [... up to 5]]] [ttl <label-ttl>]}
- p2mp-lsp-ping {ldp <p2mp-identifier> [sender-addr <ipv4-address>] [leaf-addr
    <ipv4-address> [... up to 5]]}
- p2mp-lsp-ping {ldp-ssm source <ip-address> group <ip-address> [router
    <router-instance>|service-name <service-name>] [sender-addr
    <ipv4-address>] [leaf-addr <ipv4-address> [... up to 5]]}
- options common to all p2mp-lsp-ping cases: [fc <fc-name> [profile {in|out}]]
    [size <octets>] [timeout <timeout>] [detail]
```

```
<lsp-name>          : [64 chars max]
<instance-name>     : [32 chars max]
<ipv4-address>       : a.b.c.d
<in|out>             : in|out - Default: out
<fc-name>           : be|l2|af|l1|h2|ef|h1|nc - Default: be
<octets>            : [1..9198] - Default: 1
<label-ttl>         : [1..255] - Default: 255
<timeout>           : [1..120] seconds - Default: 10
<detail>            : keyword - displays detailed information
<p2mp-identifier>    : [1..4294967295]
<ldp-ssm>           : keyword
<ip-address>        : ipv4-address - a.b.c.d
                     : ipv6-address - x:x:x:x:x:x:x:x (eight 16-bit pieces)
                     :               x:x:x:x:x:x:d.d.d.d
                     :               x - [0..FFFF]H
                     :               d - [0..255]D
<router-instance>   : <router-name>|<service-id>
                     : router-name - "Base"|"management" Default - Base
                     : service-id  - [1..2147483647]
<service-name>      : [64 chars max]
```

```
*A:PE-1#
```

```
*A:PE-1# oam p2mp-lsp-ping "LSP-p2mp-1" detail
P2MP LSP LSP-p2mp-1: 92 bytes MPLS payload
```

```
=====
S2L Information
=====
```

```

From           RTT           Return Code
-----
192.0.2.6      =5.91ms      EgressRtr(3)
192.0.2.7      =6.07ms      EgressRtr(3)
=====

Total S2L configured/up/responded = 2/2/2,
      round-trip min/avg/max   = 5.91 / 5.99 / 6.07 ms

Responses based on return code:
      EgressRtr(3)=2

*A:PE-1#

```

Note: Return codes are based on RFC4379. Value 3 means the replying router is an egress for the FEC at stack-depth.

P2MP LSP trace allows the user to trace the path of a single S2L path of a P2MP LSP from head-end node to leaf node. By the use of the downstream mapping TLV, each node along the S2L path can fill in the appropriate flags : B or E flag. The B-flag is set when the responding node is a branch LSR and the E-flag is set when the responding node is an egress LER.

```

*A:PE-1# oam p2mp-lsp-trace
- p2mp-lsp-trace <lsp-name> p2mp-instance <instance-name> s2l-dest-address
  <ip-address> [fc <fc-name> [profile {in|out}]] [size <octets>] [max-fail
  <no-response-count>] [probe-count <probes-per-hop>] [min-ttl <min-label-ttl>]
  [max-ttl <max-label-ttl>] [timeout <timeout>] [interval <interval>] [detail]

<lsp-name>           : [64 chars max]
<instance-name>      : [32 chars max]
<ip-address>         : ipv4 address   a.b.c.d
<fc-name>            : be|l2|af|l1|h2|ef|h1|nc - Default: be
<in|out>             : in|out - Default: out
<octets>             : [1..9198] - Default: 1
<no-response-count>  : [1..10] - Default: 5
<probes-per-hop>     : [1..10] - Default: 1
<min-label-ttl>      : [1..255] - Default: 1
<max-label-ttl>      : [1..255] - Default: 30
<timeout>            : [1..60] seconds - Default: 3
<detail>             : keyword - displays detailed information
<interval>           : [1..10] seconds - Default: 1

*A:PE-1#

*A:PE-1# oam p2mp-lsp-trace "LSP-p2mp-1" p2mp-instance "p-LSP-p2mp-1" s2l-dest-address
192.0.2.7 detail
P2MP LSP LSP-p2mp-1: 132 bytes MPLS payload
P2MP Instance p-LSP-p2mp-1, S2L Egress 192.0.2.7

1 192.0.2.3 rtt=2.10 ms rc=8(DSRtrMatchLabel)
   DS 1: ipaddr=192.168.35.2 ifaddr=192.168.35.2 iftype=ipv4Numbered MRU=1564
   label=262143 proto=4(RSVP-TE) B/E flags:0/0
2 192.0.2.5 rtt=3.32 ms rc=8(DSRtrMatchLabel)
   DS 1: ipaddr=192.168.57.2 ifaddr=192.168.57.2 iftype=ipv4Numbered MRU=1564
   label=262143 proto=4(RSVP-TE) B/E flags:0/0
3 192.0.2.7 rtt=4.55 ms rc=3(EgressRtr)

```

*A:PE-1#

Note: Return codes are based on RFC4379. Value 8 means that the label is switched at stack-depth. This is the case for a transit LSR, doing MPLS label swapping. No B or E flag is set.

Additional Topics

<*,G> IGMP join instead of <S,G> IGMP join

In the [Head-end Node \(Ingress LER\) on page 666](#) and [Leaf Node \(Egress LER\) on page 668](#) steps, a source specific IGMP join (<S,G> join) was used at the head-end node and leaf nodes. Another possibility is to use a source unknown or starg IGMP join (<*,G> join). When doing the latter, a rendezvous point (RP) must be defined in the PIM network. The RP allows multicast data flows between sources and receivers to meet at a predefined network location (in this example, the loopback address of node PE-1). It must be seen as an intermediate device to establish a multicast flow.

The RP can be defined in a dynamic way (BSR protocol) or a static way. In this example the static way is chosen meaning that on all involved PIM nodes, the RP address will be statically configured. The following configuration is needed on head-end and leaf nodes.

```
*A:PE-1/PE-6/PE-7# configure router pim
      rp
        static
          address 192.0.2.1
          group-prefix 227.1.1.1/32
        exit
      exit
```

The **group-prefix** is a mandatory keyword. It references a group address or group address range for which this rendez-vous point will be used.

```
*A:PE-1/PE-6/PE_7# show router pim rp
=====
PIM RP Set ipv4
=====
Group Address                                     Hold Expiry
  RP Address                                     Type      Prio Time  Time
-----
227.1.1.1/32
  192.0.2.1                                     Static    1      N/A   N/A
-----
Group Prefixes : 1
=====
*A:PE-1#
```

As previously mentioned, the configuration of the <*,G> IGMP join is done on the head-end node (PE-1) and leaf nodes (PE-6 and PE-7)

```
*A:PE-1# configure router igmp
      tunnel-interface rsvp-p2mp "LSP-p2mp-1"
        no shutdown
        static
          group 227.1.1.1
```

```

                starg
            exit
        exit
    exit

*A:PE-6# configure router igmp
    interface "int-PE-6-MC-client2"
        static
            group 227.1.1.1
                starg
            exit
        exit
    exit

*A:PE-7# configure router igmp
    interface "int-PE-7-MC-client1"
        static
            group 227.1.1.1
                starg
            exit
        exit
    exit

```

The same previous **show** command can be used to verify the multicast traffic on head-end node and leaf nodes, **show router igmp group 227.1.1.1** and **show router pim group 227.1.1.1 detail**.

```

*A:PE-7# show router igmp group 227.1.1.1
=====
IGMP Interface Groups
=====

(*,227.1.1.1)                                UpTime: 0d 00:06:58
    Fwd List  : int-PE-7-MC-client1
-----
Entries : 1
=====
IGMP Host Groups
=====
No Matching Entries
=====
IGMP SAP Groups
=====
No Matching Entries
=====

*A:PE-7#
*A:PE-7# show router pim group 227.1.1.1 detail
=====
PIM Source Group ipv4
=====
Group Address      : 227.1.1.1
Source Address     : *
RP Address         : 192.0.2.1
Advt Router        :
Flags              :                               Type           : (*,G)

```

Additional Topics

```
MRIB Next Hop      :
MRIB Src Flags     : remote
Keepalive Timer    : Not Running
Up Time           : 0d 00:05:34      Resolved By       : unresolved

Up JP State        : Joined           Up JP Expiry       : 0d 00:00:25
Up JP Rpt          : Not Joined StarG Up JP Rpt Override : 0d 00:00:00

Rpf Neighbor       :
Incoming Intf      : mpls-if-73728
Outgoing Intf List : int-PE-7-MC-client1

Curr Fwding Rate   : 0.0 kbps
Forwarded Packets  : 31               Discarded Packets  : 0
Forwarded Octets   : 1426             RPF Mismatches     : 0
Spt threshold      : 0 kbps           ECMP opt threshold : 7
Admin bandwidth    : 1 kbps
=====
PIM Source Group ipv4
=====
Group Address      : 227.1.1.1
Source Address     : 192.168.9.2
RP Address         : 192.0.2.1
Advt Router        :
Flags              : spt              Type               : (S,G)
MRIB Next Hop      :
MRIB Src Flags     : remote
Keepalive Timer Exp: 0d 00:01:27
Up Time           : 0d 00:05:34      Resolved By       : unresolved

Up JP State        : Joined           Up JP Expiry       : 0d 00:00:25
Up JP Rpt          : Not Pruned       Up JP Rpt Override : 0d 00:00:00

Register State     : No Info
Reg From Anycast RP: No

Rpf Neighbor       :
Incoming Intf      : mpls-if-73728
Outgoing Intf List : int-PE-7-MC-client1

Curr Fwding Rate   : 3901.2 kbps
Forwarded Packets  : 3539800          Discarded Packets  : 0
Forwarded Octets   : 162830800        RPF Mismatches     : 0
Spt threshold      : 0 kbps           ECMP opt threshold : 7
Admin bandwidth    : 1 kbps
-----
Groups : 2
=====
*A:PE-7#
```


Influence IGP Metric

Suppose that the IGP metric is increased on all links pointing to/from PE-2 and on the link between PE-5 and PE-7.

```
*A:PE-1# configure router ospf area 0 interface "int-PE-1-PE-2" metric 10000

*A:PE-2# configure router ospf area 0 interface "int-PE-2-PE-1" metric 10000
*A:PE-2# configure router ospf area 0 interface "int-PE-2-PE-3" metric 10000
*A:PE-2# configure router ospf area 0 interface "int-PE-2-PE-4" metric 10000

*A:PE-3# configure router ospf area 0 interface "int-PE-3-PE-2" metric 10000

*A:PE-4# configure router ospf area 0 interface "int-PE-4-PE-2" metric 10000

*A:PE-5# configure router ospf area 0 interface "int-PE-5-PE-7" metric 10000

*A:PE-7# configure router ospf area 0 interface "int-PE-7-PE-5" metric 10000
```

The existing P2MP LSP *LSP-p2mp-1* will not take into account these new constraints. The two S2L paths (one *loose* towards PE-6 and another one *loose* towards PE-7) are calculated using the default OSPF metric. What we can do to trigger MPLS to re-compute the S2L paths, is configure a p2mp-resignal-timer on the head-end node inside the global MPLS context. In this way, each time this timer expires (in our example, every 60 minutes), MPLS will trigger CSPF to re-compute the whole set of S2L paths of all active P2MP instances. MPLS performs a global make-before-break (MBB) and moves each S2L sub-LSP in the instance into its new path using a new P2MP LSP ID if the global MBB is successful. **show router mpls status** gives you an indication when the P2MP resignal timer will expire and which types of LSPs are setup on the node.

```
*A:PE-1# configure router mpls p2mp-resignal-timer 60

*A:PE-1# show router mpls status
=====
MPLS Status
=====
Admin Status           : Up           Oper Status           : Up
Oper Down Reason       : n/a
FRR Object             : Enabled    Resignal Timer        : Disabled
Hold Timer             : 1 seconds Next Resignal         : N/A
Srlg Frr               : Disabled    Srlg Frr Strict       : Disabled
Admin Group Frr        : Disabled
Dynamic Bypass         : Enabled    User Srlg Database    : Disabled
BypassResignalTimer    : Disabled  BypassNextResignal    : N/A
LeastFill Min Thd      : 5 percent LeastFill Reopti Thd  : 10 percent
Local TTL Prop         : Enabled    Transit TTL Prop      : Enabled
AB Sample Multiplier   : 1         AB Adjust Multiplier  : 288
Exp Backoff Retry      : Disabled   CSPF On Loose Hop     : Disabled
Lsp Init RetryTimeout  : 30 seconds MBB Pref Current Hops : Disabled
Logger Event Bundling  : Disabled
RetryIgpOverload       : Disabled

P2mp Resignal Timer    : 60 minutes  P2mp Next Resignal    : 48 minutes
```

Additional Topics

```

Sec FastRetryTimer      : Disabled      Static LSP FR Timer    : 30 seconds
P2P Max Bypass Association: 1000
P2PActPathFastRetry     : Disabled      P2MP S2L Fast Retry   : Disabled
In Maintenance Mode     : No
MplsTp                  : Disabled

```

---snipped---

=====

As an alternative the user can also perform a manual resignal of a P2MP instance on the head-end node using a tools command.

```
*A:PE-1# tools perform router mpls resignal p2mp-lsp "LSP-p2mp-1" p2mp-instance"p-LSP-
p2mp-1"
```

```
*A:PE-1# tools perform router mpls resignal p2mp-delay 0
```

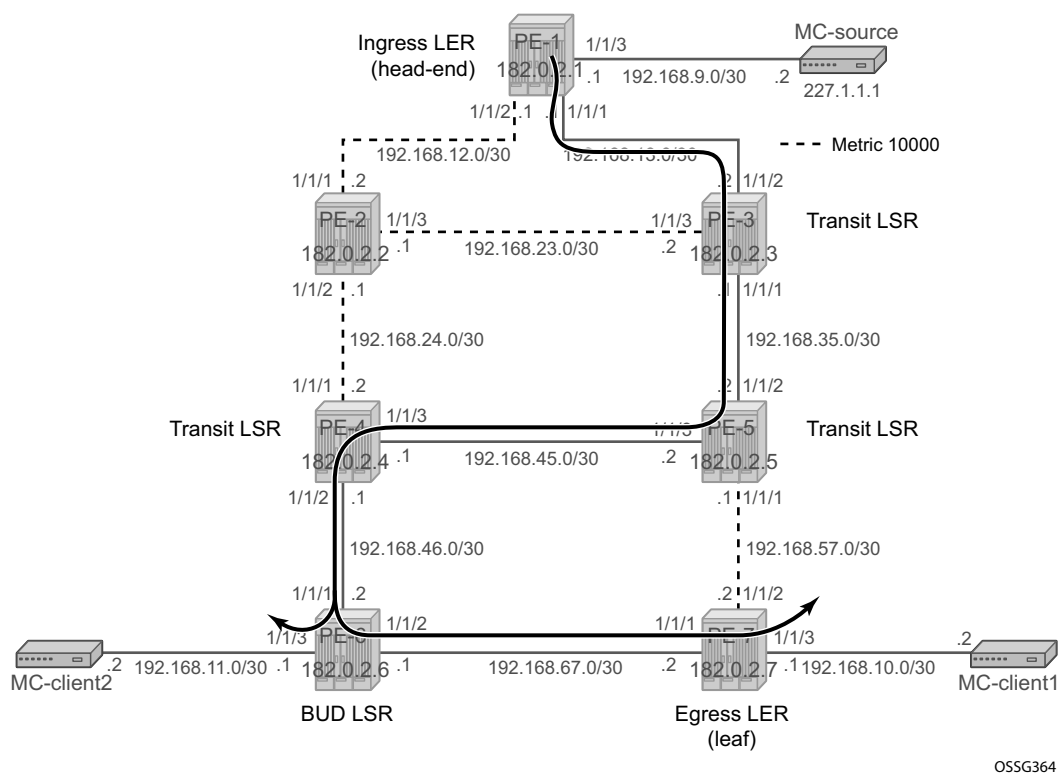


Figure 111: P2MP LSP p-to-mp-1 with Metric Change

After an instantaneous tools resignal command which is executed with no delay (p2mp-delay 0), the S2L paths can be verified, also check [Figure 111](#). Node PE-6 is acting now as bud LSR node (instead of egress LER before).

RSVP Signaled Point-to-Multipoint LSPs

```
*A:PE-1# show router mpls p2mp-lsp "LSP-p2mp-1" p2mp-instance "p-LSP-p2mp-1" s2l loose to
192.0.2.7 detail
```

```
=====
MPLS LSP LSP-p2mp-1 S2L loose (Detail)
```

```
Legend :
```

```

@ - Detour Available          # - Detour In Use
b - Bandwidth Protected      n - Node Protected
S - Strict                   L - Loose
A - ABR
s - Soft Preemption
```

```
=====
LSP LSP-p2mp-1 S2L loose
```

```
-----
LSP Name       : LSP-p2mp-1                S2l LSP ID    : 53764
P2MP ID        : 0                        S2l Grp Id    : 1
Adm State      : Up                       Oper State    : Up
S2l State      : Active                   :
S2L Name       : loose                    To           : 192.0.2.7
S2l Admin      : Up                       S2l Oper      : Up
OutInterface   : 1/1/1                    Out Label     : 262143
S2L Up Time    : 0d 04:45:36              S2L Dn Time   : 0d 00:00:00
RetryAttempt   : 0                        NextRetryIn   : 0 sec
S2L Trans      : 3                       CSPF Queries  : 3
Failure Code    : noError                 Failure Node   : n/a
Inter-area     : False
ExplicitHops    :
    No Hops Specified
Actual Hops :
    192.168.13.1 (192.0.2.1) @           Record Label  : N/A
    -> 192.168.13.2 (192.0.2.3) @         Record Label  : 262143
    -> 192.168.35.2 (192.0.2.5) @         Record Label  : 262143
    -> 192.168.45.1 (192.0.2.4) @         Record Label  : 262143
    -> 192.168.46.2 (192.0.2.6) @         Record Label  : 262143
    -> 192.168.67.2 (192.0.2.7) @         Record Label  : 262143
ComputedHops:
    192.168.13.1(S)
    -> 192.168.13.2(S)
    -> 192.168.35.2(S)
    -> 192.168.45.1(S)
    -> 192.168.46.2(S)
    -> 192.168.67.2(S)
LastResignal: n/a
=====
```

```
*A:PE-1#
```

```
*A:PE-1# show router mpls p2mp-lsp "LSP-p2mp-1" p2mp-instance "p-LSP-p2mp-1" s2l loose to
192.0.2.6 detail
```

```
=====
MPLS LSP LSP-p2mp-1 S2L loose (Detail)
```

```
Legend :
```

```

@ - Detour Available          # - Detour In Use
b - Bandwidth Protected      n - Node Protected
S - Strict                   L - Loose
A - ABR
s - Soft Preemption
```

Additional Topics

```
LSP LSP-p2mp-1 S2L loose
-----
LSP Name      : LSP-p2mp-1                S2l LSP ID   : 53764
P2MP ID       : 0                        S2l Grp Id  : 2
Adm State     : Up                      Oper State   : Up
S2l State:    : Active                  :
S2L Name      : loose                   To          : 192.0.2.6
S2l Admin     : Up                     S2l Oper    : Up
OutInterface: 1/1/1                   Out Label   : 262143
S2L Up Time   : 0d 04:45:02           S2L Dn Time : 0d 00:00:00
RetryAttempt: 0                      NextRetryIn  : 0 sec
S2L Trans     : 3                     CSPF Queries: 4
Failure Code: noError                Failure Node: n/a
Inter-area    : False
ExplicitHops:
    No Hops Specified
Actual Hops :
    192.168.13.1 (192.0.2.1) @          Record Label : N/A
-> 192.168.13.2 (192.0.2.3) @          Record Label   : 262143
-> 192.168.35.2 (192.0.2.5) @          Record Label   : 262143
-> 192.168.45.1 (192.0.2.4) @          Record Label   : 262143
-> 192.168.46.2 (192.0.2.6)          Record Label   : 262143
ComputedHops:
    192.168.13.1(S)
-> 192.168.13.2(S)
-> 192.168.35.2(S)
-> 192.168.45.1(S)
-> 192.168.46.2(S)
LastResignal: n/a
=====
*A:PE-1#
```

An **oam p2mp-lsp-trace** command toward PE-7 will now set the E flag on PE-6 since that PE acts also as an egress LER node.

```
*A:PE-1# oam p2mp-lsp-trace "LSP-p2mp-1" p2mp-instance "p-LSP-p2mp-1" s2l-dest-address
192.0.2.7 detail
P2MP LSP LSP-p2mp-1: 132 bytes MPLS payload
P2MP Instance p-LSP-p2mp-1, S2L Egress 192.0.2.7

 1 192.0.2.3  rtt=2.05 ms rc=8(DSRtrMatchLabel)
    DS 1: ipaddr=192.168.35.2 ifaddr=192.168.35.2 iftype=ipv4Numbered MRU=1564
    label=262143 proto=4(RSVP-TE) B/E flags:0/0
 2 192.0.2.5  rtt=3.25 ms rc=8(DSRtrMatchLabel)
    DS 1: ipaddr=192.168.45.1 ifaddr=192.168.45.1 iftype=ipv4Numbered MRU=1564
    label=262143 proto=4(RSVP-TE) B/E flags:0/0
 3 192.0.2.4  rtt=4.88 ms rc=8(DSRtrMatchLabel)
    DS 1: ipaddr=192.168.46.2 ifaddr=192.168.46.2 iftype=ipv4Numbered MRU=1564
    label=262143 proto=4(RSVP-TE) B/E flags:0/0
 4 192.0.2.6  rtt=7.79 ms rc=8(DSRtrMatchLabel)
    DS 1: ipaddr=192.168.67.2 ifaddr=192.168.67.2 iftype=ipv4Numbered MRU=1564
    label=262143 proto=4(RSVP-TE) B/E flags:0/1
 5 192.0.2.7  rtt=67.2 ms rc=3(EgressRtr)

*A:PE-1#
```

As a next step, the S2L path towards PE-7 is changed from **loose** to a **strict** direct MPLS path (**strict-to-PE-7**). In that way, OSPF is not calculating anymore the shortest path to the leaf node.

```
*A:PE-1# configure router mpls
      path "path-strict-to-PE-7"
        hop 10 192.168.13.2 strict
        hop 20 192.168.35.2 strict
        hop 30 192.168.57.2 strict
      no shutdown
    exit
```

Before applying this new S2L path to the existing P2MP LSP (*LSP-p2mp-1*), the existing S2L path towards PE-7 must be removed.

```
*A:PE-1# configure router mpls lsp "LSP-p2mp-1" primary-p2mp-instance "p-LSP-p2mp-1"
      s2l-path "loose" to 192.0.2.7 shutdown
      no s2l-path "loose" to 192.0.2.7
      s2l-path "path-strict-to-PE-7" to 192.0.2.7
    exit
  exit
```

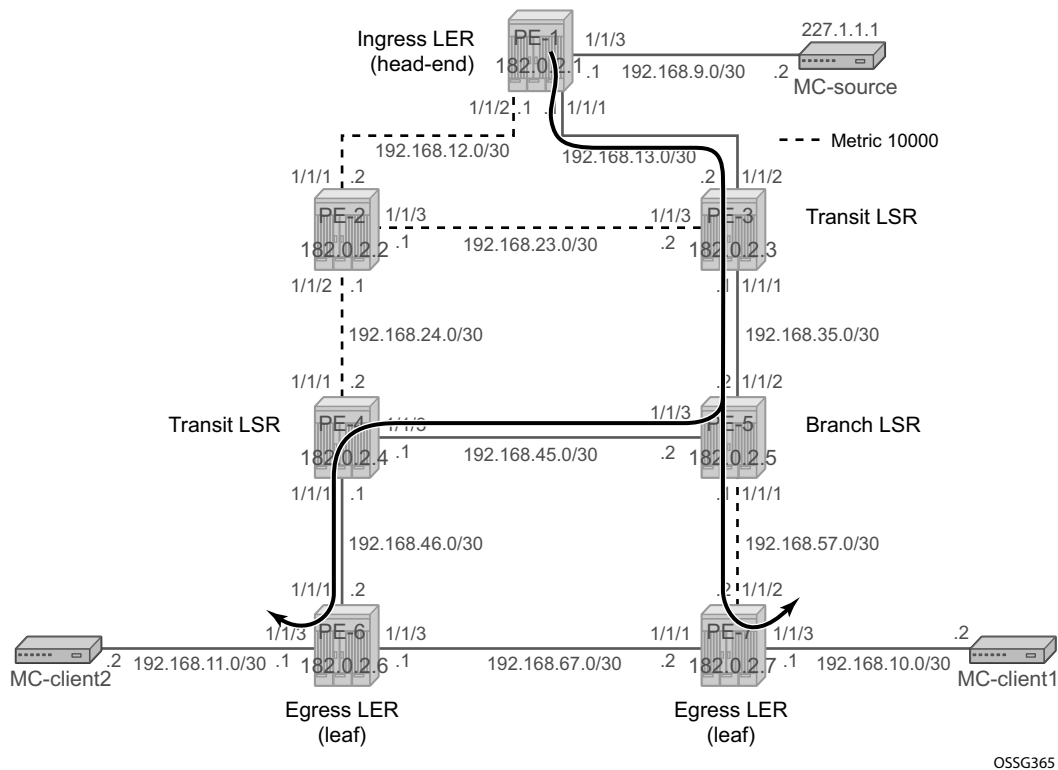


Figure 112: P2MP LSP LSP-p2mp-1 with Strict S2L Path Towards PE-7

As a consequence of this, only the **S2l Grp Id** has changed while **S2L LSP ID** remains the same as before. Now, S2L paths can be verified according to [Figure 112](#). PE-5 is acting now as a branch LSR node (instead of a transit LSR before).

```
*A:PE-1# show router mpls p2mp-lsp "LSP-p2mp-1" p2mp-instance "p-LSP-p2mp-1" s2l path-strict-to-PE-7 to 192.0.2.7 detail
=====
MPLS LSP LSP-p2mp-1 S2L path-strict-to-PE-7 (Detail)
=====
Legend :
    @ - Detour Available           # - Detour In Use
    b - Bandwidth Protected       n - Node Protected
    S - Strict                     L - Loose
    A - ABR
    s - Soft Preemption
=====
-----
LSP LSP-p2mp-1 S2L path-strict-to-PE-7
-----
LSP Name       : LSP-p2mp-1           S2l LSP ID    : 53764
P2MP ID        : 0                   S2l Grp Id    : 3
Adm State      : Up                   Oper State    : Up
S2l State      : Active               :
S2L Name       : path-strict-to-PE-7  To           : 192.0.2.7
S2l Admin      : Up                   S2l Oper      : Up
OutInterface    : 1/1/1                Out Label     : 262143
S2L Up Time    : 0d 00:09:25           S2L Dn Time   : 0d 00:00:00
RetryAttempt    : 0                     NextRetryIn   : 0 sec
S2L Trans      : 1                     CSPF Queries  : 1
Failure Code    : noError               Failure Node   : n/a
Inter-area      : False
ExplicitHops:
    192.168.13.2(S)    -> 192.168.35.2(S)    -> 192.168.57.2(S)
Actual Hops :
    192.168.13.1 (192.0.2.1) @           Record Label   : N/A
-> 192.168.13.2 (192.0.2.3) @           Record Label   : 262143
-> 192.168.35.2 (192.0.2.5) @           Record Label   : 262143
-> 192.168.57.2 (192.0.2.7)             Record Label   : 262142
ComputedHops:
    192.168.13.1(S)
-> 192.168.13.2(S)
-> 192.168.35.2(S)
-> 192.168.57.2(S)
LastResignal: n/a
=====
*A:PE-1#
```

An **oam p2mp-lsp-trace** command towards PE-7 will now set the B flag on PE-5 since that became a branch LSR now.

```
*A:PE-1# oam p2mp-lsp-trace "LSP-p2mp-1" p2mp-instance "p-LSP-p2mp-1" s2l-dest-address 192.0.2.7 detail
P2MP LSP LSP-p2mp-1: 132 bytes MPLS payload
P2MP Instance p-LSP-p2mp-1, S2L Egress 192.0.2.7
```

RSVP Signaled Point-to-Multipoint LSPs

```
1 192.0.2.3 rtt=2.05 ms rc=8(DSRtrMatchLabel)
   DS 1: ipaddr=192.168.35.2 ifaddr=192.168.35.2 iftype=ipv4Numbered MRU=1564
label=262143 proto=4(RSVP-TE) B/E flags:0/0
2 192.0.2.5 rtt=3.26 ms rc=8(DSRtrMatchLabel)
   DS 1: ipaddr=192.168.57.2 ifaddr=192.168.57.2 iftype=ipv4Numbered MRU=1564
label=262142 proto=4(RSVP-TE) B/E flags:1/0
3 192.0.2.7 rtt=6.12 ms rc=3(EgressRtr)

*A:PE-1#
```

Intelligent Re-merge

Intelligent re-merge protects users from receiving duplicate multicast traffic during convergence. It also protects against duplicate traffic in case of badly designed S2L paths. Initially, three cases exist for which intelligent re-merge is implemented.

Case 1

When the paths of two different S2Ls of the same P2MP LSP instance have Ingress Label Maps (ILMs) on different ports but go out on the same Next-hop Label Forwarding Entry (NHLFE).

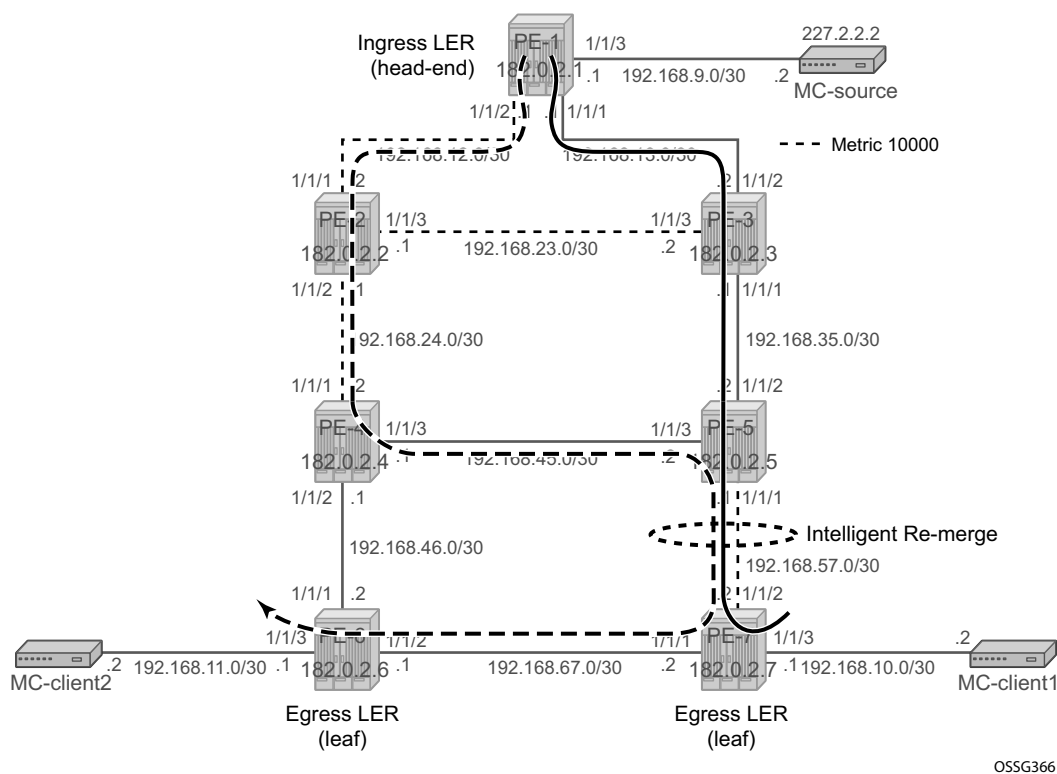


Figure 113: Intelligent Rermerge, Case 1

On the head-end node (PE-1), a new P2MP LSP (*'LSP-p2mp-2'*) will be created with two strict direct MPLS paths (*'strict-to-PE-7'* and *'strict-to-PE-6'*). See [Figure 113](#) for detailed address information. Intelligent re-merge is performed at node PE-5.


```

*A:PE-1# configure router mpls
    path "path-strict-to-PE-7"
        hop 10 192.168.13.2 strict
        hop 20 192.168.35.2 strict
        hop 30 192.168.57.2 strict
        no shutdown
    exit
    path "path-strict-to-PE-6"
        hop 10 192.168.12.2 strict
        hop 20 192.168.24.2 strict
        hop 30 192.168.45.2 strict
        hop 40 192.168.57.2 strict
        hop 50 192.168.67.1 strict
        no shutdown
    exit
    lsp "LSP-p2mp-2" p2mp-lsp
        primary-p2mp-instance "p-LSP-p2mp-2"
            s2l-path "path-strict-to-PE-7" to 192.0.2.7
            exit
            s2l-path "path-strict-to-PE-6" to 192.0.2.6
            exit
        exit
        no shutdown
    exit
    no shutdown

*A:PE-1# show router mpls p2mp-lsp "LSP-p2mp-2" p2mp-instance "p-LSP-p2mp-2"
=====
MPLS P2MP Instance (Originating)
=====
-----
Type : Originating
-----
LSP Name       : LSP-p2mp-2
P2MP ID        : 0
Adm State      : Up
LSP Tunnel ID  : 2
Oper State     : Up

P2MPInstance: p-LSP-p2mp-2
P2MP Inst Id: 2
Inst Admin    : Up
Inst Up Time: 0d 00:34:10
Hop Limit     : 255
Record Route: Record
Include Grps:
None
Bandwidth     : No Reservation
S2l-Name      : path-strict-to-PE-7
S2l Admin     : Up
S2l-Name      : path-strict-to-PE-6
S2l Admin     : Up

P2MP-Inst-type : Primary
P2MP Lsp Id    : 38912
Inst Oper      : Up
Inst Dn Time   : 0d 00:00:00
Adaptive       : Enabled
Record Label   : Record
Exclude Grps   :
None
Oper Bw        : 0 Mbps
To             : 192.0.2.7
S2l Oper       : Up
To             : 192.0.2.6
S2l Oper       : Up
-----
P2MP instances : 1
=====
*A:PE-1#

```

To verify that node PE-5 is not sending duplicate multicast traffic downstream towards PE-7 while it receives two incoming multicast streams, a new tunnel interface and a new static <S,G> IGMP join will be configured on head-end node (PE-1) and leaf nodes (PE-6 and PE-7). Also on the leaf nodes, an extension to the existing multicast information policy is needed. Translated into configuration commands, this becomes:

```
*A:PE-1# configure router pim tunnel-interface rsvp-p2mp "LSP-p2mp-2"

*A:PE-1# configure router igmp
    tunnel-interface rsvp-p2mp "LSP-p2mp-2"
        static
            group 227.2.2.2
            source 192.168.9.2
        exit
    exit
exit

*A:PE-6# configure router pim tunnel-interface rsvp-p2mp "LSP-p2mp-2" sender 192.0.2.1

*A:PE-6# configure router igmp
    interface "int-PE-6-MC-client2"
        static
            group 227.2.2.2
            source 192.168.9.2
        exit
    exit
exit

*A:PE-6# configure mcast-management
    multicast-info-policy "p2mp-pol" create
    bundle "bundle2" create
        primary-tunnel-interface rsvp-p2mp "LSP-p2mp-2" sender 192.0.2.1
        channel 227.2.2.2 create
    exit
exit

*A:PE-7# configure router pim tunnel-interface rsvp-p2mp "LSP-p2mp-2" sender 192.0.2.1

*A:PE-7# configure router igmp
    interface "int-PE-7-MC-client1"
        static
            group 227.2.2.2
            source 192.168.9.2
        exit
    exit
exit

*A:PE-7# configure mcast-management
    multicast-info-policy "p2mp-pol" create
    bundle "bundle2" create
        primary-tunnel-interface rsvp-p2mp "LSP-p2mp-2" sender 192.0.2.1
        channel 227.2.2.2 create
    exit
exit
```

For verification of incoming/outgoing multicast traffic at node PE-5, the **monitor** command is used.

```
*A:PE-5# monitor port 1/1/1 1/1/2 1/1/3 rate interval 3 repeat 100
=====
Monitor statistics for Ports
=====
```

	Input	Output
-----snip-----		
At time t = 18 sec (Mode: Rate)		

Port 1/1/1		

Octets	43	891751
Packets	1	13114
Errors	0	0
Utilization (% of port capacity)	~0.00	0.09
Port 1/1/2		

Octets	891751	50
Packets	13114	0
Errors	0	0
Utilization (% of port capacity)	0.09	~0.00
Port 1/1/3		

Octets	891779	50
Packets	13114	0
Errors	0	0
Utilization (% of port capacity)	0.09	~0.00
---snipped---		

As a conclusion we can say that two incoming multicast streams are seen at PE-5 node (**port 1/1/2** and **port 1/1/3**) and only one outgoing multicast stream (**port 1/1/1**) is sent. No traffic duplication is seen.

Case 2

When two paths of the same S2L have ILMs on different incoming ports and go out on the same NHLFE. This is the case when we perform make-before-break (MBB) on an S2L path due to graceful shutdown or global revertive. This is only a temporary situation since the original path will be torn down.

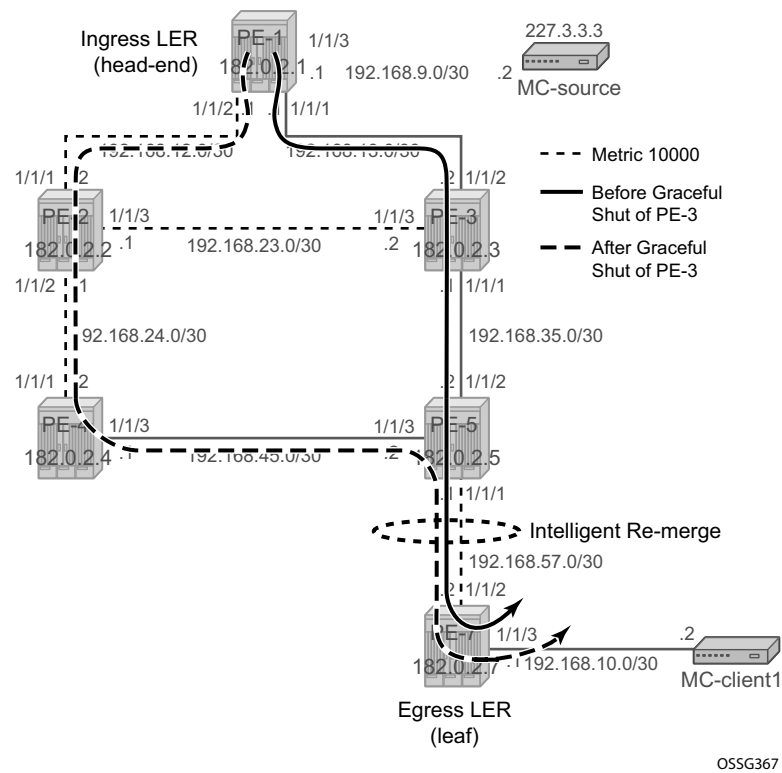


Figure 114: Intelligent Re-merge, Case 2

For this test, only one MC-client will be looked at (the one connected to head-end node PE-7). On PE-4 and PE-7 nodes, port 1/1/4 will be shutdown to isolate PE-6. On the head-end node (PE-1), a new P2MP LSP (LSP-p2mp-3) will be created with one loose MPLS path (loose) and keyword **cspf use-te-metric**. See [Figure 113](#) for detailed address information. Also in this case, intelligent re-merge is performed at node PE-5.

```
*A:PE-1# configure router mpls
      path "loose"
        no shutdown
      exit
      lsp "LSP-p2mp-3" p2mp-lsp
        cspf use-te-metric
        primary-p2mp-instance "p-LSP-p2mp-3"
```

RSVP Signaled Point-to-Multipoint LSPs

```

        s2l-path "loose" to 192.0.2.7
        exit
    exit
    no shutdown
exit
no shutdown

*A:PE-1# show router mpls p2mp-lsp "LSP-p2mp-3" p2mp-instance "p-LSP-p2mp-3" s2l loose to
192.0.2.7 detail
=====
MPLS LSP LSP-p2mp-3 S2L loose (Detail)
=====
Legend :
    @ - Detour Available                # - Detour In Use
    b - Bandwidth Protected            n - Node Protected
    S - Strict                          L - Loose
    A - ABR
    s - Soft Preemption
=====
-----
LSP LSP-p2mp-3 S2L loose
-----
LSP Name       : LSP-p2mp-3                S2l LSP ID   : 47104
P2MP ID        : 0                        S2l Grp Id   : 1
Adm State      : Up                       Oper State   : Up
S2l State:     : Active                    :
S2L Name       : loose                     To           : 192.0.2.7
S2l Admin      : Up                       S2l Oper     : Up
OutInterface: 1/1/1                       Out Label    : 262141
S2L Up Time    : 0d 00:04:23               S2L Dn Time  : 0d 00:00:00
RetryAttempt: 0                           NextRetryIn  : 0 sec
S2L Trans      : 1                        CSPF Queries: 0
Failure Code: noError                     Failure Node: n/a
Inter-area     : False
ExplicitHops:
    No Hops Specified
Actual Hops :
    192.168.13.1 (192.0.2.1)               Record Label : N/A
    -> 192.168.13.2 (192.0.2.3)           Record Label : 262141
    -> 192.168.35.2 (192.0.2.5)           Record Label : 262142
    -> 192.168.57.2 (192.0.2.7)           Record Label : 262142
LastResignal: n/a
=====
*A:PE-1#

```

In a normal situation, the P2MP LSP would follow the nodes PE-1, PE-3, PE-5 and PE-7. This can be verified with MC traffic. Therefore, a new tunnel interface and a new static <S,G> IGMP join will be configured on head-end node (PE-1) and leaf node (PE-7). On the leaf node, an extension to the existing multicast information policy is needed. Translated into configuration commands, this becomes

```
*A:PE-1# configure router pim tunnel-interface rsvp-p2mp "LSP-p2mp-3"

*A:PE-1# configure router igmp
      tunnel-interface rsvp-p2mp "LSP-p2mp-3"
        static
          group 227.3.3.3
          source 192.168.9.2
        exit
      exit
    exit

*A:PE-7# configure router pim tunnel-interface rsvp-p2mp "LSP-p2mp-3" sender 192.0.2.1

*A:PE-7# configure router igmp
      interface "int-PE-7-MC-client1"
        static
          group 227.3.3.3
          source 192.168.9.2
        exit
      exit
    exit

*A:PE-7# configure mcast-management
      multicast-info-policy "p2mp-pol" create
      bundle "bundle3" create
        primary-tunnel-interface rsvp-p2mp LSP-p2mp-3 sender 192.0.2.1
        channel 227.3.3.3 create
      exit
    exit
  exit
```

Monitor the traffic on PE-5. Under normal circumstances, the ingress port is 1/1/2 and the egress port is 1/1/1.

```
*A:PE-5# monitor port 1/1/1 1/1/2 1/1/3 rate interval 3 repeat 999
=====
Monitor statistics for Ports
=====
                                     Input                Output
-----
---snipped---
-----
At time t = 843 sec (Mode: Rate)
-----
Port 1/1/1
```

RSVP Signaled Point-to-Multipoint LSPs

```

-----
Octets                21                1446744
Packets               0                21276
Errors               0                  0
Utilization (% of port capacity)    ~0.00        0.14

Port 1/1/2
-----
Octets                1446836            109
Packets              21276                1
Errors               0                  0
Utilization (% of port capacity)    0.14        ~0.00

Port 1/1/3
-----
Octets                21                111
Packets               0                  1
Errors               0                  0
Utilization (% of port capacity)    ~0.00        ~0.00

```

Now, perform an RSVP graceful shutdown on node PE-3.

```
*A:PE-3# configure router rsvp graceful-shutdown
```

Global revertive is triggered on head-end node PE-1. A new MPLS path will be calculated (see the dotted line in [Figure 113](#)). For a few seconds or even less than a second, the old path and new path are active (two incoming MC streams on node PE-5). Node PE-5 is doing intelligent re-merge, not sending duplicate multicast traffic downstream towards PE-7:

```

*A:PE-5# monitor port 1/1/1 1/1/2 1/1/3 rate interval 3 repeat 100
=====
Monitor statistics for Ports
=====
                                Input                Output
-----
---snip---

-----
At time t = 30 sec (Mode: Rate)
-----
Port 1/1/1
-----
Octets                265                1228986
Packets               2                18070
Errors               0                  0
Utilization (% of port capacity)    ~0.00        0.12

Port 1/1/2
-----
Octets                1229058            91
Packets              18071                1
Errors               0                  0
Utilization (% of port capacity)    0.12        ~0.00

Port 1/1/3

```

Additional Topics

```
-----  
Octets                204539                319  
Packets               3005                  2  
Errors                0                    0  
Utilization (% of port capacity)  0.02          ~0.00  
-----
```

The granularity of the monitoring command is 3 seconds. The graceful shutdown takes less than 3 seconds. However, we can clearly see that the number of outgoing packets on port 1/1/1 equals the number of incoming packets on port 1/1/2. A number of these incoming packets also arrived on port 1/1/3, but no duplicate packets were sent on the outgoing port. No traffic duplication is seen.

Case 3

When a bypass is active on the S2L path and the new global revertive path of the same S2L arrives on the same incoming interface as the original path (interface flapped) at the FRR merge point node. The implementation recognizes this specific case and will signal a different label from the original S2L path coming on that same interface.

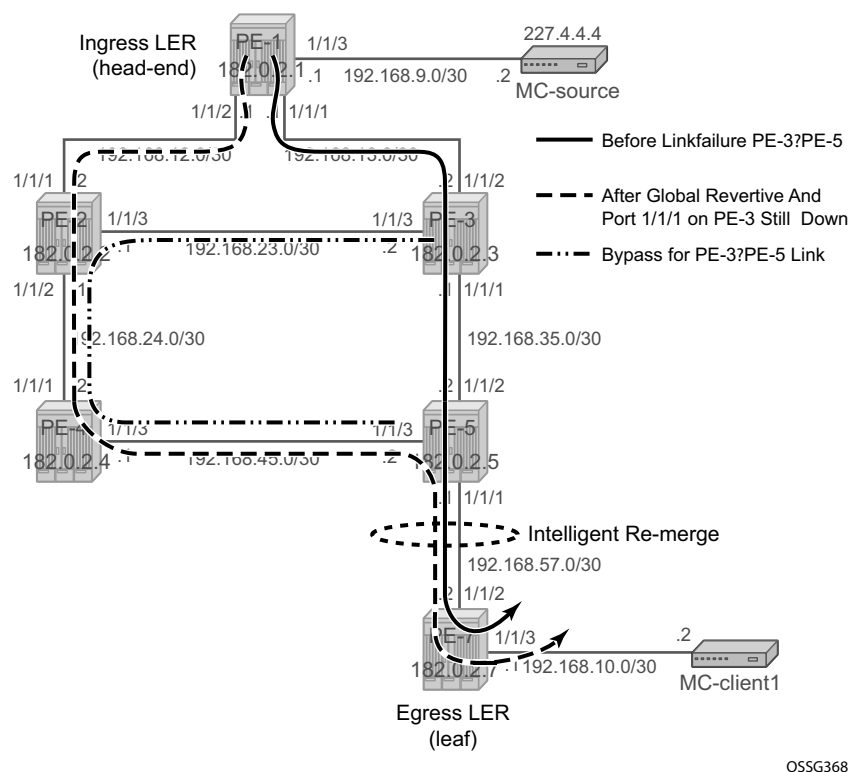


Figure 115: Intelligent Re-merge, Case 3

For this test, all the non-default OSPF metrics are removed from the interfaces. Only one MC-client will be looked at (the one connected to head-end node PE-7). On nodes PE-4 and PE-7, the port towards PE-6 will be shutdown to isolate PE-6. On the head-end node PE-1, a new P2MP LSP (LSP-p2mp-4) will be created with one loose MPLS path (loose) and FRR enabled. See [Figure 115](#) for detailed address information. Also in this case, intelligent re-merge is performed at node PE-5.

```
*A:PE-1# configure router mpls
      path "loose"
      no shutdown
      exit
      lsp "LSP-p2mp-4" p2mp-lsp
      cspf
```

```

fast-reroute facility
no node-protect
exit
primary-p2mp-instance "p-LSP-p2mp-4"
s2l-path "loose" to 192.0.2.7
exit
exit
no shutdown
exit
no shutdown

*A:PE-1# show router mpls p2mp-lsp "LSP-p2mp-4" p2mp-instance "p-LSP-p2mp-4" s2l loose to
192.0.2.7 detail
=====
MPLS LSP LSP-p2mp-4 S2L loose (Detail)
=====
Legend :
@ - Detour Available                # - Detour In Use
b - Bandwidth Protected            n - Node Protected
S - Strict                        L - Loose
A - ABR
s - Soft Preemption
=====
-----
LSP LSP-p2mp-4 S2L loose
-----
LSP Name      : LSP-p2mp-4                S2l LSP ID   : 46592
P2MP ID       : 0                        S2l Grp Id   : 1
Adm State     : Up                      Oper State   : Up
S2l State:    : Active                  :
S2L Name      : loose                   To           : 192.0.2.7
S2l Admin     : Up                     S2l Oper     : Up
OutInterface: 1/1/1                   Out Label    : 262139
S2L Up Time   : 0d 00:02:47           S2L Dn Time  : 0d 00:00:00
RetryAttempt: 0                       NextRetryIn  : 0 sec
S2L Trans     : 1                     CSPF Queries: 1
Failure Code: noError                 Failure Node: n/a
Inter-area    : False
ExplicitHops:
    No Hops Specified
Actual Hops :
    192.168.13.1 (192.0.2.1) @          Record Label : N/A
-> 192.168.13.2 (192.0.2.3) @          Record Label : 262139
-> 192.168.35.2 (192.0.2.5)           Record Label : 262141
-> 192.168.57.2 (192.0.2.7)           Record Label : 262141
ComputedHops:
    192.168.13.1(S)
-> 192.168.13.2(S)
-> 192.168.35.2(S)
-> 192.168.57.2(S)
LastResignal: n/a
=====
*A:PE-1#

```

In the normal situation, the P2MP LSP follows the nodes PE-1, PE-3, PE-5 and PE-7. This can be verified with MC traffic. Therefore, a new tunnel interface and a new static <S,G> IGMP join will be configured on head-end node PE-1 and leaf node PE-7. On the leaf node, an extension to the existing multicast information policy is needed. Translated into configuration commands, this becomes:

```
*A:PE-1# configure router pim tunnel-interface rsvp-p2mp "LSP-p2mp-4"

*A:PE-1# configure router igmp
    tunnel-interface rsvp-p2mp LSP-p2mp-4
    static
        group 227.4.4.4
        source 192.168.9.2
    exit
exit
exit

*A:PE-7# configure router pim tunnel-interface rsvp-p2mp "LSP-p2mp-4" sender 192.0.2.1

*A:PE-7# configure router igmp
    interface "int-PE-7-MC-client1"
    static
        group 227.4.4.4
        source 192.168.9.2
    exit
exit
exit

*A:PE-7# configure mcast-management
    multicast-info-policy "p2mp-pol" create
    bundle "bundle4" create
        primary-tunnel-interface rsvp-p2mp LSP-p2mp-4 sender 192.0.2.1
        channel 227.4.4.4 create
    exit
exit
exit

*A:PE-5# monitor port 1/1/1 1/1/2 1/1/3 rate interval 3 repeat 999
=====
Monitor statistics for Ports
=====
                                     Input                               Output
-----
---snip---
-----
At time t = 1701 sec (Mode: Rate)
-----
Port 1/1/1
-----
Octets                               21                               1304494
Packets                             0                                19183
Errors                              0                                 0
Utilization (% of port capacity)    ~0.00                            0.13
```

```

Port 1/1/2
-----
Octets                    1304549                87
Packets                   19184                  1
Errors                    0                    0
Utilization (% of port capacity)    0.13          ~0.00

Port 1/1/3
-----
Octets                    111                  116
Packets                   1                    1
Errors                    0                    0
Utilization (% of port capacity)    ~0.00          ~0.00

```

Now a link failure on the interface from PE-3 to PE-5 is emulated.

```
*A:PE-3# configure port 1/1/1 shutdown
```

As a consequence of this, traffic will be flowing over the bypass link (see [Figure 115](#) and note the ‘#’ symbol in the next **show** command).

```
*A:PE-1# show router mpls p2mp-lsp "LSP-p2mp-4" p2mp-instance "p-LSP-p2mp-4" s2l loose to 192.0.2.7 detail
```

```
=====
MPLS LSP LSP-p2mp-4 S2L loose (Detail)
=====
```

Legend :

```

@ - Detour Available          # - Detour In Use
b - Bandwidth Protected      n - Node Protected
S - Strict                   L - Loose
A - ABR
s - Soft Preemption

```

```
=====
LSP LSP-p2mp-4 S2L loose
=====
```

```

LSP Name       : LSP-p2mp-4                S2l LSP ID   : 46594
P2MP ID        : 0                        S2l Grp Id   : 1
Adm State      : Up                       Oper State   : Up
S2l State      : Active                   :
S2L Name       : loose                    To          : 192.0.2.7
S2l Admin      : Up                      S2l Oper     : Up
OutInterface   : 1/1/1                   Out Label    : 262140
S2L Up Time    : 0d 00:44:42              S2L Dn Time  : 0d 00:00:00
RetryAttempt    : 0                      NextRetryIn  : 0 sec
S2L Trans       : 3                      CSPF Queries: 3
Failure Code: tunnelLocallyRepaired      Failure Node: 192.0.2.3
Inter-area      : False
ExplicitHops:
  No Hops Specified
Actual Hops :
  192.168.13.1 (192.0.2.1) @
  -> 192.168.13.2 (192.0.2.3) @ #
  -> 192.168.35.2 (192.0.2.5)
  -> 192.168.57.2 (192.0.2.7)
ComputedHops:
  Record Label      : N/A
  Record Label      : 262140
  Record Label      : 262138
  Record Label      : 262140

```

```

    192.168.13.1(S)
-> 192.168.13.2(S)
-> 192.168.35.2(S)
-> 192.168.57.2(S)
LastResignal: n/a
In Prog MBB :
  MBB Type      : GlobalRevert                      NextRetryIn : 15 sec
  Started At    : 04/10/2015 09:14:45              RetryAttempt: 0
  FailureCode   : noError                          Failure Node: n/a
=====
*A:PE-1#

```

In the meantime, PE-3 will trigger a global revertive action (sending PathErr message) towards the head-end node (PE-1).

```
*A:PE-1# show router mpls p2mp-lsp "LSP-p2mp-4" p2mp-instance "p-LSP-p2mp-4" s2l loose to 192.0.2.7 detail
```

```

=====
MPLS LSP LSP-p2mp-4 S2L loose (Detail)
=====
Legend :
  @ - Detour Available           # - Detour In Use
  b - Bandwidth Protected       n - Node Protected
  S - Strict                    L - Loose
  A - ABR
  s - Soft Preemption
=====
-----
LSP LSP-p2mp-4 S2L loose
-----
LSP Name      : LSP-p2mp-4                S2l LSP ID   : 46594
P2MP ID       : 0                        S2l Grp Id   : 2
Adm State     : Up                       Oper State   : Up
S2l State:    : Active                   :
S2L Name      : loose                    To           : 192.0.2.7
S2l Admin     : Up                       S2l Oper     : Up
OutInterface: 1/1/2                      Out Label    : 262142
S2L Up Time   : 0d 00:45:19              S2L Dn Time  : 0d 00:00:00
RetryAttempt: 0                          NextRetryIn  : 0 sec
S2L Trans     : 4                        CSPF Queries: 4
Failure Code  : noError                  Failure Node  : n/a
Inter-area    : False
ExplicitHops:
  No Hops Specified
Actual Hops :
  192.168.12.1 (192.0.2.1) @             Record Label : N/A
-> 192.168.12.2 (192.0.2.2)             Record Label : 262142
-> 192.168.24.2 (192.0.2.4)             Record Label : 262142
-> 192.168.45.2 (192.0.2.5)             Record Label : 262143
-> 192.168.57.2 (192.0.2.7)             Record Label : 262140
ComputedHops:
  192.168.12.1(S)
-> 192.168.12.2(S)
-> 192.168.24.2(S)
-> 192.168.45.2(S)
-> 192.168.57.2(S)

```

Additional Topics

```
LastResignal: n/a
Last MBB      :
  MBB Type    : GlobalRevert                MBB State    : Success
  Ended At    : 04/10/2015 09:15:22
=====
*A:PE-1#
```

For a short time, PE-5 will receive two incoming MC streams (both arriving on port 1/1/3). One from 'bypass' path (PE-3 => PE-2 => PE-4 => PE-5) and one from new MPLS path (PE-1 => PE-2 => PE-4 => PE-5 => PE-7). Port 1/1/1 on PE-5 performs intelligent remerge, only one MC stream is sent downstream towards leaf node PE-7.

Conclusion

From a configuration point of view, a P2MP LSP is only configured on the head-end node of that P2MP LSP, no explicit configuration is needed on the transit LSRs, branch LSRs, bud LSRs and egress LERs/leaf nodes.

Since the PIM protocol is only needed on the head-end node and the leaf node(s), we can work in a PIM-free core network. Although convergence is not covered in this configuration note, failures in the core will be resolved by MPLS (in case of FRR, traffic loss for less than 50ms is expected). This is a major improvement compared to PIM convergence.

Segment Routing with IS-IS Control Plane

In This Chapter

This section provides information about segment routing with IS-IS control plane.

Topics in this section include:

- [Applicability on page 702](#)
- [Overview on page 703](#)
- [Configuration on page 705](#)
- [Conclusion on page 724](#)

Applicability

Segment Routing is supported on 7950 XRS-16c/20/40, 7750 SR-a4/8 and in all chassis modes of 7750 SR-7/12 and 7450 ESS-7/12 and was tested on SR OS 13.0.R3.

Overview

Segment Routing (SR) is a technology for IP/Multiprotocol Label Switching (MPLS) networks that enables source routing. With source routing, operators can specify a forwarding path, from ingress to egress, that is independent of the shortest path determined by the Interior Gateway Protocol (IGP).

The main benefit of SR compared to other source routing protocols (such as Resource Reservation Protocol with Traffic Engineering (RSVP-TE)) is that, from a control plane perspective, no signaling protocol is required. SR provides a path or tunnel, encoded as a sequential list of sub-paths or segments that are advertised within the SR domain, using extensions to well-known link-state routing protocols, such as Intermediate System to Intermediate System (IS-IS) or Open Shortest Path First (OSPF).

Implementation

An SR tunnel can contain a single segment that represents the destination node, or it can contain a list of segments that the tunnel must traverse. The tunnel can be established over an IPv4/IPv6 MPLS or IPv6 data plane, encoded as a stack of MPLS labels or as a number of IPv6 addresses contained in an IPv6 extension header.

Network elements are modeled as segments. For each segment, IGP advertises an identifier referred to as a Segment ID (SID).

The two segment types are:

1. **Prefix Segment** — Globally unique and allocated from a Segment Routing Global Block (SRGB), typically multi-hop and signaled by the IGP. It is the Equal Cost Multi-Path ECMP-aware shortest path IGP route to a related prefix. A typical example of a Prefix Segment is a Node SID. Within the SR OS implementation, the Node SID is either the system address or another interface address in the Global Routing Table (GRT) of type loopback. Node SIDs are advertised in IS-IS using a Prefix SID sub-TLV (Type Length Value).
2. **Adjacency Segment** — Locally unique and allocated from the (local) dynamic label space, so that other routers in the SR domain can use the same label space. Adjacency Segments are signaled by the IGP. Within the SR OS implementation, Adjacency SIDs are automatically assigned and advertised when the SR context within the IGP instance is set in no shutdown. Adjacency SIDs are advertised in IS-IS using an Adjacency SID sub-TLV.

To make Prefix Segments globally unique within the SR domain, an indexing mechanism is required, because production networks consist of multiple vendors and multiple products. As a result, it is often difficult to agree on a common SRGB for the Prefix SIDs.

Implementation

All routers within the SR domain are expected to configure and advertise the same Prefix SID index range for an IGP instance. The label value used by each router to represent a prefix can be local to that router by the use of an offset label, referred to as a start label:

Local label (for a prefix) = (local) start label + {Prefix SID index}

Within the SR OS implementation, Prefix Loop-Free Alternate (LFA) is supported for SR to improve the Fast Reroute (FRR) coverage. Remote LFA (RLFA) is also supported. With RLFA, SR shortest path tunnels are used as a virtual LFA or repair tunnel toward the PQ node.

The following example uses IS-IS as an IGP protocol, with an MPLS data plane and services enabled using LFA and RLFA.

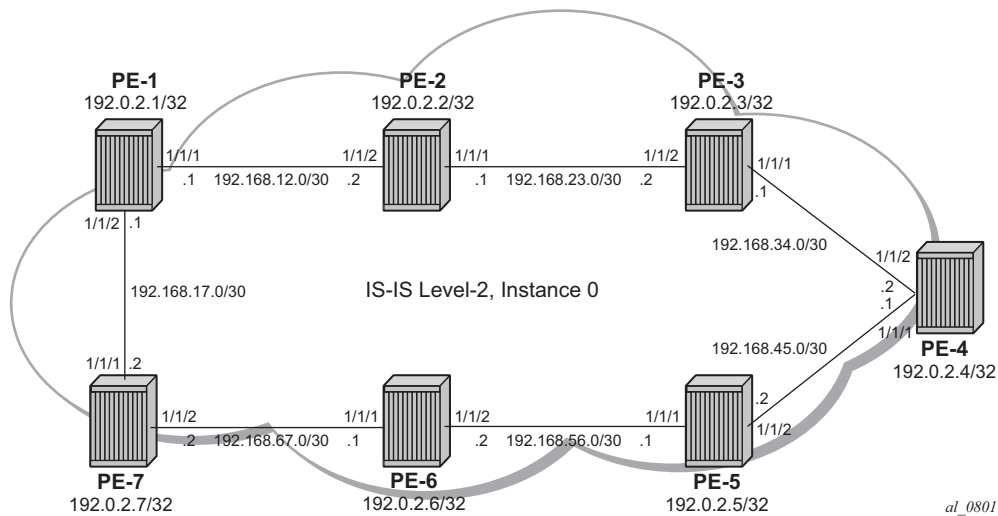


Figure 116: Network Topology

Configuration

Step 1. Configure an IP/MPLS network according to [Figure 116](#).

- The system and IP interface addresses are configured according to Figure 1.
- IS-IS level 2 is selected as the IGP to distribute routing information between all PEs. All IS-IS interfaces are of type point-to-point to avoid running the DR/BDR (Designated Router/Backup Designated Router) election process.

Step 2. Configure SR.

Before enabling SR on a router, define a dedicated SRGB. This SRGB is required on each individual router part of the SR domain and is used to allocate the Prefix SIDs.

By default, an SRGB is not instantiated and, when configured by the operator, it is taken from the system dynamic label range. Translated into a show command:

```
*A:PE-1# show router mpls-labels label-range
=====
Label Ranges
=====
```

Label Type	Start Label	End Label	Aging	Available	Total
Static	32	18431	-	18400	18400
Dynamic	18432	131071	0	112640	112640
Seg-Route	0	0	-	0	112640

For simplicity, the same SRGB is used in this example for all SR domain routers. Within the command, a start value and end value defines the size of the SRGB. Translated into configuration commands for an SRGB of 100 MPLS labels, this becomes:

```
A:PE-1# configure router mpls-labels sr-labels start 20000 end 20099
A:PE-1# show router mpls-labels label-range
=====
Label Ranges
=====
```

Label Type	Start Label	End Label	Aging	Available	Total
Static	32	18431	-	18400	18400
Dynamic	18432	524287	0	505756	505856
Seg-Route	20000	20099	-	0	100

This command is repeated for all other nodes. The allocated MPLS labels are only for the Prefix SIDs. The Adjacency SIDs, which are only locally unique, are taken from the dynamic range; in this example, between 18432 and 524287.

Step 2.1: Enable router capability in the IGP instance.

It is mandatory to enable the router-capability parameter inside the IS-IS instance, to advertise SR support among the IS-IS adjacencies. By configuring this command within the IGP instance, the

SR Capability sub-TLV is propagated and is used to indicate the index range and the start label. The SR Algorithm sub-TLV is also used to advertise the algorithm used for path calculations. Only Shortest Path First (SPF) (value 0) is defined. Translated into a configuration command, this becomes:

```
<all-nodes-within-SR-domain># configure router isis advertise-router-capability area
```

The flooding parameter is a mandatory parameter in this CLI command. The keyword area or as indicates that the router capabilities Label Switched Path (LSP) should be advertised throughout the same level or throughout the whole Autonomous System (AS). In the preceding example, all routers belong to the same level, so the area argument is sufficient. When the SR context within the IGP instance is set in no shutdown, both IS-IS sub-TLVs are flooded.

Step 2.2: Define the Prefix SID index range.

The SR OS implementation for SR provides two mutually exclusive modes of operation to define the Prefix SID index range: global mode and per-instance mode. Per-instance mode is useful in a seamless MPLS environment when multiple IGP instances are used. The main difference between the modes is the way that the start label and index range are calculated.

A comparison of the modes is shown in following table:

Table 3: Mode Comparison

Global	Per Instance
Applicable for all IGP instances on that node	Applicable for one dedicated IGP instance
Start label is first label of SRGB	Start label is configurable (but part of SRGB range); use of non-overlapping sub-ranges of SRGB
Prefix SID index range is “size” of SRGB	Prefix SID index-range is configurable
If SRGB needs to change, shutdown SR and delete prefix-SID-ranges in all IGP instances	If prefix SID index and/or label range needs to change, shutdown SR in that specific IGP instance
SW checks whether any allocated SID index/label goes out of range.	
SW checks also for overlaps of the resulting net label value range across IGP instances.	

For simplicity, global mode is used for this example. Translated into a CLI command, this becomes:

```
<all-nodes-within-SR-domain># configure router isis segment-routing prefix-sid-range
global
```

Step 2.3: Assign a Prefix SID index or label to the prefix representing a node.

To be able to set up SR shortest path tunnels to all routers of the SR domain, each router needs to be uniquely defined within the SR domain. Therefore, the system address or other loopback interface in the GRT will be assigned an ipv4-node-sid index or label value that is unique within the SR domain.

Translated into configuration commands, this becomes:

```
*A:PE-1# configure router isis interface "system" ipv4-node-sid index 1
*A:PE-2# configure router isis interface "system" ipv4-node-sid index 2
*A:PE-3# configure router isis interface "system" ipv4-node-sid index 3
*A:PE-4# configure router isis interface "system" ipv4-node-sid index 4
*A:PE-5# configure router isis interface "system" ipv4-node-sid index 5
*A:PE-6# configure router isis interface "system" ipv4-node-sid index 6
*A:PE-7# configure router isis interface "system" ipv4-node-sid index 7
```

Because the SRGB is the same on all nodes, each node in the network can be reached using the same MPLS label. For example, the Node SID for PE-5 on all nodes has a start label (first label of the SRGB (= 20000) + ipv4-node-sid index on PE-5 node (= 5)) of 20005.

When there is one consistent SRGB for the SR domain, the SR OS CLI allows the use of absolute MPLS label values instead of index values. For example, on PE-1, an operator can use an explicit MPLS label value:

```
*A:PE-1# configure router isis interface "system" ipv4-node-sid label 20001
```

Internally, this explicit value is translated into an index value (index-value 1) before advertising it toward its neighbors, taking into account the Prefix SID index-range mode (global or per-instance) and the SRGB.

Step 2.4: Enable SR context within the IGP instance. Translated into a configuration command, this becomes:

```
<all-nodes-within-SR-domain># configure router isis segment-routing no shutdown
```

After enabling the SR context within an IGP instance, the SR Capability sub-TLV, and the SR Algorithm sub-TLV between all routers within the SR domain, are flooded (see step 2.1 for the configuration command). A show command is available to display the SR related router capability information. For example, on PE-1, this becomes:

Configuration

```
A:PE-1# show router isis capabilities level 2
=====
Router Base ISIS Instance 0 Capabilities
=====
Displaying Level 2 capabilities
-----
LSP ID      : PE-1.00-00
  Router Cap : 192.0.2.1, D:0, S:0
    TE Node Cap : B E M P
    SR Cap: IPv4 , SRGB Base:20000, Range:100
    SR Alg: metric based SPF

LSP ID      : PE-2.00-00
  Router Cap : 192.0.2.2, D:0, S:0
    TE Node Cap : B E M P
    SR Cap: IPv4 , SRGB Base:20000, Range:100
    SR Alg: metric based SPF

LSP ID      : PE-3.00-00
  Router Cap : 192.0.2.3, D:0, S:0
    TE Node Cap : B E M P
    SR Cap: IPv4 , SRGB Base:20000, Range:100
    SR Alg: metric based SPF

LSP ID      : PE-4.00-00
  Router Cap : 192.0.2.4, D:0, S:0
    TE Node Cap : B E M P
    SR Cap: IPv4 , SRGB Base:20000, Range:100
    SR Alg: metric based SPF

LSP ID      : PE-5.00-00
  Router Cap : 192.0.2.5, D:0, S:0
    TE Node Cap : B E M P
    SR Cap: IPv4 , SRGB Base:20000, Range:100
    SR Alg: metric based SPF

LSP ID      : PE-6.00-00
  Router Cap : 192.0.2.6, D:0, S:0
    TE Node Cap : B E M P
    SR Cap: IPv4 , SRGB Base:20000, Range:100
    SR Alg: metric based SPF

LSP ID      : PE-7.00-00
  Router Cap : 192.0.2.7, D:0, S:0
    TE Node Cap : B E M P
    SR Cap: IPv4 , SRGB Base:20000, Range:100
    SR Alg: metric based SPF

Level (2) Capability Count : 7
```

A similar output occurs for each router in the SR domain.

After enabling the SR context within the IGP instance, the assigned index for each locally configured Prefix SID is advertised. After the advertisement of Prefix SIDs, MPLS data plane Ingress Label Mapping (ILM) is programmed with a pop operation. In this context, a show

command can be used to display the Prefix SIDs, in order, within the SR domain. As an example, on PE-1, this becomes:

A:PE-1# show router isis prefix-sids

```
=====
Router Base ISIS Instance 0 Prefix/SID Table
=====
Prefix                               SID      Lvl/Typ  SRMS  AdvRtr
                               MT      Flags
-----
192.0.2.1/32                        1        2/Int.   N      PE1
                               0      NnP
192.0.2.2/32                        2        2/Int.   N      PE2
                               0      NnP
192.0.2.3/32                        3        2/Int.   N      PE3
                               0      NnP
192.0.2.4/32                        4        2/Int.   N      PE4
                               0      NnP
192.0.2.5/32                        5        2/Int.   N      PE5
                               0      NnP
192.0.2.6/32                        6        2/Int.   N      PE6
                               0      NnP
192.0.2.7/32                        7        2/Int.   N      PE7
                               0      NnP
-----
No. of Prefix/SIDs: 7
Flags: R = Re-advertisement
       N = Node-SID
       nP = no penultimate hop POP
       E = Explicit-Null
       V = Prefix-SID carries a value
       L = value/index has local significance
```

The SR OS implementation, by default, sets the Node SID (or N-flag) and noPenultimate hop PoP (or nP-flag) inside the Prefix SID TLV. Another useful flag that can be set is the re-advertisement (or R-flag). The R-flag is set when a Prefix SID is propagated between levels or areas, or redistribution is in place (from another protocol).

Prefix SID information can also be viewed within the IGP database attached to (extended) IP Prefix reachability TLVs. For example, on PE-1, this becomes:

```
*A:PE-1# show router isis database level 2 PE-1.00-00 detail
=====
Router Base ISIS Instance 0 Database
=====

Displaying Level 2 database
-----
LSP ID   : PE-1.00-00                      Level    : L2
Sequence : 0x1                             Checksum : 0xe8ae  Lifetime : 739
Version  : 1                             Pkt Type  : 20    Pkt Ver  : 1
Attributes: L1L2                         Max Area  : 3
SysID Len : 6                           Used Len  : 248   Alloc Len : 1492
```

Configuration

```
TLVs :
  Supp Protocols:
    Protocols      : IPv4
  IS-Hostname     : PE-1
  Router ID      :
    Router ID     : 192.0.2.1
  ---snip---
  TE IP Reach    :
    ---snip---
  Default Metric : 0
  Control Info:   S, prefLen 32
  Prefix        : 192.0.2.1
  Sub TLV       :
    Prefix-SID Index:1, Algo:0, Flags:NnP
```

```
Level (2) LSP Count : 1
```

After enabling the SR context within the IGP instance, Adjacency SIDs are also automatically assigned and advertised for each formed adjacency over an IP interface. From a data plane perspective, one local Adjacency SID consumes one ILM entry, programming a pop operation.

Similar to Prefix SIDs, Adjacency SID information can be viewed within the IGP database attached to IS Neighbor TLVs. Translated into the previous show command on PE-1, this becomes:

```
A:PE-1# show router isis database level 2 PE-1.00-00 detail
=====
Router Base ISIS Instance 0 Database
=====
Displaying Level 2 database
-----
LSP ID      : PE-1.00-00                      Level      : L2
Sequence    : 0x1d                            Checksum   : 0xb0ca  Lifetime   : 911
Version     : 1                               Pkt Type   : 20    Pkt Ver    : 1
Attributes: L1L2                             Max Area   : 3
SysID Len   : 6                               Used Len   : 248   Alloc Len  : 1492

TLVs :
  Supp Protocols:
    Protocols      : IPv4
  IS-Hostname     : PE-1
  Router ID      :
    Router ID     : 192.0.2.1
  ---snip---
  TE IS Nbrs     :
    Nbr           : PE-2.00
    Default Metric : 10
    Sub TLV Len    : 19
    IF Addr       : 192.168.12.1
    Nbr IP        : 192.168.12.2
    Adj-SID: Flags:v4VL Weight:0 Label:262141
  TE IS Nbrs     :
    Nbr           : PE-7.00
    Default Metric : 10
    Sub TLV Len    : 19
    IF Addr       : 192.168.17.1
```

```
Nbr IP      : 192.168.17.2
Adj-SID: Flags:v4VL Weight:0 Label:262140
---snip---
```

The SR OS implementation, by default, sets the Value (or V-flag), meaning that the Adjacency SID carries a value (as opposed to an index). Also, the Local (or L-flag) is set by default, meaning that the Adjacency SID has only local significance. The v4-flag set to 0 means that the Adjacency SID references to an adjacency with outgoing IPv4 encapsulation.

Another way to display Adjacency SID information is using the show router isis adjacency detail command.

```
A:PE-1# show router isis adjacency "int-PE-1-PE-2" detail
=====
Router Base ISIS Instance 0 Adjacency
=====
SystemID      : PE-2                      SNPA          : 4a:c5:01:01:00:02
Interface     : int-PE-1-PE-2             Up Time       : 1d 01:26:23
State         : Up                       Priority       : 0
Nbr Sys Typ   : L2                      L. Circ Typ   : L2
Hold Time     : 19                      Max Hold      : 27
Adj Level     : L2                      MT Enabled    : No
Topology      : Unicast

IPv6 Neighbor  : ::
IPv4 Neighbor  : 192.168.12.2
IPv4 Adj SID   : Label 262141
---snip---
```

```
=====
A:PE-1# show router isis adjacency "int-PE-1-PE-7" detail
=====
Router Base ISIS Instance 0 Adjacency
=====
SystemID      : PE-7                      SNPA          : 4a:a4:01:01:00:01
Interface     : int-PE-1-PE-7             Up Time       : 1d 01:26:13
State         : Up                       Priority       : 0
Nbr Sys Typ   : L2                      L. Circ Typ   : L2
Hold Time     : 26                      Max Hold      : 27
Adj Level     : L2                      MT Enabled    : No
Topology      : Unicast

IPv6 Neighbor  : ::
IPv4 Neighbor  : 192.168.17.2
IPv4 Adj SID   : Label 262140
---snip---
```

Finally, when enabling the SR context within the IGP instance, the SR module resolves received prefixes with Prefix SID sub-TLVs present. As a result, MPLS data plane resources are consumed. The ILM is programmed with a swap operation and the Label-to-next-hop-Label-Forwarding-Entry (LTN) with a push operation, both pointing to the primary and/or LFA Next-Hop Label Forwarding Entry (NHLFE). Also, an SR tunnel is added in the Tunnel Table Manager (TTM). As a result, an SR shortest path tunnel is set up to each other router that is part of the SR domain. Now, SR shortest path tunnels can be used for all users of TTM.

Examples

Example 1: VPRN service with LFA and RLFA enabled

In the network topology of [Figure 116](#), no LDP and RSVP-TE signaling protocols are enabled. Each router of the SR domain has a full mesh of SR shortest path tunnels to the other routers, and no LDP and RSVP-TE LSPs are present. For example, on PE-1, the TTM looks like:

```
A:PE-1# show router tunnel-table
=====
IPv4 Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId Pref  Nexthop      Metric
-----
192.0.2.2/32      isis (0)   MPLS  524443    11    192.168.12.2  10
192.0.2.3/32      isis (0)   MPLS  524448    11    192.168.12.2  20
192.0.2.4/32      isis (0)   MPLS  524449    11    192.168.12.2  30
192.0.2.5/32      isis (0)   MPLS  524452    11    192.168.17.2  30
192.0.2.6/32      isis (0)   MPLS  524450    11    192.168.17.2  20
192.0.2.7/32      isis (0)   MPLS  524451    11    192.168.17.2  10
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
```

The objective is to configure a VPRN between PE-1 and PE-7, using SR shortest path tunnels as transport tunnel. The configuration will look like this:

```
*A:PE-1# configure service vprn 100 customer 1 create
      autonomous-system 64496
      route-distinguisher 64496:10001
      auto-bind-tunnel
        resolution any
      exit
      vrf-target target:64496:100
      interface "loopback" create
        address 192.0.1.1/32
        loopback
      exit
      no shutdown

*A:PE-7# configure service vprn 100 customer 1 create
      autonomous-system 64496
      route-distinguisher 64496:10007
      auto-bind-tunnel
        resolution any
```

```

exit
vrf-target target:64496:100
interface "loopback" create
    address 192.0.1.7/32
    loopback
exit
no shutdown

```

Within the VPRN service configuration, a loopback interface is created on both PEs to verify the transport mechanism.

Tunnel information displaying the MPLS label value is retrieved using the show router fp-tunnel-table <slotnumber> command. For example, on PE-1:

```

*A:PE-1# show router fp-tunnel-table 1 192.0.2.7/32
=====
Tunnel Table Display
Legend:
B - FRR Backup
=====
Destination                                Protocol  Tunnel-ID
      Lbl                                NextHop    Intf/Tunnel
-----
192.0.2.7/32                               SR-ISIS-0  -
      20007                               192.168.17.2  1/1/2

```

This means that, when traffic arrives on PE-1, the MPLS label 20007 is pushed to reach destination PE-7. Because, in this example, the Prefix SID index range global mode is used, the value 20007 comes from the start label on PE-7 (first label of the SRGB, which is 20000, plus the configured index value of Node SID PE-7 (7)), so 20007.

Enabling Prefix LFA within the IS-IS context on PE-1 will enable LFA/FRR protection. Next-hop LFA protection is present for node PE-4, node PE-5, and the link between PE-4 and PE-5.

Translated into CLI and show commands, this becomes:

```

*A:PE-1# configure router isis loopfree-alternate
*A:PE-1# show router isis lfa-coverage
=====
Router Base ISIS Instance 0 LFA Coverage
=====
Topology      Level  Node          IPv4          IPv6
-----
---snip---
IPv4 Unicast  L2      2/6 (33%)    3/11 (27%)    0/0 (0%)
---snip---

*A:PE-1# show router route-table alternative
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type  Proto  Age      Metric  Pref
      Next Hop[Interface Name]
      Alt-NextHop                                Alt-
                                                Metric
-----

```

Configuration

```

---snip---
192.0.2.4/32                               Remote  ISIS      00h27m26s  18
      192.168.12.2                          30
      192.168.17.2 (LFA)                     40
192.0.2.5/32                               Remote  ISIS      00h23m34s  18
      192.168.17.2                          30
      192.168.12.2 (LFA)                     40
---snip---
192.168.45.0/30                           Remote  ISIS      00h27m27s  18
      192.168.12.2                          40
      192.168.17.2 (LFA)                     50
---snip---

```

No. of Routes: <...>

Flags: n = Number of times nexthop is repeated

Backup = BGP backup route

LFA = Loop-Free Alternate nexthop

S = Sticky ECMP requested

*A:PE-1# show router fp-tunnel-table 1

Tunnel Table Display

Legend:

B - FRR Backup

```

=====
Destination                                Protocol  Tunnel-ID
      Lbl                                NextHop  Intf/Tunnel
-----
---snip---
192.0.2.4/32                               SR-ISIS-0  -
      20004                            192.168.12.2  1/1/1
      20004                            192.168.17.2 (B)  1/1/2
192.0.2.5/32                               SR-ISIS-0  -
      20005                            192.168.17.2  1/1/2
      20005                            192.168.12.2 (B)  1/1/1

```

*A:PE-1# show router tunnel-table detail

Tunnel Table (Router: Base)

---snip---

```

-----
Destination      : 192.0.2.4/32
NextHop          : 192.168.12.2
Tunnel Flags     : has-lfa exclude-for-igpshortcuts
Age              : 00h08m22s
Owner            : isis (0)
Tunnel ID        : 524449
Tunnel Label     : 20004
Tunnel MTU       : 1500
Encap            : MPLS
Preference       : 11
Tunnel Metric    : 30

```

```

-----
Destination      : 192.0.2.5/32
NextHop          : 192.168.17.2
Tunnel Flags     : has-lfa exclude-for-igpshortcuts
Age              : 00h08m24s
Owner            : isis (0)
Encap            : MPLS

```

```
Tunnel ID       : 524452           Preference      : 11
Tunnel Label    : 20005            Tunnel Metric    : 30
Tunnel MTU      : 1500
```

---snip---

Number of tunnel-table entries with LFA : 2

When a failure occurs on the primary SR path (only applicable for prefix PE-4/PE-5 and the link between PE-4 and PE-5), the traffic takes the LFA backup SR path to the destination using the same MPLS label value.

To extend the LFA/FRR coverage, for example, to find an LFA protection for node PE-7, which is one of the VPRN service endpoints, RLFA can be enabled. RLFA creates a virtual LFA by using a repair tunnel to carry packets to a point in the network from where they will not be looped back to the source, but forwarded (SPF-based) toward the destination prefix.

The RLFA implementation uses the PQ algorithm. The node where RLFA is configured (PE-1 in this example) computes an extended P-space and a Q-space. The intersection of both spaces is called the PQ-node. This PQ node is the destination node of the repair tunnel using an SR shortest path tunnel. To compute both spaces, SPF is used.

In this example, IS-IS is used as the IGP, using a default metric value of 10 for all links. With the assumption that the link between PE-1 and PE-7 is broken, the calculation of both the extended P-space and the Q-space at PE-1 is as follows:

- extended P-space — An SPF computed from node PE-1 and rooted at PE-2. It is used to calculate the set of routers that are reachable without any path transiting the protected link between PE-1 and PE-7. The following nodes belong to the extended P-space: PE-2, PE-3, PE-4, and PE-5.
- Q-space — A reverse SPF computed from PE-1 and rooted from PE-7 (acting as destination proxy). It is used to calculate the set of routers that can reach PE-7 without transiting the protected link between PE-1 and PE-7. The nodes PE-4, PE-5, and PE-6 belong to the Q-space.

Possible PQ-nodes are PE-4 or PE-5, because they are in the intersection of both spaces. PE-4 is taken as PQ node because PE-4 has the lowest IGP cost from the PE-1 point of view. As a result, PE-1 uses the SR shortest path tunnel toward PE-4, when the direct link between PE-1 and PE-7 is broken.

Translated into CLI and show commands, this becomes:

```
*A:PE-1# configure router isis loopfree-alternate remote-lfa
*A:PE-1# show router fp-tunnel-table 1
=====
Tunnel Table Display

Legend:
B - FRR Backup
=====
```

Configuration

Destination Lbl	NextHop	Intf/Tunnel	Protocol	Tunnel-ID
192.0.2.2/32			SR-ISIS-0	-
20002	192.168.12.2	1/1/1		
20002/20004	192.168.17.2 (B)	1/1/2		
192.0.2.3/32			SR-ISIS-0	-
20003	192.168.12.2	1/1/1		
20003/20004	192.168.17.2 (B)	1/1/2		
192.0.2.4/32			SR-ISIS-0	-
20004	192.168.12.2	1/1/1		
20004	192.168.17.2 (B)	1/1/2		
192.0.2.5/32			SR-ISIS-0	-
20005	192.168.17.2	1/1/2		
20005	192.168.12.2 (B)	1/1/1		
192.0.2.6/32			SR-ISIS-0	-
20006	192.168.17.2	1/1/2		
20006/20004	192.168.12.2 (B)	1/1/1		
192.0.2.7/32			SR-ISIS-0	-
20007	192.168.17.2	1/1/2		
20007/20004	192.168.12.2 (B)	1/1/1		

In the preceding example, the nodes PE-2, PE-3, PE-6, and PE-7 now have RLFA protection, as well as the LFA protection provided for PE-4 and PE-5.

The main difference between normal prefix LFA and RLFA is that for RLFA a two-MPLS label stack is pushed by the head-end node (PE-1). The top-label is the SR-label to reach the PQ node (PE-4, 20004) and the bottom-label is the SR-label to reach the destination node (for example, for PE-7, this is 20007). The notation inside the show-command is bottom-label/top-label.

Figure 117 illustrates the RLFA traffic path during protection:

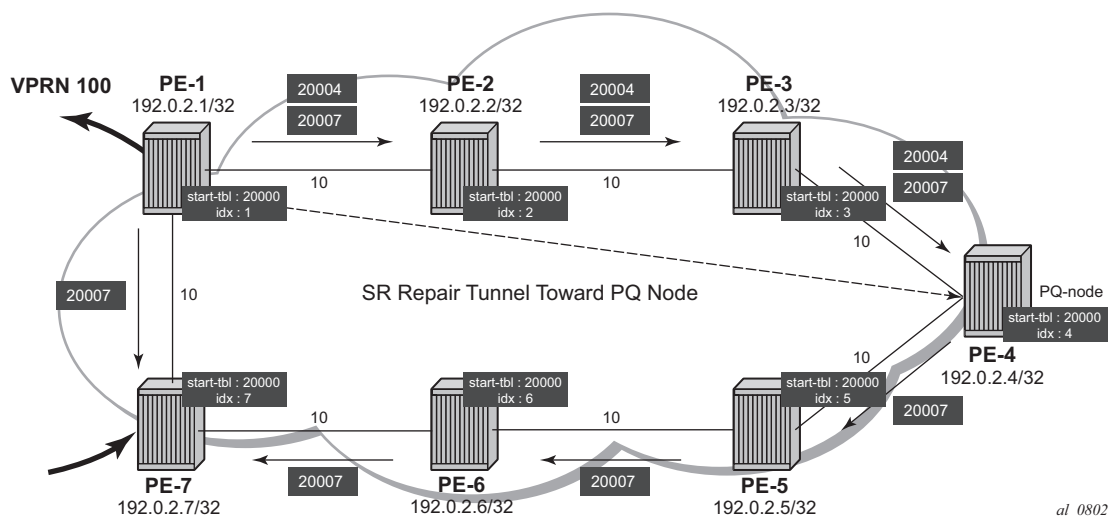


Figure 117: RLFA Traffic Path During Protection

Inside the TTM, a tunnel-flag, has-lfa exclude-for-igpshortcuts, is set for all destination nodes that have LFA protection available:

```
*A:PE-1# show router tunnel-table detail
=====
Tunnel Table (Router: Base)
=====
Destination      : 192.0.2.2/32
NextHop          : 192.168.12.2
Tunnel Flags     : has-lfa exclude-for-igpshortcuts
Age              : 00h11m03s
Owner            : isis (0)           Encap              : MPLS
Tunnel ID        : 524443             Preference         : 11
Tunnel Label     : 20002              Tunnel Metric      : 10
Tunnel MTU       : 1500
-----
Destination      : 192.0.2.3/32
NextHop          : 192.168.12.2
Tunnel Flags     : has-lfa exclude-for-igpshortcuts
Age              : 00h11m03s
Owner            : isis (0)           Encap              : MPLS
Tunnel ID        : 524448             Preference         : 11
Tunnel Label     : 20003              Tunnel Metric      : 20
Tunnel MTU       : 1500
-----
---snip---
-----
Destination      : 192.0.2.6/32
NextHop          : 192.168.17.2
Tunnel Flags     : has-lfa exclude-for-igpshortcuts
Age              : 00h11m03s
Owner            : isis (0)           Encap              : MPLS
Tunnel ID        : 524450             Preference         : 11
Tunnel Label     : 20006              Tunnel Metric      : 20
Tunnel MTU       : 1500
-----
Destination      : 192.0.2.7/32
NextHop          : 192.168.17.2
Tunnel Flags     : has-lfa exclude-for-igpshortcuts
Age              : 00h11m03s
Owner            : isis (0)           Encap              : MPLS
Tunnel ID        : 524451             Preference         : 11
Tunnel Label     : 20007              Tunnel Metric      : 10
Tunnel MTU       : 1500
-----
---snip---
Number of tunnel-table entries with LFA : 6
```

Verification of the loopback address configured within the VPRN service context on PE-7 (using loopback address 192.0.1.7/32) shows that an SR shortest path tunnel is used as the transport mechanism:

```
*A:PE-1# show router 100 route-table 192.0.1.7/32 extensive
=====
Route Table (Service: 100)
=====
```

Configuration

```
Dest Prefix      : 192.0.1.7/32
Protocol         : BGP_VPN
Age              : 00h21m36s
Preference       : 170
Indirect Next-Hop : 192.0.2.7
  QoS            : Priority=n/c, FC=n/c
  Source-Class   : 0
  Dest-Class     : 0
  ECMP-Weight    : N/A
Resolving Next-Hop : 192.0.2.7 (ISIS tunnel)
  Metric         : 10
  ECMP-Weight    : N/A
```

No. of Destinations: 1

Example 2: TTM preference with VPRN service

The following example is a variant on the previous example. The difference in this example is that, in addition to SR, LDP and RSVP-TE are also enabled between PE-1 and PE-7. A single RSVP LSP is configured originating at PE-1 and terminating at PE-7.

The objective of this example is to show the difference in protocol preference within TTM and how to influence the default behavior. This can be useful in case of migration scenarios from a non-SR environment toward a hybrid environment having LDP/RSVP and SR enabled.

In the following example, LFA/RLFA is no longer configured on the PE-1 node.

Translated into configuration commands, this becomes:

```
*A:PE-1# configure router isis no loopfree-alternate
*A:PE-1# configure router ldp interface-parameters interface "int-PE-1-PE-7"
*A:PE-1# configure router mpls interface "int-PE-1-PE-7"
*A:PE-1# configure router mpls no shutdown
*A:PE-1# configure router rsvp no shutdown
*A:PE-1# configure router mpls
    path "dyn"
    no shutdown
  exit
  lsp "LSP-PE-1-PE-7"
    to 192.0.2.7
    primary "dyn"
  exit
  no shutdown
exit
no shutdown

*A:PE-7# configure router ldp interface-parameters interface "int-PE-7-PE-1"
*A:PE-7# configure router mpls no shutdown
*A:PE-7# configure router rsvp no shutdown
```

By enabling LDP and RSVP between PE-1 and PE-7, the TTM on both nodes changed. With the VPRN service between PE-1 and PE-7 of example 1, only those two specific service endpoints are displayed:

```
*A:PE-1# show router tunnel-table 192.0.2.7
=====
IPv4 Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId  Pref    Nexthop      Metric
-----
192.0.2.7/32     rsvp      MPLS    2         7      192.168.17.2  10
192.0.2.7/32     ldp       MPLS    65537     9      192.168.17.2  10
192.0.2.7/32     isis (0)  MPLS    524451    11     192.168.17.2  10

*A:PE-7# show router tunnel-table 192.0.2.1
=====
IPv4 Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId  Pref    Nexthop      Metric
-----
192.0.2.1/32     ldp       MPLS    65537     9      192.168.17.1  10
192.0.2.1/32     isis (0)  MPLS    524396    11     192.168.17.1  10
```

On node PE-1, an RSVP LSP, an LDP LSP, and an SR shortest path tunnel (using IS-IS) are present. Because the VPRN service has auto-bind-tunnel resolution any enabled, the protocol type with the highest TTM preference (meaning the lowest absolute preference value in TTM) is taken; in this case, the RSVP LSP. This can be verified for the configured loopback address within the VPRN service context, as follows:

```
*A:PE-1# show router 100 route-table 192.0.1.7/32 extensive
=====
Route Table (Service: 100)
=====
Dest Prefix      : 192.0.1.7/32
Protocol         : BGP_VPN
Age              : 00h34m55s
Preference       : 170
Indirect Next-Hop : 192.0.2.7
Label            : 262143
QoS              : Priority=n/c, FC=n/c
Source-Class     : 0
Dest-Class       : 0
ECMP-Weight      : N/A
Resolving Next-Hop : 192.0.2.7 (RSVP tunnel:2)
Metric           : 10
ECMP-Weight      : N/A
```

On node PE-7, only an LDP LSP and an SR shortest path tunnel (using IS-IS) are present. Because the VPRN service has auto-bind-tunnel resolution any enabled, the protocol type with highest TTM preference (meaning the lowest absolute preference value in TTM) is taken; in this case, the LDP LSP. This can be verified for the configured loopback address within the VPRN service context, as follows:

```
*A:PE-7# show router 100 route-table 192.0.1.1/32 extensive
=====
Route Table (Service: 100)
=====
Dest Prefix      : 192.0.1.1/32
```

Configuration

```
Protocol           : BGP_VPN
Age                : 00h40m21s
Preference         : 170
Indirect Next-Hop  : 192.0.2.1
  Label            : 262142
  QoS               : Priority=n/c, FC=n/c
  Source-Class     : 0
  Dest-Class       : 0
  ECMP-Weight      : N/A
Resolving Next-Hop : 192.0.2.1 (LDP tunnel)
  Metric           : 10
  ECMP-Weight      : N/A
```

Some configuration changes are possible to change this default behavior:

1. It is possible to change the auto-bind-tunnel resolution any command into auto-bind-tunnel resolution filter. Because this is a service-specific parameter, the operator has the choice to only configure this on one specific service endpoint. From a migration point of view, a smooth and easy SR migration is possible, having no affect on other deployed services on this node.
2. It is possible to change the SR tunnel-table protocol preference on a node. From a migration point of view, this affects all services initiating on this node.

Using the current example, PE-1 implements the auto-bind-tunnel change (option 1), while PE-7 implements the TTM preference change (option 2).

First, a resolution-filter CLI context within VPRN service 100 on node PE-1 must be created. The example uses a resolution-filter context, which uses a filter to only allow SR shortest path tunnels (IS-IS based). Translated into configuration commands, this becomes:

```
*A:PE-1# configure service vprn 100 auto-bind-tunnel resolution-filter sr-isis
```

Then, change the auto-bind-tunnel resolution any command into resolution filter on PE-1. Translated into a configuration command, this becomes:

```
*A:PE-1# configure service vprn 100 auto-bind-tunnel resolution filter
```

As a result, the RSVP LSP is no longer used. Instead, the SR shortest path tunnel is used for the traffic from PE-1 to PE-7:

```
*A:PE-1# show router 100 route-table 192.0.1.7/32 extensive
=====
Route Table (Service: 100)
=====
Dest Prefix       : 192.0.1.7/32
Protocol          : BGP_VPN
Age               : 00h00m11s
Preference        : 170
Indirect Next-Hop : 192.0.2.7
  QoS              : Priority=n/c, FC=n/c
```

```
Source-Class      : 0
Dest-Class       : 0
ECMP-Weight      : N/A
Resolving Next-Hop : 192.0.2.7 (ISIS tunnel)
Metric          : 10
ECMP-Weight      : N/A
```

The VPRN service on node PE-7 is still using the LDP LSP as transport mechanism to reach node PE-1 at this point. Because the previous CLI change is only done within the VPRN service context 100 on PE-1, only the direction from PE-1 to PE-7 is affected.

Another way to influence the default TTM preference is shown as follows on the PE-7 node. Using the default behavior, the LDP LSP is used, because of the preference value of 9. If the SR tunnel table preference value is lowered to a value smaller than LDP, for instance 4, the SR shortest path tunnels originating on this node will always have preference compared to LDP LSP. Translated into a configuration command, this becomes:

```
*A:PE-7# configure router isis segment-routing tunnel-table-pref 4
*A:PE-7# show router tunnel-table 192.0.2.1
=====
IPv4 Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId Pref      Nexthop      Metric
-----
192.0.2.1/32     isis (0)  MPLS  524396   4          192.168.17.1  10
192.0.2.1/32     ldp       MPLS  65537   9          192.168.17.1  10
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
```

As a result, the LDP LSP is no longer used and the SR shortest path tunnel is the preferred transport tunnel:

```
*A:PE-7# show router 100 route-table 192.0.1.1/32 extensive
=====
Route Table (Service: 100)
=====
Dest Prefix      : 192.0.1.1/32
Protocol         : BGP_VPN
Age              : 00h00m01s
Preference       : 170
Indirect Next-Hop : 192.0.2.1
  QoS            : Priority=n/c, FC=n/c
  Source-Class   : 0
  Dest-Class     : 0
  ECMP-Weight    : N/A
  Resolving Next-Hop : 192.0.2.1 (ISIS tunnel)
  Metric        : 10
  ECMP-Weight    : N/A
```

At this point, within the VPRN service, the SR shortest path tunnels are used bidirectionally between PE-1 and PE-7.

Configuration

If, for example, an operator configures explicit SDP binding within the same VPRN service on both endpoints, the explicit SDPs will always have preference. In this example, manual SDPs are configured on nodes PE-1 and PE-7, both using LDP. Translated into configuration commands, this becomes:

```
*A:PE-1# configure service sdp 17 mpls create
      far-end 192.0.2.7
      ldp
      keep-alive
      shutdown
      exit
      no shutdown

*A:PE-1# configure service vprn 100 spoke-sdp 17 create
*A:PE-7# configure service sdp 71 mpls create
      far-end 192.0.2.1
      ldp
      keep-alive
      shutdown
      exit
      no shutdown

*A:PE-7# configure service vprn 100 spoke-sdp 71 create
```

As a result, SR shortest path tunnels are no longer used, but rather LDP-based SDPs are used instead:

```
*A:PE-1# show router 100 route-table 192.0.1.7/32 extensive
=====
Route Table (Service: 100)
=====
Dest Prefix      : 192.0.1.7/32
Protocol         : BGP_VPN
Age              : 00h00m29s
Preference       : 170
Indirect Next-Hop : 192.0.2.7
Label            : 262143
QoS              : Priority=n/c, FC=n/c
Source-Class     : 0
Dest-Class       : 0
ECMP-Weight      : N/A
Resolving Next-Hop : 192.0.2.7 (SDP tunnel:17)
Metric           : 10
ECMP-Weight      : N/A

*A:PE-7# show router 100 route-table 192.0.1.1/32 extensive
=====
Route Table (Service: 100)
=====
Dest Prefix      : 192.0.1.1/32
Protocol         : BGP_VPN
Age              : 00h10m11s
Preference       : 170
Indirect Next-Hop : 192.0.2.1
Label            : 262142
QoS              : Priority=n/c, FC=n/c
```

Segment Routing with IS-IS Control Plane

```
Source-Class      : 0
Dest-Class        : 0
ECMP-Weight       : N/A
Resolving Next-Hop : 192.0.2.1 (SDP tunnel:71)
  Metric          : 10
  ECMP-Weight      : N/A
```

Conclusion

Segment Routing is a technique using extensions of the existing link state protocols, and using existing MPLS or IPv6 infrastructure as the data plane. It is a source routing technique similar to RSVP-TE, but without the need to run an extra signaling protocol. SR also avoids other scaling restrictions of associated RSVP-TE, such as midpoint state. SR is simple to control and operate because the intelligence and state are part of the packet, not held by the network. Other benefits are that SR can be introduced in an incremental way using different migration scenarios to assure a smooth transition.

Shared Risk Link Groups for RSVP-Based LSP

In This Chapter

This section provides information about Shared Risk Link Groups for RSVP-Based LSPs.

Topics in this section include:

- [Applicability on page 726](#)
- [Overview on page 727](#)
- [Configuration on page 729](#)
- [Conclusion on page 747](#)

Applicability

This feature is applicable to all of the 7750 SR and 7450 ESS series, and is tested on release 13.0.R1. No prerequisites are needed.

Overview

Introduction

Shared Risk Link Group (SRLG) is a feature which allows the user to establish a backup secondary LSP (label switched path) path or a FRR (fast-reroute) LSP path which is disjoint from the path of the primary LSP. Links which are members of the same SRLG represent resources which share the same risk. For example, fiber links sharing the same conduit or multiple wavelengths sharing the same fiber.

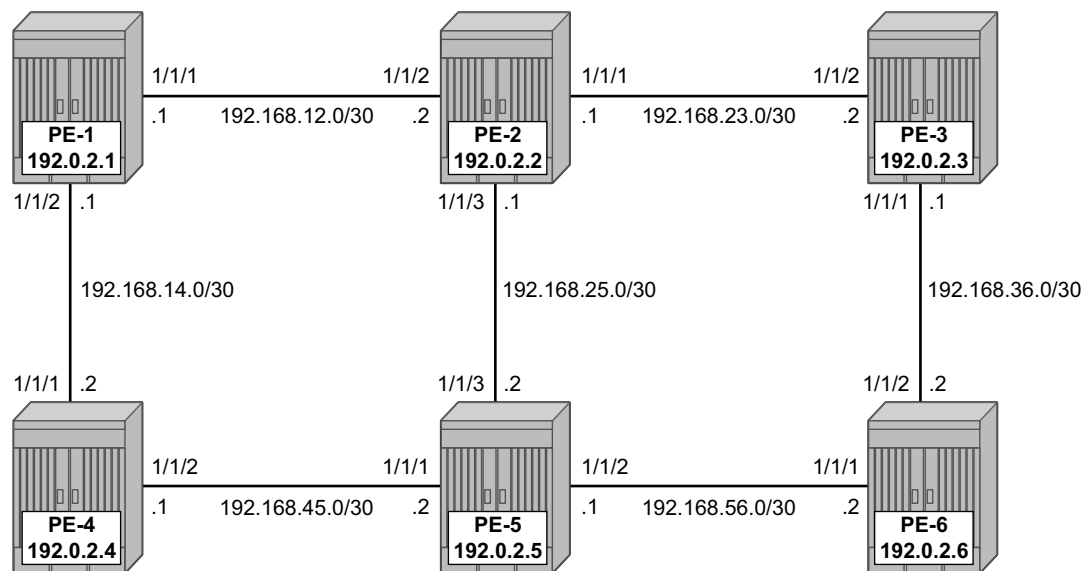
A typical application of the SRLG feature is to provide an automatic placement of secondary backup LSPs or FRR bypass/detour LSPs that minimizes the probability of fate sharing with the path of the primary LSP.

SRLG groups are used to determine which links belong to the same SRLG. The mechanism is similar to MPLS admin groups. To advertise SRLG, the information is part of the IGP TE parameters in an opaque LSA (link state advertisement). The SRLG is advertised in a new Shared Risk Link Group TLV (type 138) in IS-IS (RFC 4205, *Intermediate System to Intermediate System (IS-IS) Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)*). It is advertised in a new SRLG sub-TLV (type 16) of the existing Link TLV in OSPF (RFC 4203, *OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)*).

For FRR a choice can be made on what to do when no FRR tunnel can be found with the SRLG constraints. No FRR tunnel might be signaled or a FRR tunnel might be signaled not taking the SRLG constraints into account.

SRLG

Figure 118 displays the initial topology for this section.



OSSG413

Figure 118: Initial Topology

A single IGP area (IS-IS in this case) with traffic engineering enabled is required for the SRLG feature to work properly.

When OSPF is used as the IGP, the functionality is similar.

Configuration

Step 1. Configuring the IP/MPLS network.

This is part of the general P2P LSP configuration. For more details check the related configurations of the PE-nodes.

In addition, ECMP is set to 2, instead of the default value 1 in order to highlight the application of SRLG in the final example.

```
A:PE-1# configure router ecmp 2
A:PE-1#
```

Step 2. Define the SRLG groups, and link them to the related MPLS interfaces.

Two SRLG groups are defined, named blue and grey. On following drawing the related IP/MPLS interfaces are indicated.

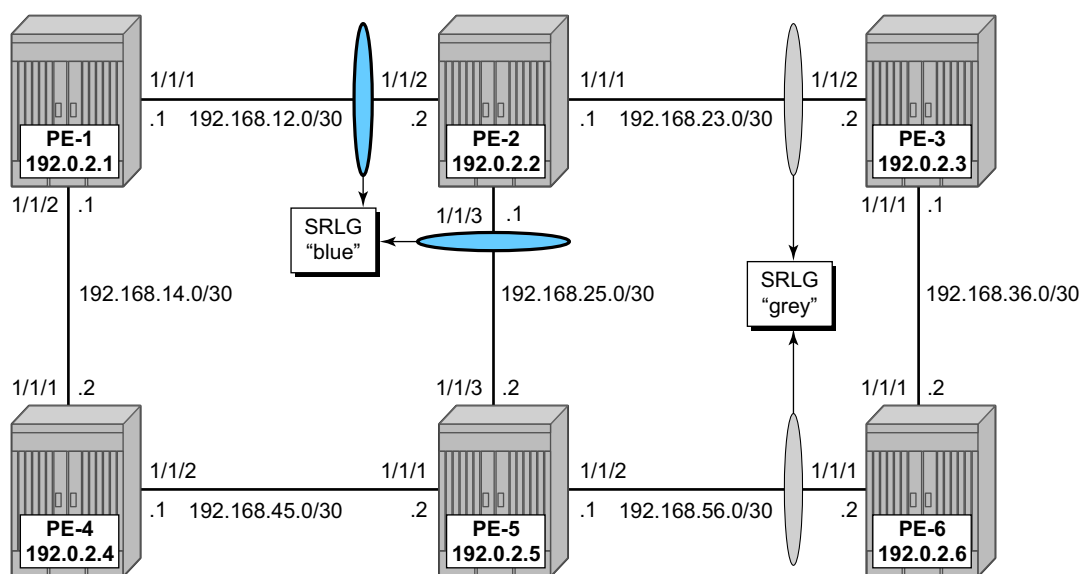


Figure 119: SRLG Topology

From configuration point of view, both SRLG groups must be configured on all nodes as follows.

```
*A:PE-1# configure router if-attribute srlg-group blue value 1
*A:PE-1# configure router if-attribute srlg-group grey value 2
```

The IP/MPLS interfaces need to be linked to the related SRLG group, which is a uni-directional indicator, applying only at the egress direction; hence, it needs to be configured on both sides of

the IP/MPLS interface. For example on PE-1, the interface to PE-2 is part of **srlg-group blue**. Note that an interface can be part of multiple SRLG groups similar to the admin-group functionality.

```
A:PE-1>config>router>if-attr# info
-----
      admin-group "green" value 1
      admin-group "red" value 2
      srlg-group "blue" value 1
      srlg-group "grey" value 2

*A:PE-1>config>router>mpls# info
-----
      interface "system"
        no shutdown
      exit
      interface "int-PE-1-PE-2"
        admin-group "green"
        no shutdown
      exit
      interface "int-PE-1-PE-4"
        admin-group "red"
        no shutdown
      exit

*A:PE-1>config>router>mpls# interface "int-PE-1-PE-2"
*A:PE-1>config>router>mpls>if# srlg-group "blue"
```

The same must be done on PE-2, PE-3, PE-5 and PE-6. Afterwards, verify the MPLS configuration for example on PE-2, where the SRLG groups are linked to the interfaces. Admin-groups are configured in parallel to indicate that both can be configured and will work independently.

```
A:PE-2>config>router>mpls# info
-----
      interface "system"
        no shutdown
      exit
      interface "int-PE-2-PE-1"
        admin-group "green"
        srlg-group "blue"
        no shutdown
      exit
      interface "int-PE-2-PE-3"
        admin-group "green"
        srlg-group "grey"
        no shutdown
      exit
      interface "int-PE-2-PE-5"
        srlg-group "blue"
        no shutdown
      exit
      no shutdown
-----
A:PE-2>config>router>mpls#
```

Some useful show commands to verify the SRLG configuration.

To show all SRLG groups on the node:

```
*A:PE-2# show router if-attribute srlg-group
=====
Interface Srlg Groups
=====
Group Name                Group Value    Penalty Weight
-----
blue                      1              0
grey                      2              0
-----
No. of Groups: 2
=====
*A:PE-2#
```

In the list of MPLS interfaces, admin groups and SRLG groups are indicated.

```
A:PE-2# show router mpls interface
=====
MPLS Interfaces
=====
Interface                Port-id        Adm   Opr   TE-metric
-----
system                   system         Up    Up    None
  Admin Groups           None
  SRLG Groups            None
int-PE-2-PE-1            1/1/2         Up    Up    None
  Admin Groups           green
  SRLG Groups            blue
int-PE-2-PE-3            1/1/1         Up    Up    None
  Admin Groups           green
  SRLG Groups            grey
int-PE-2-PE-5            1/1/3         Up    Up    None
  Admin Groups           None
  SRLG Groups            blue
-----
Interfaces : 4
=====
A:PE-2#
```

To verify the SRLG groups in the IGP TE database, the following command can be used. The output can be extensive but searching on the SRLG group name will lead to the correct interface(s).

As an example, following command shows the link-state advertisements of PE-2 on PE-1 in this case. Note that the SRLG information is linked to the IP interfaces in a dedicated (TE-)TLV.

```
*A:PE-1# show router isis database PE-2.00-00 detail
=====
Router Base ISIS Instance 0 Database
=====

Displaying Level 1 database
-----
LSP ID       : PE-2.00-00                      Level       : L1
Sequence     : 0x69                            Checksum    : 0x5f3d  Lifetime    : 962
Version      : 1                               Pkt Type    : 18     Pkt Ver     : 1
Attributes: L1L2                             Max Area    : 3
SysID Len    : 6                               Used Len    : 508    Alloc Len   : 508

TLVs :

<snipped>

TE SRLGs      :
  SRLGs : PE-1.00
  Lcl Addr : 192.168.12.2
  Rem Addr : 192.168.12.1
  Num SRLGs : 1
  1

<snipped>

TE SRLGs      :
  SRLGs : PE-3.00
  Lcl Addr : 192.168.23.1
  Rem Addr : 192.168.23.2
  Num SRLGs : 1
  2

<snipped>

TE SRLGs      :
  SRLGs : PE-5.00
  Lcl Addr : 192.168.25.1
  Rem Addr : 192.168.25.2
  Num SRLGs : 1
  1

<snipped>
```


On-Line Verification

An on-line verification can be done by a **tools perform** CLI command. This will trigger a real CSPF call to the IGP TE database, and the result will be an ERO object which can potentially be used to set-up a CSPF based LSP.

The following shows the command syntax.

```
*A:PE-1# tools perform router mpls cspf
- cspf to <ip-addr> [from <ip-addr>] [bandwidth <bandwidth>] [include-bitmap <bitmap>]
[exclude-bitmap
  <bitmap>] [hop-limit <limit>] [exclude-address <excl-addr> [<excl-addr>...(upto 8
max)]]
  [use-te-metric] [strict-srlg] [srlg-group <grp-id>...(upto 8 max)] [exclude-node
<excl-node-id>
  [<excl-node-id>...(upto 8 max)]] [skip-interface <interface-name>] [ds-class-type
<class-type>]
  [cspf-reqtype <req-type>] [least-fill-min-thd <thd>] [setup-priority <val>] [hold-pri-
ority <val>]

<ip-addr>           : a.b.c.d
<bandwidth>         : [1..100000] in Mbps
<bitmap>            : [0..4294967295] - accepted in decimal, hex(0x) or binary(0b)
<limit>             : [2..255]
<excl-addr>         : a.b.c.d (outbound interface)
<use-te-metric>     : keyword
<strict-srlg>       : keyword
<grp-id>            : [0..4294967295]
<excl-node-id>      : [a.b.c.d] (outbound interface)
<interface-name>    : [max 32 chars]
<class-type>        : [0..7]
<req-type>          : all|random|least-fill : keywords
<thd>               : [1..100]
<priority>          : [0..7]
```

Where the relevant parameters are:

- **to** — Defines the far-end address of the LSP. This is the system-address of the destination LER
- **srlg-group** — Specifies which SRLG groups should be avoided while building the path to the destination (ERO object)
- **strict-srlg** — Indicates whether the SRLG group is a strict requirement or not. When this parameter is given, only paths without traversing the SRLG will be displayed.

An example:

On PE-1 a CSPF calculation is made with PE-3 as destination, without any SRLG restrictions, this will look like the following output:

```
*A:PE-1# tools perform router mpls cspf to 192.0.2.3
Req CSPF for all ECMP paths
  from: this node to: 192.0.2.3 w/(no Diffserv) class: 0 , setup Priority 7, Hold Priority 0 TE Class: 7

CSPF Path
To      : 192.0.2.3
Path 1  : (cost 20)
  Src:   192.0.2.1   (= Rtr)
  Egr:   192.168.12.1 -> Ingr: 192.168.12.2      Rtr: 192.0.2.2      (met 10)
  Egr:   192.168.23.1 -> Ingr: 192.168.23.2      Rtr: 192.0.2.3      (met 10)
  Dst:   192.0.2.3   (= Rtr)

*A:PE-1#
```

Given a restriction on **srlg-group blue** (grp-id =1), the result is as follows.

```
*A:PE-1# tools perform router mpls cspf to 192.0.2.3 srlg-group 1
Req CSPF for all ECMP paths
  from: this node to: 192.0.2.3 w/(no Diffserv) class: 0 , setup Priority 7, Hold Priority 0 TE Class: 7

CSPF Path
To      : 192.0.2.3
Path 1  : (cost 40)
  Src:   192.0.2.1   (= Rtr)
  Egr:   192.168.14.1 -> Ingr: 192.168.14.2      Rtr: 192.0.2.4      (met 10)
  Egr:   192.168.45.1 -> Ingr: 192.168.45.2      Rtr: 192.0.2.5      (met 10)
  Egr:   192.168.56.1 -> Ingr: 192.168.56.2      Rtr: 192.0.2.6      (met 10)
  1 SRLGs: 2
  Egr:   192.168.36.2 -> Ingr: 192.168.36.1      Rtr: 192.0.2.3      (met 10)
  Dst:   192.0.2.3   (= Rtr)

*A:PE-1#
```

The path will be through PE-4, PE-5 and PE-6.

When a strict restriction is requested on **srlg-group grey**, no valid CSPF path towards the destination can be found. Removing the **strict** restriction results in a successful return of CSPF.

```
*A:PE-1# tools perform router mpls cspf to 192.0.2.3 srlg-group 2 strict-srlg
Req CSPF for all ECMP paths
  from: this node to: 192.0.2.3 w/(no Diffserv) class: 0 , setup Priority 7, Hold Priority 0 TE Class: 7

MINOR: CLI No CSPF path to "192.0.2.3" with specified constraints.
*A:PE-1#
```

Shared Risk Link Groups for RSVP-Based LSP

```
*A:PE-1# tools perform router mpls cspf to 192.0.2.3 srlg-group 2
Req CSPF for all ECMP paths
  from: this node to: 192.0.2.3 w/(no Diffserv) class: 0 , setup Priority 7, Hold Priority 0 TE Class: 7
```

```
CSPF Path
To      : 192.0.2.3 (NOT SRLG DISJOINT)
Path 1  : (cost 20)
  Src:   192.0.2.1   (= Rtr)
    Egr: 192.168.12.1 -> Ingr: 192.168.12.2      Rtr: 192.0.2.2      (met 10)
      1 SRLGs: 1
    Egr: 192.168.23.1 -> Ingr: 192.168.23.2      Rtr: 192.0.2.3      (met 10)
      1 SRLGs: 2
  Dst:   192.0.2.3   (= Rtr)
```

```
*A:PE-1#
```

The best practice for debugging is to enable debug-tracing on the CSPF process, with following command.

```
*A:PE-1# debug router isis cspf
```

SRLG for FRR

The fast-reroute mechanism used here is facility link-protection. The SRLG feature is independent of the FRR type and works for all combinations (facility versus one-to-on, link versus node protection).

Step 1. Configure an LSP.

An LSP from PE-1 to PE-3 will be created, CSPF based.

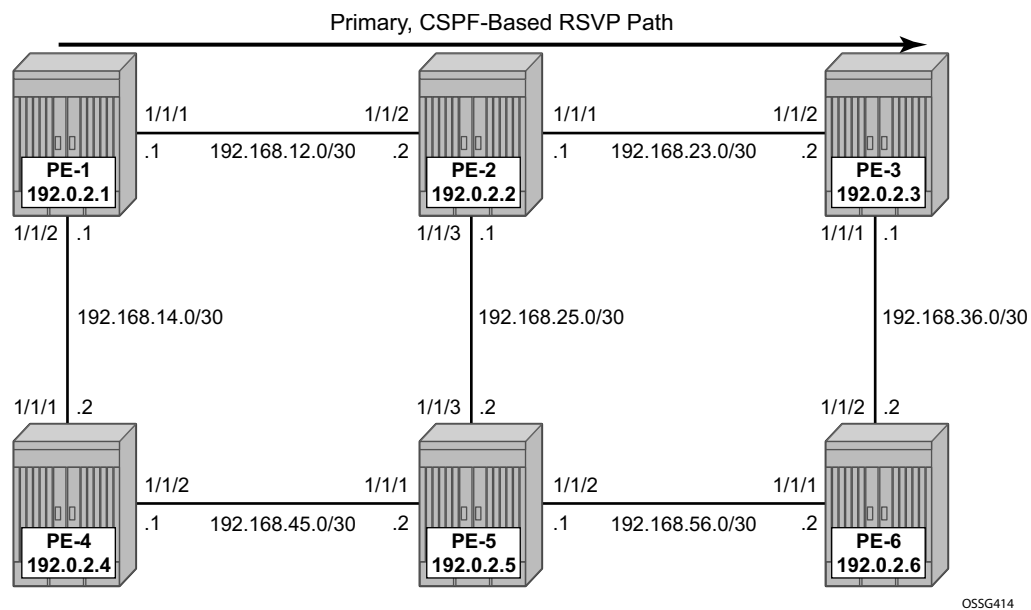


Figure 120: Path Primary RSVP_TE LSP

The configuration of the LSP lsp-PE-1-PE-3-FRR-facility-link is based on an empty path, with FRR facility link protection enabled.

```
*A:PE-1>config>router>mpls>lsp# info
-----
to 192.0.2.3
 cspf
 fast-reroute facility
   no node-protect
 exit
 primary "loose"
 exit
 no shutdown
-----
*A:PE-1>config>router>mpls>lsp#
```

To verify the primary path, **oam lsp-trace** command can be used, checking the intermediate nodes.

```
*A:PE-1# oam lsp-trace "lsp-PE-1-PE-3_FRR_facility-link" detail
lsp-trace to lsp-PE-1-PE-3_FRR_facility-link: 0 hops min, 0 hops max, 116 byte packets
1 192.0.2.2 rtt=3.23ms rc=8(DSRtrMatchLabel) rsc=1
    DS 1: ipaddr=192.168.23.2 ifaddr=192.168.23.2 iftype=ipv4Numbered MRU=1564
        label[1]=131069 protocol=4(RSVP-TE)
2 192.0.2.3 rtt=3.31ms rc=3(EgressRtr) rsc=1
*A:PE-1#
```

To check if the bypass tunnels are up and running, an indication (@) can be found in the detail output of **show router mpls ls <x> path detail** as seen in the following output.

```
*A:PE-1# show router mpls lsp "lsp-PE-1-PE-3_FRR_facility-link" path detail
=====
MPLS LSP lsp-PE-1-PE-3_FRR_facility-link Path (Detail)
=====
Legend :
    @ - Detour Available          # - Detour In Use
    b - Bandwidth Protected       n - Node Protected
    s - Soft Preemption
    S - Strict                    L - Loose
    A - ABR

=====
-----
LSP lsp-PE-1-PE-3_FRR_facility-link Path loose
-----
-----
LSP Name      : lsp-PE-1-PE-3_FRR_facility-link      Path LSP ID : 24074
From          : 192.0.2.1                            To          : 192.0.2.3
Adm State     : Up                                    Oper State  : Up
Path Name     : loose                                Path Type   : Primary
Path Admin    : Up                                    Path Oper   : Up
OutInterface  : 1/1/1                                Out Label   : 131070
Path Up Time  : 0d 00:06:25                          Path Dn Time: 0d 00:00:00
Retry Limit   : 0                                    Retry Timer : 30 sec
RetryAttempt  : 0                                    NextRetryIn : 0 sec

AdsSpec       : Disabled                             Oper AdsSpec : Disabled
CSPF          : Enabled                               Oper CSPF    : Enabled
Least Fill    : Disabled                             Oper LeastF* : Disabled
FRR           : Enabled                               Oper FRR     : Enabled
FRR NodePro*  : Disabled                             Oper FRR NP  : Disabled
FR Hop Limit  : 16                                   Oper FRHopL* : 16
FR Prop Adm*  : Disabled                             Oper FRProp* : Disabled
Prop Adm Grp  : Disabled                             Oper PropAG  : Disabled
Inter-area    : False

Neg MTU       : 1560                                  Oper MTU     : 1560
Bandwidth     : No Reservation                        Oper Bw      : 0 Mbps
Hop Limit     : 255                                   Oper HopLim* : 255
Record Route  : Record                               Oper RecRou* : Record
Record Label  : Record                               Oper RecLab* : Record
SetupPriori*  : 7                                    Oper SetupP* : 7
Hold Priori*  : 0                                    Oper HoldPr* : 0
Class Type    : 0                                    Oper CT      : 0
Backup CT     : None
```

SRLG for FRR

```
MainCT Retry: n/a
  Rem      :
MainCT Retry: 0
  Limit    :
Include Grps:
None
Exclude Grps:
None
Adaptive    : Enabled
Preference  : n/a
Path Trans  : 8
Failure Code: noError
ExplicitHops:
  No Hops Specified
Actual Hops :
  192.168.12.1 (192.0.2.1) @
  -> 192.168.12.2 (192.0.2.2) @
  -> 192.168.23.2 (192.0.2.3)
ComputedHops:
  192.168.12.1(S)
  -> 192.168.12.2(S)
  -> 192.168.23.2(S)
ResigEligib*: False
LastResignal: n/a
Oper InclGr*:
None
Oper ExclGr*:
None
Oper Metric : 20
CSPF Queries: 1876
Failure Node: n/a
Record Label : N/A
Record Label : 131070
Record Label : 131069
CSPF Metric : 20
=====
* indicates that the corresponding row element may have been truncated.
*A:PE-1#
```

The expected path(s) followed by the bypass tunnels are shown in [Figure 121](#).

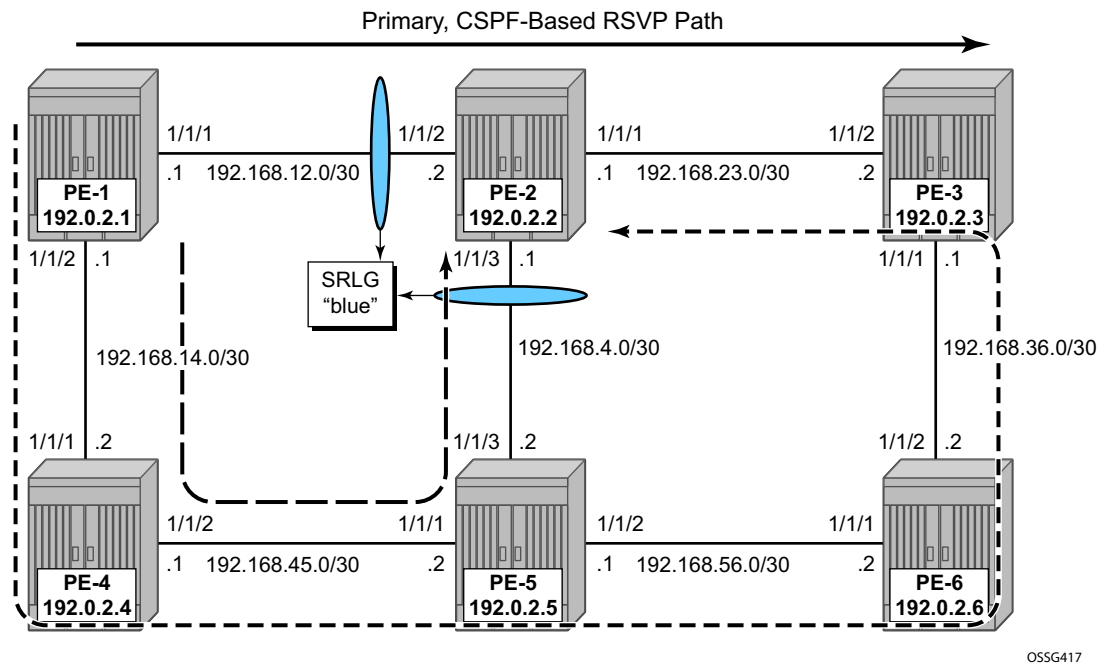


Figure 121: SRLG for FRR Path With and Without SRLG

To verify the data path on the point of local repair (PLR), the next CLI commands can be used.

```
*A:PE-1# show router mpls bypass-tunnel detail
=====
MPLS Bypass Tunnels (Detail)
=====
-----
bypass-link192.168.12.2-61443
-----
-----
To          : 192.168.25.1      State       : Up
Out I/F     : 1/1/2            Out Label   : 131071
Up Time     : 0d 00:06:35      Active Time  : n/a
Reserved BW : 0 Kbps           Protected LSP Count : 1
Type        : Dynamic          Bypass Path Cost : 30
Setup Priority : 7              Hold Priority  : 0
Class Type   : 0
Exclude Node : None
Computed Hops :
    192.168.14.1 (S)
    -> 192.168.14.2 (S)
    -> 192.168.45.2 (S)
    -> 192.168.25.1 (S)
Actual Hops  :
    192.168.14.1 (192.0.2.1)
Egress Admin Groups :
red
Egress Admin Groups :
red
Egress Admin Groups : None
Egress Admin Groups : None
Inter-Area         : False
Record Label       : N/A
```

SRLG for FRR

```
-> 192.168.14.2 (192.0.2.4)      Record Label      : 131071
-> 192.168.45.2 (192.0.2.5)      Record Label      : 131070
-> 192.168.25.1 (192.0.2.2)      Record Label      : 131069
Last Resignal   :
Attempted At    : n/a            Resignal Reason    : n/a
Resignal Status: n/a            Reason              : n/a
```

```
=====
*A:PE-1#
```

The SRLG restriction is not taken into account at this moment at PLR PE-1. The actual hops are PE-4, PE-5 and PE-2 visualized by the dashed path in [Figure 121](#).

To take the SRLG restrictions into account, additional configuration is needed for MPLS.

```
*A:PE-1>config>router>mpls# srlg-
srlg-database srlg-frr

*A:PE-1>config>router>mpls# srlg-frr
- no srlg-frr
- srlg-frr [strict]

<strict>                : keyword

*A:PE-1>config>router>mpls# srlg-frr strict
*A:PE-1>config>router>mpls# info
-----
      srlg-frr strict
      interface "system"
          no shutdown
      exit

<snipped>
```

The option **strict** should only be taken if the logical topology allows this. In other words, one must be sure that an alternative path is possible which avoids SRLG-groups.

After applying the SRLG FRR feature, the related LSP needs to be resigaled in order to set up the bypass tunnel with the new constraints.

```
*A:PE-1# tools perform router mpls resignal lsp "lsp-PE-1-PE-3_FRR_facility-link" path
"loose"
*A:PE-1#
```


This can be verified with previous commands.

```
*A:PE-1# show router mpls bypass-tunnel detail
=====
MPLS Bypass Tunnels (Detail)
=====
-----
bypass-link192.168.12.2-61444
-----
To          : 192.168.23.1      State          : Up
Out I/F     : 1/1/2            Out Label     : 131070
Up Time    : 0d 00:00:16       Active Time    : n/a
Reserved BW : 0 Kbps           Protected LSP Count : 1
Type       : Dynamic           Bypass Path Cost : 50
Setup Priority : 7              Hold Priority   : 0
Class Type  : 0
Exclude Node : None            Inter-Area     : False
Computed Hops :
    192.168.14.1 (S)           Egress Admin Groups :
                                red
    -> 192.168.14.2 (S)         Egress Admin Groups :
                                red
    -> 192.168.45.2 (S)         Egress Admin Groups :
                                red
    -> 192.168.56.2 (S)         Egress Admin Groups :
                                green
    -> 192.168.36.1 (S)         Egress Admin Groups :
                                green
    -> 192.168.23.1 (S)         Egress Admin Groups : None
Actual Hops  :
    192.168.14.1 (192.0.2.1)   Record Label    : N/A
    -> 192.168.14.2 (192.0.2.4) Record Label    : 131070
    -> 192.168.45.2 (192.0.2.5) Record Label    : 131069
    -> 192.168.56.2 (192.0.2.6) Record Label    : 131070
    -> 192.168.36.1 (192.0.2.3) Record Label    : 131068
    -> 192.168.23.1 (192.0.2.2) Record Label    : 131066
Last Resignal :
Attempted At  : n/a            Resignal Reason  : n/a
Resignal Status: n/a           Reason            : n/a
=====
*A:PE-1#
```

This path is represented by the dotted line in [Figure 121](#), taking the SRLG constraints into account.

SRLG for Standby Path

Where SRLG groups will be constraints for bypass tunnels, they will also be a constraint to set-up a secondary path. Looking at the following picture, the secondary path is expected to follow the dotted-line instead of passing over the direct link between PE-5 and PE-2.

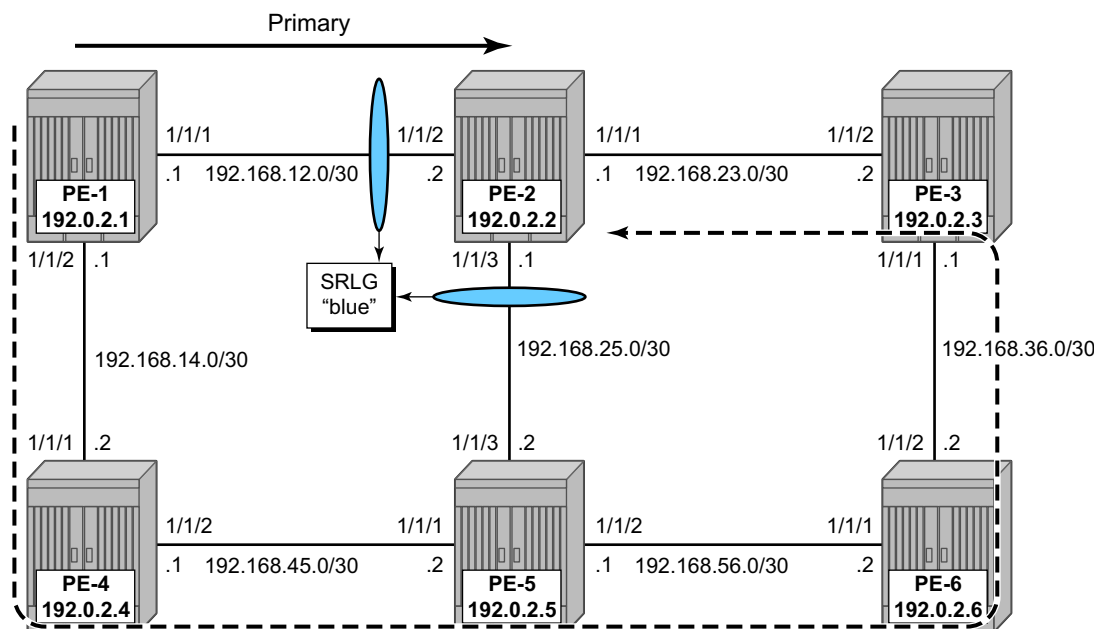


Figure 122: SRLG for Secondary Path

The configuration of the LSP will need a specific indication at the level of the secondary path to enable the restriction on the srlg-groups.

```
*A:PE-1# configure router mpls lsp "lsp-PE-1-PE-2-srlg"
*A:PE-1>config>router>mpls>lsp# info
-----
to 192.0.2.2
cspf
primary "prim"
exit
secondary "secon"
standby
srlg
exit
no shutdown
-----
*A:PE-1>config>router>mpls>lsp#
```

Where both paths are empty paths, the ERO object creation solely relies on CPSF without any specific hop.

To verify the datapath, the detailed output of the **show router mpls** command can be used, as well as the **lsp-trace** OAM command. This output shows both ERO objects of the primary and secondary path.

```
*A:PE-1# show router mpls lsp "lsp-PE-1-PE-2-srlg" path detail
=====
MPLS LSP lsp-PE-1-PE-2-srlg Path (Detail)
=====
Legend :
    @ - Detour Available          # - Detour In Use
    b - Bandwidth Protected       n - Node Protected
    s - Soft Preemption
    S - Strict                    L - Loose
    A - ABR
=====
-----
LSP lsp-PE-1-PE-2-srlg Path prim
-----
<snipped>

ExplicitHops:
    No Hops Specified
Actual Hops :
    192.168.12.1 (192.0.2.1)      Record Label      : N/A
    -> 192.168.12.2 (192.0.2.2)   Record Label      : 131068
ComputedHops:
    192.168.12.1(S)
    -> 192.168.12.2(S)
ResigEligib*: False
LastResignal: n/a                CSPF Metric : 10
-----
LSP lsp-PE-1-PE-2-srlg Path secon
-----
<snipped>

ExplicitHops:
    No Hops Specified
Actual Hops :
    192.168.14.1 (192.0.2.1)      Record Label      : N/A
    -> 192.168.14.2 (192.0.2.4)   Record Label      : 131070
    -> 192.168.45.2 (192.0.2.5)   Record Label      : 131069
    -> 192.168.56.2 (192.0.2.6)   Record Label      : 131069
    -> 192.168.36.1 (192.0.2.3)   Record Label      : 131067
    -> 192.168.23.1 (192.0.2.2)   Record Label      : 131067
ComputedHops:
    192.168.14.1(S)
    -> 192.168.14.2(S)
    -> 192.168.45.2(S)
    -> 192.168.56.2(S)
    -> 192.168.36.1(S)
    -> 192.168.23.1(S)
Srlg          : Enabled
SrlgDisjoint: True
ResigEligib*: False
LastResignal: n/a                CSPF Metric : 50
=====
* indicates that the corresponding row element may have been truncated.
*A:PE-1#
```

The **lsp-trace** command can be used for secondary path as well. The intermediate LSRs and the MPLS labels used can be clearly seen.

```
*A:PE-1# oam lsp-trace "lsp-PE-1-PE-2-srlg" path "secon" detail
lsp-trace to lsp-PE-1-PE-2-srlg: 0 hops min, 0 hops max, 116 byte packets
1  192.0.2.4  rtt=2.27ms rc=8(DSRtrMatchLabel) rsc=1
    DS 1: ipaddr=192.168.45.2 ifaddr=192.168.45.2 iftype=ipv4Numbered MRU=1564
        label[1]=131069 protocol=4 (RSVP-TE)
2  192.0.2.5  rtt=4.42ms rc=8(DSRtrMatchLabel) rsc=1
    DS 1: ipaddr=192.168.56.2 ifaddr=192.168.56.2 iftype=ipv4Numbered MRU=1564
        label[1]=131069 protocol=4 (RSVP-TE)
3  192.0.2.6  rtt=29.8ms rc=8(DSRtrMatchLabel) rsc=1
    DS 1: ipaddr=192.168.36.1 ifaddr=192.168.36.1 iftype=ipv4Numbered MRU=1564
        label[1]=131067 protocol=4 (RSVP-TE)
4  192.0.2.3  rtt=4.48ms rc=8(DSRtrMatchLabel) rsc=1
    DS 1: ipaddr=192.168.23.1 ifaddr=192.168.23.1 iftype=ipv4Numbered MRU=1564
        label[1]=131067 protocol=4 (RSVP-TE)
5  192.0.2.2  rtt=4.23ms rc=3(EgressRtr) rsc=1
*A:PE-1#
```

SRLG Database

In case not all IP/MPLS routers in the area support SRLG, a static SRLG database can be created on the systems which will be used as an additional constraint when performing the CSPF calculation to define the path.

An example can be seen [Figure 123](#) where an additional SRLG group (red) is locally on PE-1, with information related to the interface between PE-4 and PE-5.

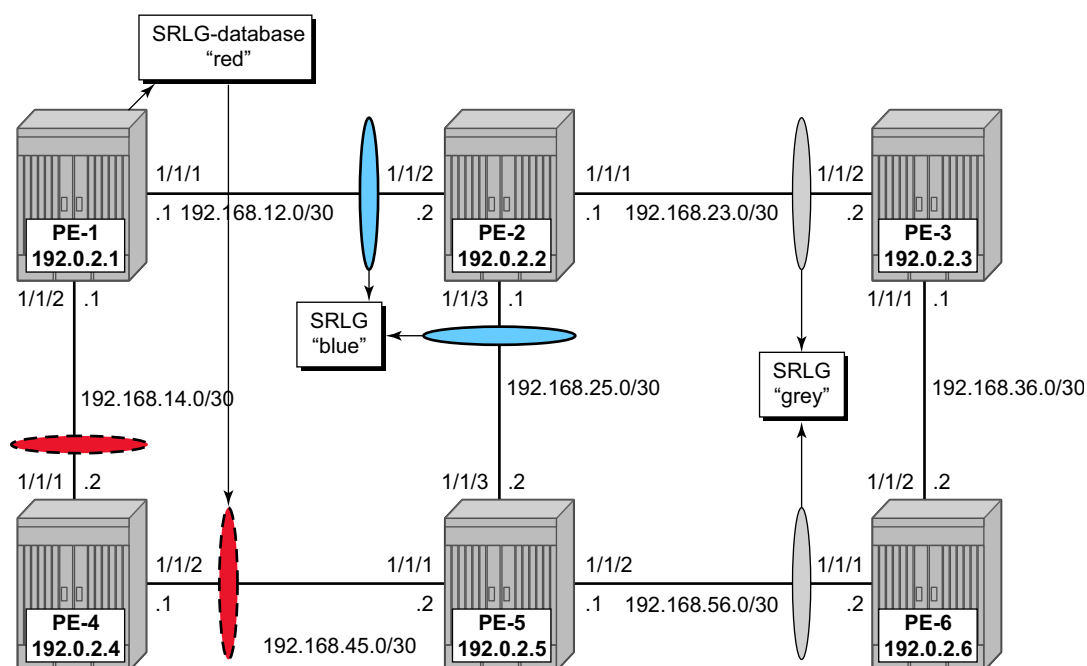


Figure 123: SRLG Database Example

```
*A:PE-1# configure router if-attribute srlg-group "red" value 3

*A:PE-1>config>router>mpls# interface "int-PE-1-PE-4"
*A:PE-1>config>router>mpls>if# srlg-group "red"
*A:PE-1>config>router>mpls>if# exit
*A:PE-1>config>router>mpls# srlg-database
*A:PE-1>config>router>mpls>srlg-database# router-id 192.0.2.4
*A:PE-1>config>router>mpls>srlg-database>router-id# interface 192.168.45.1 srlg-group
"red"
*A:PE-1>config>router>mpls>srlg-database>router-id# no shutdown
*A:PE-1>config>router>mpls>srlg-database>router-id# exit
*A:PE-1>config>router>mpls>srlg-database# router-id 192.0.2.5
*A:PE-1>config>router>mpls>srlg-database>router-id# interface 192.168.45.2 srlg-group
"red"
*A:PE-1>config>router>mpls>srlg-database>router-id# no shutdown
```

SRLG Database

```
*A:PE-1>config>router>mpls>srlg-database>router-id# exit
*A:PE-1>config>router>mpls>srlg-database#
```

```
*A:PE-1>config>router>mpls# info
-----
      srlg-frr strict
      interface "system"
        no shutdown
      exit
      interface "int-PE-1-PE-2"
        admin-group "green"
        srlg-group "blue"
        no shutdown
      exit
      interface "int-PE-1-PE-4"
        admin-group "red"
        srlg-group "red"
        no shutdown
      exit
      srlg-database
        router-id 192.0.2.4
          interface 192.168.45.1 srlg-group "red"
          no shutdown
        exit
        router-id 192.0.2.5
          interface 192.168.45.2 srlg-group "red"
          no shutdown
        exit
      exit
exit
```

Note that this information is only local and will only have effect on CSPF calculations on PE-1, not on the other nodes.

When a CSPF calculation is done for a path from PE-1 to PE-5, the result will be two equal-cost paths. When adding the **srlg-group red** as a restriction, only a single path will be found, passing PE-2.

Conclusion

Interpreting the SRLG information into the TE database makes it possible to protect an LSP even when multiple IP/MPLS interfaces fail as a result of an underlying transmission failure.

Transmission failures can occur quite often since not all transmission links are 1:1 protected.

SRLG groups in MPLS provide a very dynamic and simple way to assure LSP FRR path protection on every PLR throughout the followed LSP path. The SRLG groups are also taken into account when defining the ERO for secondary paths, at least if the configured secondary path is empty.

For interoperability reasons the SRLG-database is available, as systems can link interface to an SRLG with interconnecting systems that do not support the SRLG feature; hence they can not advertise the SRLG information through the IGP.

Note that the creation and maintenance of an SRLG database requires operational effort and systems that do not support SRLG will never take any SRLG information into account during CSPF calculation for the creation of FRR bypass or detour tunnels.

Services Overview

In This Section

This section provides configuration information for the following topics:

- [G.8032 Ethernet Ring Protection Multiple Ring Topology on page 751](#)
- [G.8032 Ethernet Ring Protection Single Ring Topology on page 799](#)

G.8032 Ethernet Ring Protection Multiple Ring Topology

In This Chapter

This section provides information about G.8032 Ethernet ring protection multiple ring topologies.

Topics in this section include:

- [Applicability on page 752](#)
- [Overview on page 753](#)
- [Configuration on page 761](#)
- [Conclusion on page 798](#)

Applicability

This example is applicable to the 7950 XRS (as of 10.0.R4), the 7750 SR-7/12 and 7450 ESS-7/12 (as of 9.0.R1), and the 7450 ESS-6/6v with IOM3-XP or IMM and 7750 SR-c4/12 (as of 11.0.R1). It is not supported on a 7750 SR-1, 7450 ESS-1, 7710 SR, or using an IOM-2 or lower.

The configuration was tested on release 12.0.R5 and covers both a single ring and multiple ring topologies.

Overview

G.8032 Ethernet ring protection is supported for data service SAPs within a regular VPLS service, a PBB VPLS (I/B-component) or a routed VPLS (R-VPLS). G.8032 is one of the fastest protection schemes for Ethernet networks. This example covers the advanced topic of Multiple Ring Control, sometimes referred to as multi-chassis protection, with access rings being the most common form of multiple ring topologies. Single Rings are covered in [G.8032 Ethernet Ring Protection Single Ring Topology on page 799](#). This example will use a VPLS service to illustrate the configuration of G.8032. For very large ring topologies, Provider Backbone Bridging (PBB) can also be used but is not configured in this example.

ITU-T G.8032v2 specifies protection switching mechanisms and a protocol for Ethernet layer network (ETH) Ethernet rings. Ethernet rings can provide wide-area multipoint connectivity more economically due to their reduced number of links. The mechanisms and protocol defined in ITU-T G.8032v2 are highly reliable with stable protection and never form loops, which would negatively affect network operation and service availability. Each ring node is connected to adjacent nodes participating in the same ring using two independent paths, which use ring links (configured on ports or LAGs). A ring link is bounded by two adjacent nodes and a port for a ring link is called a ring port. The minimum number of nodes on a ring is two.

The fundamentals of this ring protection switching architecture are:

- the principle of loop avoidance and
- the utilization of learning, forwarding, and address table mechanisms defined in the ITU-T G.8032v2 Ethernet flow Forwarding Function (ETH_FF) (Control plane).

Loop avoidance in the ring is achieved by guaranteeing that, at any time, traffic may flow on all but one of the ring links. This particular link is called the Ring Protection Link (RPL) and under normal conditions this link is blocked, so it is not used for traffic. One designated node, the RPL Owner, is responsible to block traffic over the one designated RPL. Under a ring failure condition, the RPL Owner is responsible for unblocking the RPL, allowing the RPL to be used for traffic. The protocol ensures that even without an RPL owner defined, one link will be blocked and it operates as a **break before make** protocol, specifically the protocol guarantees that no link is restored until a different link in the ring is blocked. The other side of the RPL is configured as an RPL neighbor. An RPL neighbor blocks traffic on the link.

The event of a ring link or ring node failure results in protection switching of the traffic. This is achieved under the control of the ETH_FF functions on all ring nodes. A Ring Automatic Protection Switching (R-APS) protocol is used to coordinate the protection actions over the ring. The protection switching mechanisms and protocol supports a multi-ring/ladder network that consists of connected Ethernet rings.

Ring Protection Mechanism

The Ring Protection protocol is based on the following building blocks:

- Ring status change on failure
 - Idle -> Link failure -> Protection -> Recovery -> Idle
- Ring Control State changes
 - Idle -> Protection -> Manual Switch -> Forced Switch -> Pending
- Re-use existing ETH OAM
 - Monitoring: ETH Continuity Check messages
 - Failure Notification: Y.1731 Signal Failure
- Forwarding Database MAC Flush on ring status change
- RPL (Ring Protection Link)
 - Defines blocked link in idle status

When sub-rings are used they can either connect to a major ring (which is configured in the exact same way as a single ring) or another sub-ring, or to a VPLS service. When connected to a major/sub ring, there is the option to extend the sub-ring control service through the major ring or not. This gives the following three options for sub-ring connectivity:

1. **Sub-ring to a major/sub ring with a virtual channel** — In this case, a data service on the major/sub ring is created which is used to forward the R-APS messages for the sub-ring over the major/sub ring, between the interconnection points of the sub-ring to the major/sub ring. This allows the sub-ring to operate as a fully connected ring and is mandatory if the sub-ring connects two major/sub rings since the virtual channel is the only mechanism that the sub-rings can use to exchange control messages. It also could improve failover times if the sub-ring was large as it provides two paths on the sub-ring interconnection nodes to propagate the fault indication around the sub-ring, whereas without a virtual channel the fault indication may need to traverse the entire sub-ring. Each sub-ring requires its own data service on the major/sub ring for the virtual channel.
2. **Sub-ring to a major/sub ring without a virtual channel** — In this case the sub-ring is not fully connected and does not require any resources on the major/sub ring. This option requires that the R-APS messages are not blocked on the sub-ring over its RPL.
3. **Sub-ring to a VPLS service** — This is similar to (2) above but uses a VPLS service instead of a major ring. In this option, sub-ring failures can initiate the sending of an LDP MAC flush message into the VPLS service when spoke or MPLS mesh SDPs are used in the VPLS service.

Eth-Ring Terminologies

The implementation of Ethernet Ring (eth-ring) on an SR/ESS uses a VPLS as the construct for a ring flow function (one for ETH_FF (solely for control) and one for each service_FF) and SAPs (on ports or LAGs) as ring links. The control VPLS must be a regular VPLS but the data VPLS can be a regular VPLS, a PBB (B/I-) VPLS or a routed VPLS. The state of the data service SAPs is inherited from the state of the control service SAPs. [Table 4](#) displays a comparison between the ITU-T and SR/ESS terminologies.

Table 4: Terminology Comparison

ITU-T G.8032v2 Terminology	SR/ESS Terminology
ETH_FF	control vpls
Service_FF	data vpls
East Ring Link	path a
West Ring Link	path b
RPL owner	rpl-node owner
RPL Link	path {a b} rpl-end
MEP	control-mep
ERP control process	eth-ring instance or ring-id
Major Ring	eth-ring
Sub-ring	eth-ring sub-ring
Ring node	Ring Node PE
Ring-ID	Not used; fixed at 1 per G.8032v2

There are various ways that multiple rings can be interconnected and the possible topologies may be large. Customers typically have two forms of networks; access ring edge networks or larger multiple ring networks. Both topologies require ring interconnection.

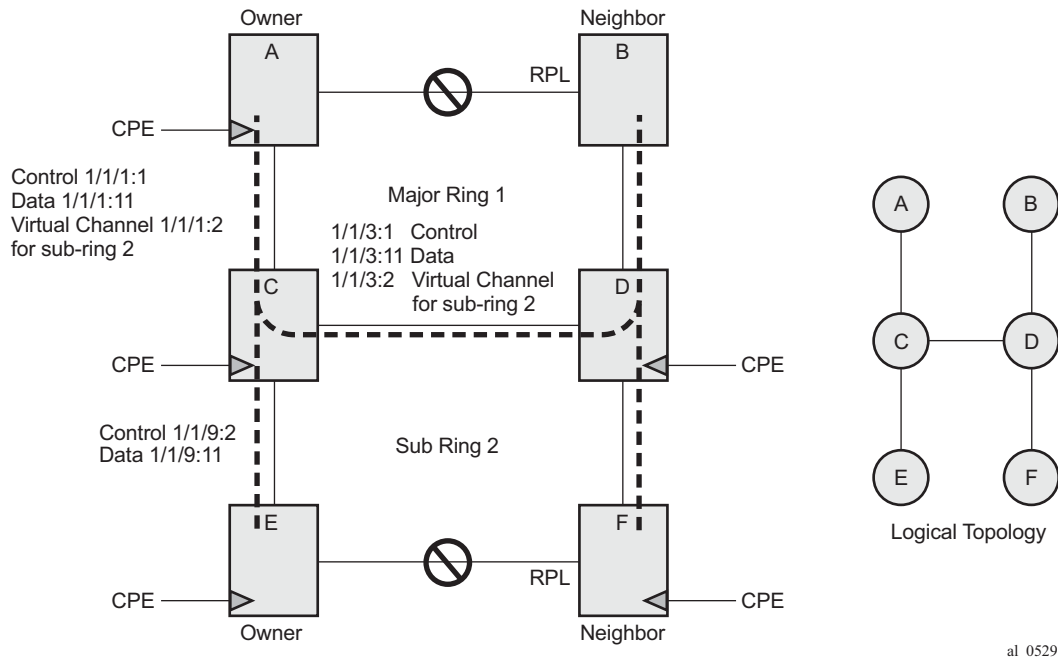


Figure 124: G.8032 Major Ring and Sub-Ring

Figure 124 shows a ring of six nodes, with a major ring (regular Ethernet ring) on the top four nodes and a sub-ring on the bottom. A major ring is a fully connected ring. A sub-ring is a partial ring that depends on a major ring or a VPLS topology for part of the ring interconnect. Two major rings can be connected by a single sub-ring or a sub-ring can support other sub-rings.

In the major ring (on nodes A, B, C and D), one path of the RPL owner is designated to be the RPL and the respective SAPs will be blocked in order to prevent a loop. The choice of where to put the RPL is up to the network administrator and can be different for different control instances of the ring allowing an RPL to be used for some other ring's traffic. In the sub-ring, one path is designated as the RPL and will be blocked. Both the major ring and the sub-ring have their own RPL. The sub-ring interconnects to the major ring on nodes C and D and has a virtual channel on the major ring. The SR/ESS supports both virtual channel and non-virtual channel rings. Schematics of the physical and logical topologies are also shown in Figure 124.

Note: G.8032 has defined a Ring-ID value (1-255) in the G.8032 protocol. The SR/ESS implementation only uses a Ring-ID value of 1, which complies with G.8032v2. The configuration on a node uses a ring instance with a number but all rings use a Ring-ID of 1. This ring instance number is purely local and does not have to match on other ring nodes. Only the VLAN ID must match between SR/ESS ring nodes. For consistency in this example, VPLS instances and Ethernet ring instances are shown as matching for the same ring.

An RPL owner and RPL neighbor are configured for both the major ring and sub-ring. The path and associated link will be the RPL when the ring is fully operational and will be blocked by the RPL owner whenever there is no fault on other ring links. Each ring RPL is independent. If a different ring link fails then the RPL will be unblocked by the RPL owner. The link shared between a sub-ring and the major ring is completely controlled by the major ring as if the sub-ring were not there. Each ring can completely protect one fault within its ring. When the failed link recovers, it will initially be blocked by one of its adjacent nodes. The adjacent node sends an R-APS message across the ring to indicate the error is cleared and after a configurable time, if reversion is enabled, the RPL will revert to being blocked with all other links unblocked. This ensures that the ring topology when fully operational is predictable.

If a specific RPL owner is not configured (not recommended by G.8032 specification), then the last link to become active will be blocked and the ring will remain in this state until another link fails. This operation makes the selection of the blocked link non-deterministic and is not recommended.

The protection protocol uses a specific control VLAN, with the associated data VLANs taking their forwarding state from the control VLAN. The control VLAN cannot carry data.

Load Balancing with Multiple Ring Instances

Each control ring is independent of the other control rings on the same topology. Therefore since the RPL is not used by one control ring it is often desirable to set up a second control ring that uses a different link as RPL. This spreads out traffic in the topology but if there is a link failure in the ring all traffic will be on the remaining links. In the examples below only a single control ring instance is configured. Other control and data rings could be configured if desired.

Provider Backbone (PBB) Support

PBB services also support G.8032 as data services (the services used for the control VPLS must be a regular VPLS). B/I-VPLS rings support both major rings and sub-rings. B-VPLS rings support MC-LAG as a dual homing option when aggregating I-VPLS traffic onto a B-VPLS ring. In other words, I-VPLS rings should not be dual-homed into two BEB (Backbone Edge Bridge) nodes where the B-VPLS uses G.8032 to get connected to the rest of the B-VPLS network as the only mechanism which can propagate MAC flushes between an I-VPLS and B-VPLS is an LDP MAC flush.

SR/ESS Implementation

G.8032 is built from VPLS components and each ring consists of the configuration components illustrated in [Figure 125](#).

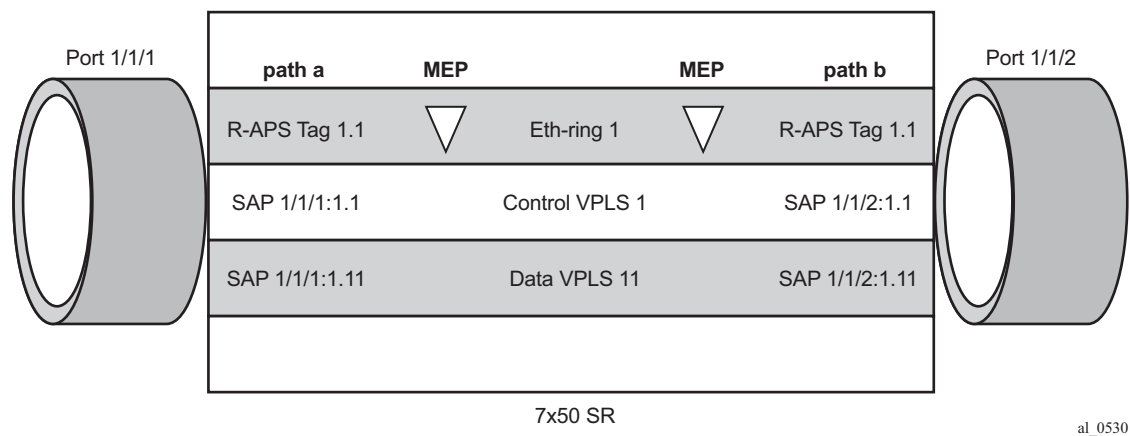
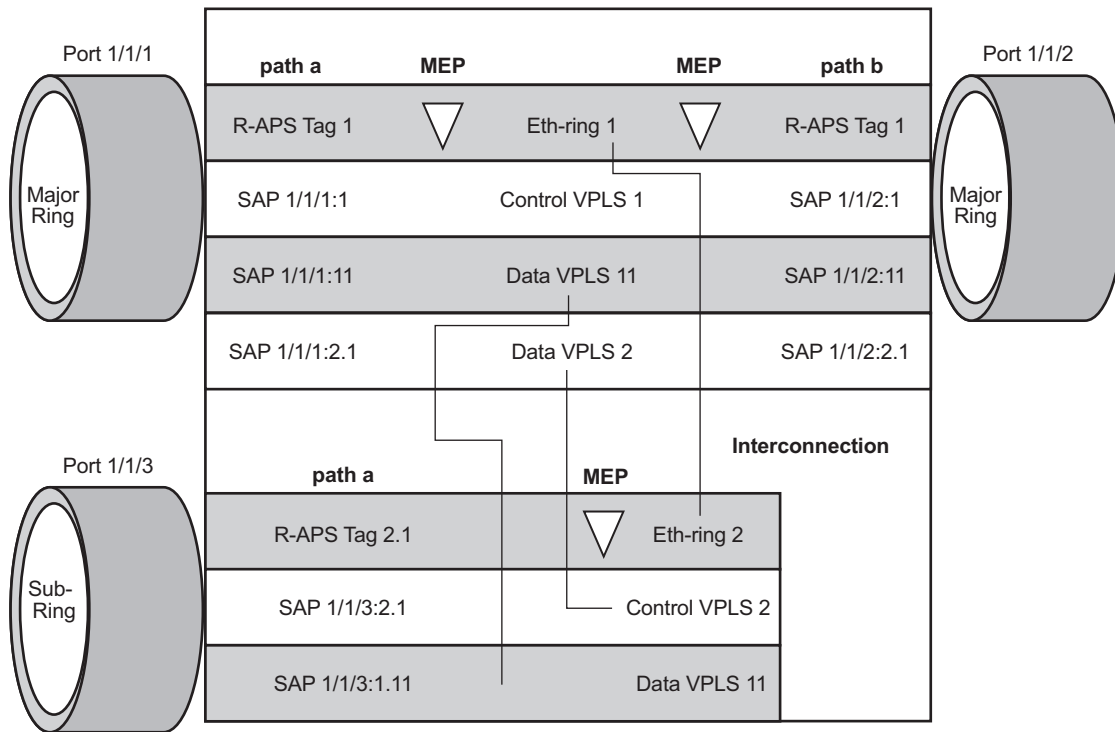


Figure 125: G.8032 Ring Components

These components consist of:

- The Eth-ring instance which defines the R-APS tags, the MEPs and the ring behavior.
- The control VPLS which has the SAPs that match the R-APS.
- The data VPLS which is linked to the ring. All of the data VPLS SAPs follow the operational state of the control VPLS SAPs in that each blocked SAP controlled by the ring is blocked for all control and data instances.

Figure 126 depicts the major ring and sub-ring interconnection components:



al_0531

Figure 126: G.8032 Sub-Ring Interconnection Components

For a sub-ring, the configuration is the same as a single ring except at the junction of the major ring and the sub-ring. The interconnection of a sub-ring and a major ring links the control VPLS of the sub-ring to a data VPLS of the major ring when a virtual link is used. Similarly the data VPLS of the sub-ring is linked to a data VPLS of the major ring. G.8032 Sub-Ring Interconnection Components illustrates the relationship of a sub-ring and a major ring. Since this sub-ring has a virtual channel, the data VPLS 2 has both data SAPs from the sub-ring and data SAPs from the major ring. The virtual channel is also optional and in non-virtual-link cases, no VPLS instance is required (see non-virtual-link in the section [Configuration of a Sub-Ring to a VPLS Service \(with a Non-Virtual Link\)](#) on page 791).

In [Figure 126](#), the inner tag values are kept the same for clarity but in fact any encapsulation that is consistent with the next ring link will work. In other words, ring SAPs can perform VLAN ID translation and even when connecting a sub-ring to a major ring. This also means that other ports may reuse the same tags when connecting independent services.

The R-APS tags (ring automatic protection switching tags) and SAPs on the rings can either be dot1q or QinQ encapsulated. It is also possible to have the control VPLS using single tagged frames with the data VPLSs using double tagged frames; this requires the system to be configured with the new-qinq-untagged-sap parameter (configure system ethernet new-qinq-untagged-sap), with the ring path raps-tags and control VPLS SAPs configured as qtag.0, and the data VPLSs configured as QinQ SAPs. Note that STP cannot be enabled on SAPs connected to eth-rings.

R-APS messages received from other nodes are normally blocked on the RPL interface but the sub-ring case with non-virtual channel recommends that R-APS messages be propagated over the RPL. Configuring sub-ring non-virtual-link on all nodes on the sub-ring propagation of R-APS messages is mandatory in order to achieve this.

R-APS messages are forwarded out of the egress using forwarding class NC, this should be prioritized accordingly in the SAP egress QoS policy to ensure that congestion does not cause R-APS messages to be dropped which could cause the ring to switch to another path.

Configuration

This section describes the configuration of multiple rings. The eth-ring configuration commands are shown below.

```
configure eth-ring <ring-index>
    ccm-hold-time { [down <down-timeout>] [up <up-timeout>] }
    compatible-version {1|2}
    description <description-string>
    guard-time <time>
    node-id <xx:xx:xx:xx:xx:xx or xx-xx-xx-xx-xx-xx>
    path {a|b} [ {<port-id>|<lag-id>}] raps-tag <qtag[.<qtag>] ]
        description <description-string>
        eth-cfm
            mep <mep-id> domain <md-index> association <ma-index>
            ...
        rpl-end
        shutdown
    revert-time <time>
    rpl-node {owner|nbr}
    shutdown
    sub-ring {virtual-link|non-virtual-link}
        interconnect [ring-id <ring-index>|vpls]
        propagate-topology-change
```

Parameters:

- <ring-index> — This is the number by which the ring is referenced, values: 1 to 128.
- ccm-hold-time { [down <down-timeout>] [up <up-timeout>] }
 - **down** — This command specifies the timer which controls the delay between detecting that ring path is down and reporting it to the G.8032 protection module. If a non-zero value is configured, the system will wait for the time specified in the value parameter before reporting it to the G.8032 protection module. Note that this parameter applies only to ring path CCM. It does not apply to the ring port link state. To dampen ring port link state transitions, use the hold-time parameter from the physical member port. This is useful if the underlying path between two nodes is going across an optical system which implements its own protection.
 - **up** — This command specifies the timer which controls the delay between detecting that ring path is up and reporting it to the G.8032 protection module. If a non-zero value is configured, the system will wait for the time specified in the value parameter before reporting it to the G.8032 protection module. Note that this parameter applies only to ring path CCM. It does not apply to the member port link state. To dampen member port link state transitions, use the hold-time parameter from the physical member port.

Values:

```
<down-timeout> : [0..5000] in centiseconds - Default: 0
<up-timeout>   : [0..5000] in deciseconds - Default: 20
```

1 centisecond = 10ms

1 decisecond = 100ms

- **compatible-version** — This command configures eth-ring compatibility version for the G.8032 state machine and messages. The default is version 2 (ITU G.8032v2) and all 7x50 systems use version 2. If there is a need to interwork with third party devices that only support version 1, this can be set to version 1 allowing the reception of version 1 PDUs. Note that version 2 is encoded as 1 in the R-APS messages. Compatibility allows the reception of version 1 (encoded as 0) R-APS PDUs but, as per the G.8032 specification, higher versions are ignored on reception. For the SR/ESS, messages are always originated with version 2. Therefore if a third party switch supported version 3 (encoded as 2) or higher interworking is also supported provided the other switch is compatible with version 2.
- **description** *<description-string>* — This configures a text string, up to 80 characters, which can be used to describe the use of the eth-ring.
- **guard-time** *<time>* — The forwarding method, in which R-APS messages are copied and forwarded at every Ethernet ring node, can result in a message corresponding to an old request, that is no longer relevant, being received by Ethernet ring nodes. Reception of an old R-APS message may result in erroneous ring state interpretation by some Ethernet ring nodes. The guard timer is used to prevent Ethernet ring nodes from acting upon outdated R-APS messages and prevents the possibility of forming a closed loop. Messages are not forwarded when the guard-timer is running.
Values: [1..20] in deciseconds - Default: 5
1 decisecond = 100ms
- **node-id** *<xx:xx:xx:xx:xx:xx>* or *<xx-xx-xx-xx-xx-xx>* — This allows the node identifier to be explicitly configured. By default the chassis MAC is used. Not required in typical configurations.
- **path** {**a**|**b**} [{*<port-id>*|*<lag-id>*} **raps-tag** *<qtag[.<qtag>]>*] — This parameter defines the paths around the ring, of which there are two in different directions on the ring: an “a” path and a “b” path, except on the interconnection node where a sub-ring connects to another major/sub ring in which case there is one path (either a or b) configured together with the **sub-ring** command. The paths are configured on a dot1q or QinQ encapsulated access or hybrid port or a LAG with the encapsulation used for the R-APS messages on the ring. These can be either single or double tagged.
 - **description** *<description-string>* — This configures a text string, up to 80 characters, which can be used to describe the use of the path.
 - **eth-cfm** — Configures the associated Ethernet CFM parameters.
 - **mep** *<mep-id>* **domain** *<md-index>* **association** *<ma-index>* — The MEP defined under the path is used for the G.8032 protocol messages, which are based on IEEE 802.1ag/Y.1731 CFM frames.
 - **rpl-end** — When configured, this path is expected to be one end of the RPL. This parameter must be configured in conjunction with the *rpl-node*.
 - **shutdown** — This command shuts down the path.

- **revert-time** *<time>* — This command configures the revert time for an Eth-Ring. Revert time is the time that the RPL will wait before returning to the blocked state. Configuring **no revert-time** disables reversion, effectively setting the revert-time to zero. Values: [60..720] in seconds - Default: 300
- **rpl-node** {**owner**|**nbr**} — A node can be designated as either the **owner** of the RPL, in which case this node is responsible for the RPL, or the **nbr**, in which case this node is expected to be the neighbor to the RPL owner across the RPL. The **nbr** is optional and is included to be compliant with the specification. This parameter must be configured in conjunction with the **rpl-end** command. On a sub-ring without virtual channel it is mandatory to configure **sub-ring non-virtual-link** on all nodes on the sub-ring to ensure propagation of the R-APS messages around the sub-ring.
- **shutdown** — This command shuts down the ring.
- **sub-ring** {**virtual-link**|**non-virtual-link**} — This command is configured on the interconnection node between the sub-ring and its major/sub ring to indicate that this ring is a sub-ring. The parameter specifies whether it uses a virtual link through the major/sub ring for the R-APS messages or not. A ring configured as a sub-ring can only be configured with a single path.
 - **interconnect** [**ring-id** *<ring-index>*]**vpls**] — A sub-ring connects to either another ring or a VPLS service. If it connects to another ring (either a major ring or another sub-ring), the ring identifier must be specified and the ring to which it connects must be configured with both a path “a” and a path “b”, meaning that it is not possible to connect a sub-ring to another sub-ring on an interconnection node. Alternatively, the **vpls** parameter is used to indicate the sub-ring connects to a VPLS service. Interconnection using a VPLS service requires the sub-ring to be configured with **non-virtual-link**.
 - **propagate-topology-change** — If a topology change event happens in the sub-ring, it can be optionally propagated with the use of this parameter to either the major/sub-ring it is connected to, using R-APS messages, or to the LDP VPLS SDP peers using an LDP “flush-all-from-me” message if the sub-ring is connected to a VPLS service.

The operation and configuration of G.8032 with multiple rings is described below based on the topology shown in [Figure 127](#).

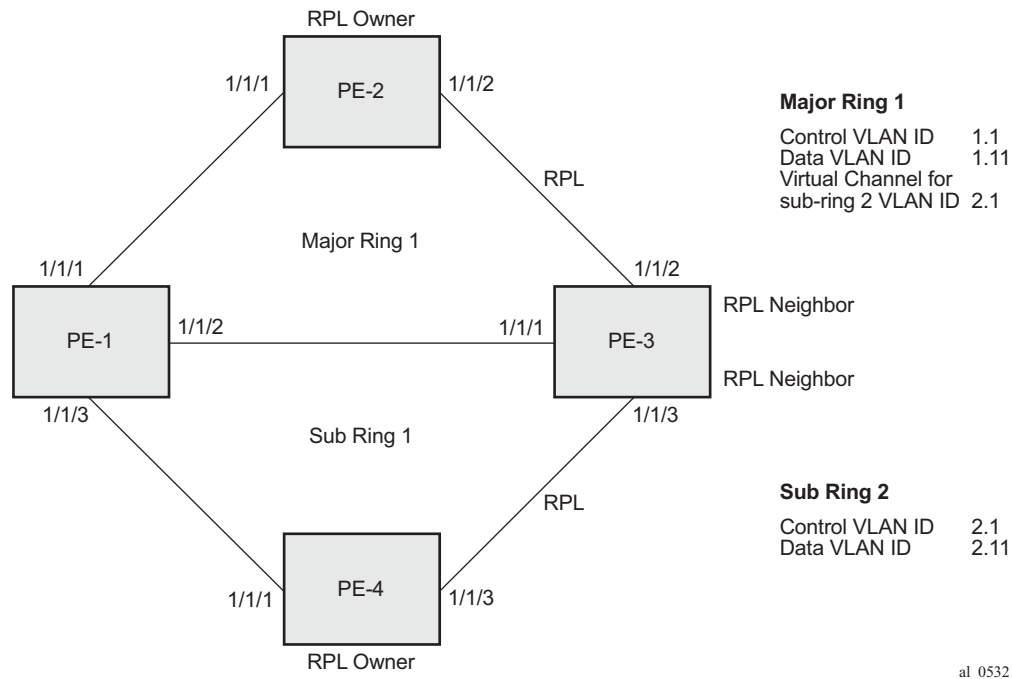


Figure 127: Ethernet Test Topology

The configuration is divided into the following sections:

- A sub-ring connected to a major ring using a virtual link through the major ring.
- A sub-ring connected to a major ring without a virtual link.
- A sub-ring connected to a VPLS service (without a virtual link).

Configuration of a Sub-Ring to a Major Ring with a Virtual Link

To configure an Ethernet ring using R-APS, there will be at least 2 VPLS services required for one Eth-Ring instance, one for the control channel and the other (or more) for data channel(s). The control channel is used for R-APS signaling while the data channel is for user data traffic. The state of the data channels is inherited from the state of the control channel.

Step 1. Configuring the encapsulation for each ring port.

Eth-Ring needs an R-APS tag to send/receive G.8032 signaling messages. To configure a control channel, an access SAP configuration is required on each path a/b port. The SAP configuration follows that of the port and must be either dot1q or QinQ, consequently the control and data packets are either single tagged or double tagged. Note that single tagged control frames are supported on a QinQ port by configuring the system with the `new-qinq-untagged-sap` parameter (configure system ethernet new-qinq-untagged-sap), and the ring path raps-tags and control VPLS SAPs configured as `qtag.0`.

In this example QinQ tags are used. The commands for the major and sub rings ring, on PE-1 for example, are:

```
*A:PE-1# configure port 1/1/1 ethernet mode access
*A:PE-1# configure port 1/1/2 ethernet mode access
*A:PE-1# configure port 1/1/3 ethernet mode access
*A:PE-1# configure port 1/1/1 ethernet encap-type qinq
*A:PE-1# configure port 1/1/2 ethernet encap-type qinq
*A:PE-1# configure port 1/1/3 ethernet encap-type qinq
```

Step 2. Configuring ETH-CFM.

Configuring ETH-CFM domain, association and MEP is required before configuring Ethernet ring. The standard domain format is none and the association name should be icc-based (Y.1731), however, the SR/ESS implementation is flexible in that it supports both IEEE and ICC formats. The *eth-ring* MEP requires sub-second CCM interval (10ms or 100ms) to be configured (or 1 second from 11.0.R.1 or later).

Note that the MEPs used for R-APS control normally will have CCM configured on the control channel path MEPs for failure detection. Alternatively, detecting a failure of the ring may be achieved by running Ethernet in the First Mile (EFM) at the port level if CCM is not possible at 100ms or 10ms (or 1 second as of release 11.0.R1). Also rings can be run without CFM although the ETH-CFM association must be configured for R-APS messages to be exchanged. To omit the failure detecting CCMs, it would be necessary to remove the *ccm-enable* from under the path MEPs and to remove the *remote-mepid* on the corresponding ETH CFM configuration.

Loss-of-signal, in conjunction with other OAM mechanisms, is applicable only when the nodes are directly connected.

Configuration of a Sub-Ring to a Major Ring with a Virtual Link

Figure 128 shows the details of the MEPs and their associations configured when both the major and sub rings are used. Note the associations only need to be pair wise unique but for clarity five unique associations are used. Also, any name format can be used but it must be consistent on both adjacent nodes.

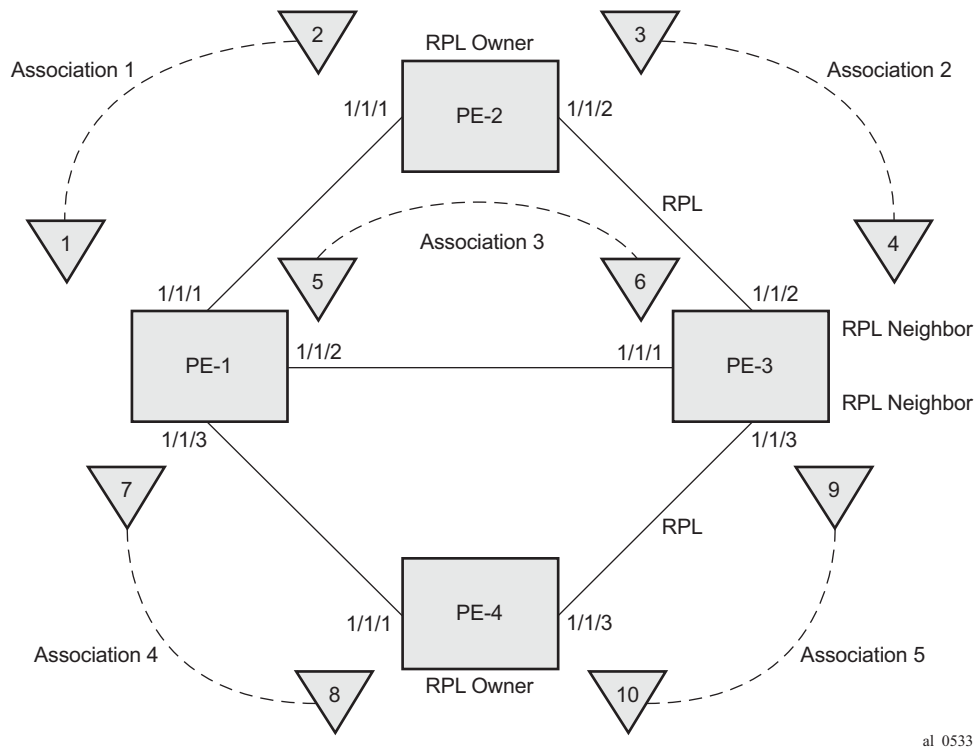


Figure 128: ETH-CFM MEP Associations

The configuration of ETH-CFM for the major and sub rings on each node is as follows. Note that the CCMs for failure detection are configured for 1 second intervals.

Ring node PE-1: Association 1 and 3 are used for the major ring and association 4 is used for the sub-ring.

```
*A:PE-1>config>eth-cfm# info
```

```
-----
domain 1 format none level 2
  association 1 format icc-based name "10000000000001"
    ccm-interval 1
    remote-mepid 2
  exit
  association 3 format icc-based name "10000000000003"
    ccm-interval 1
    remote-mepid 6
  exit
```

```

        association 4 format icc-based name "10000000000004"
        ccm-interval 1
        remote-mepid 8
    exit
exit

```

Ring node PE-2: Association 1 and 2 are used for the major ring.

```

*A:PE-2>config>eth-cfm# info
-----
    domain 1 format none level 2
        association 1 format icc-based name "10000000000001"
        ccm-interval 1
        remote-mepid 1
    exit
        association 2 format icc-based name "10000000000002"
        ccm-interval 1
        remote-mepid 4
    exit
exit

```

Ring node PE-3: Association 2 and 3 are used for the major ring and association 5 is used for the sub-ring.

```

*A:PE-3>config>eth-cfm# info
-----
    domain 1 format none level 2
        association 2 format icc-based name "10000000000002"
        ccm-interval 1
        remote-mepid 3
    exit
        association 3 format icc-based name "10000000000003"
        ccm-interval 1
        remote-mepid 5
    exit
        association 5 format icc-based name "10000000000005"
        ccm-interval 1
        remote-mepid 10
    exit
exit

```

Ring node PE-4: Association 4 and 5 are used for the sub-ring.

```

*A:PE-4>config>eth-cfm# info
-----
    domain 1 format none level 2
        association 4 format icc-based name "10000000000004"
        ccm-interval 1
        remote-mepid 7
    exit
        association 5 format icc-based name "10000000000005"
        ccm-interval 1
        remote-mepid 9
    exit
exit

```

Configuration of a Sub-Ring to a Major Ring with a Virtual Link

Step 3. Configuring Eth-Ring – major ring 1.

Two paths must be configured to form a ring. In this example, VLAN tag 1.1 is used as control channel for R-APS signaling for the major ring (ring 1) on the ports shown in [Figure 127](#) using the ETH CFM information shown in [Figure 128](#). The revert-time is set to its minimum value and CCM messages are enabled on the MEP. The **control-mep** parameter is required to indicate that this MEP is used for ring R-APS messages.

Ring node PE-1:

```
*A:PE-1>config# eth-ring 1
*A:PE-1>config>eth-ring# info
-----
description "Ethernet Ring 1"
revert-time 60
path a 1/1/1 raps-tag 1.1
    description "Ethernet Ring : 1 Path : pathA"
    eth-cfm
        mep 1 domain 1 association 1
        ccm-enable
        control-mep
        no shutdown
    exit
exit
no shutdown
exit
path b 1/1/2 raps-tag 1.1
    description "Ethernet Ring : 1 Path : pathB"
    eth-cfm
        mep 5 domain 1 association 3
        ccm-enable
        control-mep
        no shutdown
    exit
exit
no shutdown
exit
no shutdown
```

It is mandatory to configure a MEP under the path otherwise this error will be displayed.

```
*A:PE-1>config>eth-ring>path# no shutdown
INFO: ERMGR #1001 Not permitted - must configure eth-cfm MEP first
```

Note that while MEPs are mandatory, enabling CCMs on the MEPs under the paths as a failure detection mechanism is optional as explained earlier.

Ring node PE-2: This is configured as the RPL owner with the RPL being on path “b” as indicated by the **rpl-end** parameter.

```
*A:PE-2# configure eth-ring 1
*A:PE-2>config>eth-ring# info
```

```

-----
description "Ethernet Ring 1"
revert-time 60
rpl-node owner
path a 1/1/1 raps-tag 1.1
    description "Ethernet Ring : 1 Path : pathA"
    eth-cfm
        mep 2 domain 1 association 1
        ccm-enable
        control-mep
        no shutdown
    exit
exit
no shutdown
exit
path b 1/1/2 raps-tag 1.1
    description "Ethernet Ring : 1 Path : pathB"
    rpl-end
    eth-cfm
        mep 3 domain 1 association 2
        ccm-enable
        control-mep
        no shutdown
    exit
exit
no shutdown
exit
no shutdown

```

It is not permitted to configure a path as an RPL end without having configured the node on this ring to be either the RPL *owner* or *nbr* otherwise the following error message is reported.

```

*A:PE-2>config>eth-ring>path# rpl-end
INFO: ERMGR #1001 Not permitted - path-type rpl-end is not consistent with eth-ring 'rpl-
node' type

```

Ring node PE-3: This is configured as the RPL neighbor with the RPL being on path “b” as indicated by the **rpl-end** parameter.

```

*A:PE-3>config>eth-ring# info
-----
description "Ethernet Ring 1"
revert-time 60
rpl-node nbr
path a 1/1/1 raps-tag 1.1
    description "Ethernet Ring : 1 Path : pathA"
    eth-cfm
        mep 6 domain 1 association 3
        ccm-enable
        control-mep
        no shutdown
    exit
exit
no shutdown
exit
path b 1/1/2 raps-tag 1.1
    description "Ethernet Ring : 1 Path : pathB"

```

Configuration of a Sub-Ring to a Major Ring with a Virtual Link

```

rpl-end
eth-cfm
    mep 4 domain 1 association 2
    ccm-enable
    control-mep
    no shutdown
    exit
exit
no shutdown
exit
no shutdown
```

The link between PE-2 and PE-3 will be the RPL with PE-2 and PE-3 blocking that link when the ring is fully operational. In this example the RPL is using the same path, namely path “b”, on both PE-2 and PE-3 but this is not mandatory.

Step 4. Configuring Eth-Ring – sub-ring 2.

Ring nodes PE-1, PE-3 and PE-4 have a sub-ring. The sub-ring attaches to the major ring (ring 1). The sub-ring in this case will use a virtual-link. The interconnection ring instance identifier (*ring-id*) is specified and *propagate-topology-change* indicates that sub-ring flushing will be propagated to the major ring. Only one path is specified since the other path is not required at an interconnection node. Sub-rings are almost identical to major rings in operation except that sub-rings send MAC flushes towards their connected ring (either a major or sub ring). Major or sub rings never send MAC flushes to their sub-rings. Therefore a couple of sub-rings connected to a major ring can cause MACs to flush on the major ring but the major ring will not propagate a sub-ring MAC flush to other sub-rings.

Ring node PE-1: This node provides an interconnection between the major ring (1) and the sub-ring (2). Ring 2 is configured to be a sub-ring which interconnects to ring 1. It will use a virtual link on ring 1 to send R-APS messages to the other interconnection node and topology changes will be propagated from the sub-ring 2 to the major ring 1.

```
*A:PE-1>config# eth-ring 2
*A:PE-1>config>eth-ring# info
-----
description "Ethernet Sub-ring 2 on Ring 1"
revert-time 60
sub-ring virtual-link
    interconnect ring-id 1
    propagate-topology-change
    exit
exit
path a 1/1/3 raps-tag 2.1
description "Ethernet Ring : 2 Path : pathA"
eth-cfm
    mep 7 domain 1 association 4
    ccm-enable
    control-mep
    no shutdown
    exit
exit
no shutdown
```

```
exit
no shutdown
```

Ring node PE-3: The configuration of PE-3 is similar to PE-1 but PE-3 is the RPL neighbor, with the RPL end on path “a”, for the RPL between PE-3 and PE-4.

```
*A:PE-3>config# eth-ring 2
*A:PE-3>config>eth-ring# info
-----
description "Ethernet Sub-ring 2 on Ring 1"
revert-time 60
rpl-node nbr
sub-ring virtual-link
    interconnect ring-id 1
    propagate-topology-change
exit
exit
path a 1/1/3 raps-tag 2.1
    description "Ethernet Ring : 2 Path : pathA"
    rpl-end
    eth-cfm
        mep 9 domain 1 association 5
        ccm-enable
        control-mep
        no shutdown
    exit
exit
no shutdown
exit
no shutdown
```

Ring node PE-4: This node only has configuration for the sub-ring, ring 2. It is also the RPL owner, with path “b” being the RPL end, for the RPL between PE-3 and PE-4.

```
*A:PE-4>config# eth-ring 2
*A:PE-4>config>eth-ring# info
-----
description "Ethernet Ring : 2"
revert-time 60
rpl-node owner
path a 1/1/1 raps-tag 2.1
    description "Ethernet Ring : 2 : pathA"
    eth-cfm
        mep 8 domain 1 association 4
        ccm-enable
        control-mep
        no shutdown
    exit
exit
no shutdown
exit
path b 1/1/3 raps-tag 2.1
    description "Ethernet Ring : 2 : pathB"
    rpl-end
    eth-cfm
        mep 10 domain 1 association 5
        ccm-enable
```

Configuration of a Sub-Ring to a Major Ring with a Virtual Link

```
        control-mep
        no shutdown
    exit
    exit
    no shutdown
exit
no shutdown
```

Until the Ethernet Ring instance is attached to a VPLS service, the ring operational status is down and the forwarding status of each port is blocked. This prevents the operator from creating a loop by mis-configuration. This state can be seen on ring node PE-1 as follows:

```
*A:PE-1# show eth-ring 1
=====
Ethernet Ring 1 Information
=====
Description          : Ethernet Ring 1
Admin State          : Up                Oper State          : Down
Node ID              : d8:30:ff:00:00:00
Guard Time           : 5 deciseconds    RPL Node             : rplNone
Max Revert Time      : 60 seconds        Time to Revert       : N/A
CCM Hold Down Time   : 0 centiseconds    CCM Hold Up Time    : 20 deciseconds
Compatible Version    : 2
APS Tx PDU           : Request State: 0xB
                     Sub-Code          : 0x0
                     Status             : 0x20 ( BPR )
                     Node ID            : d8:30:ff:00:00:00
Defect Status         :
Sub-Ring Type         : none
-----
Ethernet Ring Path Summary
-----
Path Port    Raps-Tag    Admin/Oper    Type    Fwd State
-----
a 1/1/1      1.1            Up/Down      normal  blocked
b 1/1/2      1.1            Up/Down      normal  blocked
=====
```


Step 5. Adding Eth-Ring SAP to the control channel service

Path “a” and “b” configured in the eth-ring must be added as SAPs into a VPLS service (standard VPLS) using the **eth-ring** parameter. The SAP encapsulation values must match the values of the *raps-tag* configured for the associated path.

G.8032 uses the same raps-tag value on all nodes on the ring, as configured in this example. However, the SR/ESS implementation relaxes this constraint by requiring the tag to match only on adjacent nodes.

A VPLS service (identifier 1) is configured on PE-1, PE-2 and PE-3 for the control channel for the major ring (ring1), and another VPLS service (identifier 2) is used on PE-1, PE-3 and PE-4 for the sub-ring (ring 2).

Ring node PE-1: Control service for the major ring.

```
*A:PE-1>config>service# vpls 1
*A:PE-1>config>service>vpls# info
-----
description "Control VID 1.1 for Ring 1 Major Ring"
stp
    shutdown
exit
sap 1/1/1:1.1 eth-ring 1 create
    stp
        shutdown
    exit
exit
sap 1/1/2:1.1 eth-ring 1 create
    stp
        shutdown
    exit
exit
no shutdown
```

Ring node PE-2: Control service for the major ring.

```
*A:PE-2>config>service# vpls 1
*A:PE-2>config>service>vpls# info
-----
description "Control VID 1.1 for Ring 1 Major Ring"
stp
    shutdown
exit
sap 1/1/1:1.1 eth-ring 1 create
    stp
        shutdown
    exit
exit
sap 1/1/2:1.1 eth-ring 1 create
    stp
        shutdown
    exit
```

Configuration of a Sub-Ring to a Major Ring with a Virtual Link

```
exit
no shutdown
```

Ring node PE-3: Control service for the major ring.

```
*A:PE-3>config>service# vpls 1
*A:PE-3>config>service>vpls# info
-----
description "Control VID 1.1 for Ring 1 Major Ring"
stp
    shutdown
exit
sap 1/1/1:1.1 eth-ring 1 create
    stp
        shutdown
    exit
exit
sap 1/1/2:1.1 eth-ring 1 create
    stp
        shutdown
    exit
exit
no shutdown
```

Note that you cannot add a normal SAP or SDP in a control channel VPLS, only SAPs with an **eth-ring** parameter can be added. Trying to add a SAP without this parameter into a control channel VPLS will result in the message below being displayed.

```
*A:PE-1>config>service>vpls# sap 1/1/7:1.1 create
MINOR: SVCMGR #1321 Service contains an Ethernet ring control SAP
```

For the sub-ring, the configuration of a split horizon group for the virtual channel on the major ring on the interconnection nodes is recommended. This avoids the looping of control R-APS messages in the case there is a mis-configuration in the major ring.

Ring node PE-1: Control service for the sub-ring. Notice that the first two SAPs connect to the major ring (ring1), these being for the virtual channel, and the third SAP connects to the sub-ring (ring 2).

```
*A:PE-1>config>service# vpls 2
*A:PE-1>config>service>vpls# info
-----
description "Control/Virtual Channel VID 2.1 for Ring 2"
split-horizon-group "shg-ring2" create
exit
stp
    shutdown
exit
sap 1/1/1:2.1 split-horizon-group "shg-ring2" eth-ring 1 create
description "Ring 2 Interconnection using Ring 1"
    stp
        shutdown
```

```

        exit
    exit
    sap 1/1/2:2.1 split-horizon-group "shg-ring2" eth-ring 1 create
        description "Ring 2 Interconnection using Ring 1"
        stp
            shutdown
        exit
    exit
    sap 1/1/3:2.1 eth-ring 2 create
        stp
            shutdown
        exit
    exit
    no shutdown

```

Ring node PE-2: Control service for the sub-ring. Sub-ring 2 is not present on PE-2, however, its virtual channel on major ring 1 needs to exist throughout ring 1.

```

*A:PE-2>config>service>vpls# info
-----
        description "Virtual Channel VID 2.1 for Ring 2"
        stp
            shutdown
        exit
    sap 1/1/1:2.1 eth-ring 1 create
        stp
            shutdown
        exit
    exit
    sap 1/1/2:2.1 eth-ring 1 create
        stp
            shutdown
        exit
    exit
    no shutdown

```

Note: If multiple virtual channels are used (due to the aggregation of multiple sub-rings into the same major ring), their configuration could be simplified on non-interconnection nodes on the major ring. To achieve this on a ring node such as PE-2, a default SAP could be used rather than configuring a VPLS per virtual channel. If QinQ SAPs are used then a default SAP of 1/1/[1,2]:qtag.* could be used but requires all control channels for sub-rings to be using qtag as the outer VLAN ID, or 1/1/[1,2]:* if dot1q SAPs were used. This is because the SAPs match explicit SAPs definitions first and the default SAP will handle any other traffic.

Ring node PE-3: Control service for the sub-ring. This is similar to the configuration of PE-1.

```

*A:PE-3>config>service# vpls 2
*A:PE-3>config>service>vpls# info
-----
        description "Control/Virtual Channel VID 2.1 for Ring 2"
        split-horizon-group "shg-ring2" create
        exit
        stp

```

Configuration of a Sub-Ring to a Major Ring with a Virtual Link

```
        shutdown
    exit
    sap 1/1/1:2.1 split-horizon-group "shg-ring2" eth-ring 1 create
    stp
        shutdown
    exit
    exit
    sap 1/1/2:2.1 split-horizon-group "shg-ring2" eth-ring 1 create
    stp
        shutdown
    exit
    exit
    sap 1/1/3:2.1 eth-ring 2 create
    stp
        shutdown
    exit
    exit
    no shutdown
```

Ring node PE-4: Control service for the sub-ring. Both SAPs are configured on the sub-ring (ring 2).

```
*A:PE-4>config>service# vpls 2
*A:PE-4>config>service>vpls# info
-----
    description "Control VID 2.1 for Ring 2 Sub-ring"
    stp
        shutdown
    exit
    sap 1/1/1:2.1 eth-ring 2 create
    stp
        shutdown
    exit
    exit
    sap 1/1/3:2.1 eth-ring 2 create
    stp
        shutdown
    exit
    exit
    no shutdown
```

At this point, the Eth-Ring is operationally up and the RPL is blocking successfully on ring node PE-2 port 1/1/2, as expected for the RPL owner/end configuration and on port 1/1/3 on PE-3 as the RPL neighbor.

Show Output

An overview of all of the ring(s) can be shown using the following commands, in this case on PE-1.

First, the ETH ring status is shown.

```
*A:PE-1# show eth-ring status
=====
Ethernet Ring (Status information)
=====
```

Ring ID	Admin State	Oper State	Path Information		State	MEP Information		
			Path	Tag		Ctrl-MEP	CC-Intvl	Defects
1	Up	Up	a - 1/1/1	1.1	Up	Yes	1	----
			b - 1/1/2	1.1	Up	Yes	1	----
2	Up	Up	a - 1/1/3	2.1	Up	Yes	1	----
			b - N/A		-	-	-	----

```
=====
Ethernet Tunnel MEP Defect Legend:
R = Rdi, M = MacStatus, C = RemoteCCM, E = ErrorCCM, X = XconCCM
*A:PE-1#
```

It is expected that the state is “up”, even on ring paths which are blocked. The “Defects” column refers to the CFM defects of the MEPs. If there is a problem, these will be flagged.

This output shows the ring and path forwarding states.

```
*A:PE-1# show eth-ring
=====
Ethernet Rings (summary)
=====
```

Ring ID	Int ID	Admin State	Oper State	Paths Summary				Path States	
				a	b	c	d	a	b
1	-	Up	Up	a - 1/1/1	1.1	b - 1/1/2	1.1	U	U
2	1	Up	Up	a - 1/1/3	2.1	b - Not configured		U	-

```
=====
Ethernet Ring Summary Legend:  B - Blocked    U - Unblocked
*A:PE-1#
```

Configuration of a Sub-Ring to a Major Ring with a Virtual Link

The specific ring information can be shown as follows.

Ring node PE-1:

```
*A:PE-1# show eth-ring 1
=====
Ethernet Ring 1 Information
=====
Description          : Ethernet Ring 1
Admin State          : Up                Oper State          : Up
Node ID              : d8:30:ff:00:00:00
Guard Time           : 5 deciseconds    RPL Node             : rplNone
Max Revert Time      : 60 seconds        Time to Revert       : N/A
CCM Hold Down Time   : 0 centiseconds    CCM Hold Up Time    : 20 deciseconds
Compatible Version    : 2
APS Tx PDU           : N/A
Defect Status        :

Sub-Ring Type        : none
-----
Ethernet Ring Path Summary
-----
Path Port    Raps-Tag    Admin/Oper    Type          Fwd State
-----
a  1/1/1      1.1           Up/Up         normal        unblocked
b  1/1/2      1.1           Up/Up         normal        unblocked
=====
*A:PE-1#
```

The status around the major ring can also be checked.

Ring node PE-2: Major ring.

```
*A:PE-2# show eth-ring 1
=====
Ethernet Ring 1 Information
=====
Description          : Ethernet Ring 1
Admin State          : Up                Oper State          : Up
Node ID              : d8:31:ff:00:00:00
Guard Time           : 5 deciseconds    RPL Node             : rplOwner
Max Revert Time      : 60 seconds        Time to Revert       : N/A
CCM Hold Down Time   : 0 centiseconds    CCM Hold Up Time    : 20 deciseconds
Compatible Version    : 2
APS Tx PDU           : Request State: 0x0
                      Sub-Code         : 0x0
                      Status           : 0xA0 ( RB BPR )
                      Node ID          : d8:31:ff:00:00:00
Defect Status        :

Sub-Ring Type        : none
-----
Ethernet Ring Path Summary
-----
Path Port    Raps-Tag    Admin/Oper    Type          Fwd State
-----
```

```

-----
a 1/1/1 1.1 Up/Up normal unblocked
b 1/1/2 1.1 Up/Up rplEnd blocked
=====
*A:PE-2#

```

Note that PE-2 is the RPL owner with port 1/1/2 as an RPL end, which is blocked as expected. The *revert-time* is also shown to be the configured value. Detailed information is shown relating to the R-APS PDUs being transmitted on this ring as this node is the RPL owner.

When a revert is pending, the “Time to Revert” will show the number of seconds remaining before the revert occurs.

Ring node PE-3: Major ring.

```

*A:PE-3# show eth-ring 1
=====
Ethernet Ring 1 Information
=====
Description      : Ethernet Ring 1
Admin State      : Up           Oper State       : Up
Node ID          : d8:32:ff:00:00:00
Guard Time       : 5 deciseconds RPL Node           : rplNeighbor
Max Revert Time  : 60 seconds   Time to Revert    : N/A
CCM Hold Down Time : 0 centiseconds CCM Hold Up Time : 20 deciseconds
Compatible Version : 2
APS Tx PDU       : N/A
Defect Status     :
Sub-Ring Type     : none
-----
Ethernet Ring Path Summary
-----
Path Port      Raps-Tag  Admin/Oper  Type      Fwd State
-----
a 1/1/1 1.1 Up/Up normal unblocked
b 1/1/2 1.1 Up/Up rplEnd blocked
=====
*A:PE-3#

```

PE-3 is the RPL neighbor with port 1/1/2 as an RPL end which is blocked as expected.

The information for the sub-ring can also be shown using the same command.

Ring node PE-1: Sub-ring.

```

*A:PE-1# show eth-ring 2
=====
Ethernet Ring 2 Information
=====
Description      : Ethernet Sub-ring 2 on Ring 1
Admin State      : Up           Oper State       : Up
Node ID          : d8:30:ff:00:00:00
Guard Time       : 5 deciseconds RPL Node           : rplNone
Max Revert Time  : 60 seconds   Time to Revert    : N/A

```

Configuration of a Sub-Ring to a Major Ring with a Virtual Link

```
CCM Hold Down Time :    0 centiseconds  CCM Hold Up Time :   20 deciseconds
Compatible Version : 2
APS Tx PDU          : N/A
Defect Status       :
```

```
Sub-Ring Type       : virtualLink          Interconnect-ID : 1
Topology Change     : Propagate
```

Ethernet Ring Path Summary

Path	Port	Raps-Tag	Admin/Oper	Type	Fwd State
a	1/1/3	2.1	Up/Up	normal	unblocked
b	-	-	-/-	-	-

```
=====
*A:PE-1#
```

Note that only path “a” is active and unblocked. The second path, path “b” is not configured as only one path is required on an interconnection node. The “Sub-Ring Type” is shown to be a virtual link interconnecting to ring 1, with topology propagation enabled.

Ring node PE-3: Sub-ring.

```
*A:PE-3# show eth-ring 2
```

Ethernet Ring 2 Information

```
=====
Description          : Ethernet Sub-ring 2 on Ring 1
Admin State          : Up                      Oper State       : Up
Node ID              : d8:32:ff:00:00:00
Guard Time           : 5 deciseconds          RPL Node            : rplNeighbor
Max Revert Time      : 60 seconds              Time to Revert       : N/A
CCM Hold Down Time   : 0 centiseconds          CCM Hold Up Time    : 20 deciseconds
Compatible Version    : 2
APS Tx PDU           : N/A
Defect Status        :
```

```
Sub-Ring Type       : virtualLink          Interconnect-ID : 1
Topology Change     : Propagate
```

Ethernet Ring Path Summary

Path	Port	Raps-Tag	Admin/Oper	Type	Fwd State
a	1/1/3	2.1	Up/Up	rplEnd	blocked
b	-	-	-/-	-	-

```
=====
*A:PE-3#
```

PE-3 is the RPL neighbor with port 1/1/3 as an RPL end, which is blocked as expected.

Ring Node PE-4: Sub-ring.

```
*A:PE-4# show eth-ring 2
```



```

Ethernet Ring 2 Information
=====
Description          : Ethernet Ring : 2
Admin State          : Up              Oper State           : Up
Node ID              : d8:33:ff:00:00:00
Guard Time           : 5 deciseconds   RPL Node             : rplOwner
Max Revert Time      : 60 seconds       Time to Revert        : N/A
CCM Hold Down Time   : 0 centiseconds   CCM Hold Up Time     : 20 deciseconds
Compatible Version    : 2
APS Tx PDU           : Request State: 0x0
                      Sub-Code          : 0x0
                      Status             : 0xA0 ( RB BPR )
                      Node ID            : d8:33:ff:00:00:00

Defect Status        :

Sub-Ring Type         : none
-----
Ethernet Ring Path Summary
-----
Path Port    Raps-Tag    Admin/Oper    Type          Fwd State
-----
a  1/1/1     2.1           Up/Up         normal        unblocked
b  1/1/3     2.1           Up/Up         rplEnd        blocked
=====
*A:PE-4#

```

PE-4 is the RPL owner with port 1/1/3 as an RPL end, which is blocked as expected.

The details of an individual path can be shown.

```

*A:PE-1# show eth-ring 1 path a
=====
Ethernet Ring 1 Path Information
=====
Description          : Ethernet Ring : 1 Path : pathA
Port                 : 1/1/1           Raps-Tag             : 1.1
Admin State          : Up              Oper State            : Up
Path Type            : normal           Fwd State             : unblocked
                                           Fwd State Change     : 09/19/2014 17:45:19

Last Switch Command: noCmd
APS Rx PDU           : Request State: 0x0
                      Sub-Code          : 0x0
                      Status             : 0xA0 ( RB BPR )
                      Node ID            : d8:31:ff:00:00:00
=====
*A:PE-1#

```

The ring hierarchy created can be shown, either for all rings, or as below for a specific ring.

```

*A:PE-1# show eth-ring 1 hierarchy
=====
Ethernet Ring 1 (hierarchy)
=====
Ring Int  Admin Oper          Paths Summary          Path States
ID   ID   State State                a      b
-----

```

Configuration of a Sub-Ring to a Major Ring with a Virtual Link

1	-	Up	Up	a - 1/1/1	1.1	b - 1/1/2	1.1	U	U
2	1	Up	Up	a - 1/1/3	2.1	b - Not configured		U	-

=====

Ethernet Ring Summary Legend: B - Blocked U - Unblocked

*A:PE-1#

Step 6. Configuring the user data channel VPLS service

The user data channels are created on a separate VPLS, VPLS 11 in this example, using VLAN tag 1.11. The ring data channels must be on the same ports as the corresponding control channels configured above. The access into the data services can use normal SAPs and/or SDPs, for example the SAP on port 1/1/9 below. Customer data traverses the ring on a data SAP. Multiple parallel data SAPs in different data services can be controlled by one control ring instance (eth-ring 1 in the example).

Ring node PE-1: The first two data SAPs correspond to the major ring 1, while the third SAP is the data SAP on the sub-ring 2.

```
*A:PE-1>config>service# vpls 11
*A:PE-1>config>service>vpls# info
-----
description "Data VPLS"
stp
    shutdown
exit
sap 1/1/1:1.11 eth-ring 1 create
    stp
        shutdown
    exit
exit
sap 1/1/2:1.11 eth-ring 1 create
    stp
        shutdown
    exit
exit
sap 1/1/3:1.11 eth-ring 2 create
    stp
        shutdown
    exit
exit
sap 1/1/9:1 create
    description "Sample Customer Service SAP"
exit
no shutdown
```

Ring node PE-3 (not shown) would be similar to ring node 1.

Ring node PE-2 is also similar with a single ring data service using VPLS 11 and tag 1.11.

```
*A:PE-2# configure service vpls 11
*A:PE-2>config>service>vpls# info
-----
description "Data VPLS"
stp
    shutdown
exit
sap 1/1/1:1.11 eth-ring 1 create
    stp
        shutdown
    exit
```

Configuration of a Sub-Ring to a Major Ring with a Virtual Link

```
exit
sap 1/1/2:1.11 eth-ring 1 create
    stp
        shutdown
    exit
exit
sap 1/1/9:1 create
    description "Sample Customer Service SAP"
exit
no shutdown
```

Ring node PE- 4: On ring node PE-4 the data VLAN ID is configured as a normal ring data VPLS on ring 2.

```
*A:PE-4# configure service vpls 11
*A:PE-4>config>service>vpls# info
-----
description "Data VPLS"
stp
    shutdown
exit
sap 1/1/1:1.11 eth-ring 2 create
    stp
        shutdown
    exit
exit
sap 1/1/3:1.11 eth-ring 2 create
    stp
        shutdown
    exit
exit
sap 1/1/9:1 create
    description "Sample Customer Service SAP"
exit
no shutdown
```

All of the SAPs which are configured to use Ethernet rings can be shown. The output below is taken from PE-1, where there are:

- two SAPs in VPLS 1 for the control channel of ring 1 (VLAN ID 1.1)
- two SAPs in VPLS 2 on ring 1 for the virtual channel for ring 2 (VLAN ID 2.1).
- one SAP in VPLS 2 on ring 2 for the control channel for ring 2 (VLAN ID 2.1)
- three SAPs in VPLS 11, two on ring 1 and one on ring 2, for the data service (VLAN ID 1.11). This matches the information in Figure 3.

```
*A:PE-1# show service sap-using eth-ring
=====
Service Access Points (Ethernet Ring)
=====
SapId          SvcId          Eth-Ring Path Admin Oper  Blocked Control/
                State State          Data
-----
1/1/1:1.1      1              1          a    Up    Up    No     Ctrl
```

```

1/1/2:1.1      1      1      b      Up      Up      No      Ctrl
1/1/1:2.1      2      1      a      Up      Up      No      Ctrl
1/1/2:2.1      2      1      b      Up      Up      No      Ctrl
1/1/3:2.1      2      2      a      Up      Up      No      Ctrl
1/1/1:1.11     11     1      a      Up      Up      No      Data
1/1/2:1.11     11     1      b      Up      Up      No      Data
1/1/3:2.11     11     2      a      Up      Up      No      Data
-----
Number of SAPs : 8
=====
*A:PE-1#

```

Statistics are available showing both the CCM and R-APS messages sent and received on a node. An associated **clear** command is available.

```

*A:PE-1# show eth-cfm statistics
=====
ETH-CFM System Statistics
=====
Rx Count      : 1201      Tx Count      : 1066
Dropped Congestion : 0      Discarded Error : 0
AIS Currently Act : 0      AIS Currently Fail : 0
=====
=====
ETH-CFM System Op-code Statistics
=====
Op-code      Rx Count      Tx Count
-----
ccm           1018          1018
...
raps           183           48
...
-----
Total          1201          1066
=====
*A:PE-1#

```

To see an example of the console messages on a ring failure, when the unblocked port (1/1/1) on PE-2 is shutdown the following messages are displayed.

```

*A:PE-2# configure port 1/1/1 shutdown
2 2014/09/19 18:03:01.00 PDT WARNING: SNMP #2004 Base 1/1/1
"Interface 1/1/1 is not operational"

3 2014/09/19 18:03:01.00 PDT MINOR: ERING #2001 Base eth-ring-1
"Eth-Ring 1 path 0 changed fwd state to blocked"

4 2014/09/19 18:03:01.00 PDT MINOR: ERING #2001 Base eth-ring-1
"Eth-Ring 1 path 1 changed fwd state to unblocked"
*A:PE-2#
5 2014/09/19 18:03:01.01 PDT MAJOR: SVCMMGR #2210 Base
"Processing of an access port state change event is finished and the status of a
11 affected SAPs on port 1/1/1 has been updated."

6 2014/09/19 18:03:04.33 PDT MINOR: ETH_CFM #2001 Base
"MEP 1/1/2 highest defect is now defRemoteCCM"

```

Configuration of a Sub-Ring to a Major Ring with a Virtual Link

*A:PE-2#

For troubleshooting, the **tools dump eth-ring** <ring-index> command displays path information, the internal state of the control protocol, related statistics information and up to the last 16 protocol events (including messages sent and received, and the expiration of timers). An associated **clear** parameter exists, which clears the event information in this output when the command is entered. The following is an example of the output on PE-1.

```
*A:PE-1# tools dump eth-ring 1
ringId 1 (Up/Up): numPaths 2 nodeId d8:30:ff:00:00:00
SubRing: none (interconnect ring 0, propagateTc No), Cnt 1
  path-a, port 1/1/1 (Up), tag 1.1(Dn) status (Up/Dn/Blk)
    cc (Dn/Up): Cnt 8/7 tm 001 00:27:12.220/001 00:09:23.340
    state: Cnt 9 B/F 001 00:27:12.220/001 00:09:25.410, flag: 0x0
  path-b, port 1/1/2 (Up), tag 1.1(Up) status (Up/Up/Fwd)
    cc (Dn/Up): Cnt 4/4 tm 001 00:07:09.200/001 00:09:23.920
    state: Cnt 8 B/F 001 00:07:09.200/001 00:09:26.410, flag: 0x0
FsmState= PROT, Rpl = None, revert = 60 s, guard = 5 ds
Defects =
Running Timers = PduReTx
lastTxPdu = 0xb000 Sf
path-a Normal, RxId(I)= d8:31:ff:00:00:00, rx(F)= v1-0x00a0 Nr, cmd= None
path-b Normal, RxId(I)= d8:31:ff:00:00:00, rx= v1-0xb000 Sf, cmd= None
DebugInfo: aPathSts 10, bPathSts 7, pm (set/cclr) 0/0, txFlush 2
RxRaps: ok 41 nok 0 self 4, TmrExp - wtr 0(0), grd 5, wtb 0
Flush: cnt 22 (8/14/0) tm 001 00:27:12.220-001 00:27:12.220 Out/Ack 0/1
RxRawRaps: aPath 17167 bPath 147 vPath 0
Now: 001 00:28:16.500 , softReset: No - noTx 0
```

Seq	Event	RxInfo(Path: NodeId-Bytes)	state:TxInfo (Bytes)	Dir	pA	pB	Time
===	=====	=====	=====	=====	=====	=====	=====
015	aDn		PROT : 0xb000 Sf	TxF->	Blk	Fwd	001 00:07:09.190
016	bDn		PROT : 0xb020 Sf	TxF->	Blk	Blk	001 00:07:09.200
017	pdu B:	d8:32:ff:00:00:00-0xb000 Sf	PROT : 0xb020 Sf	RxF<-	Blk	Blk	001 00:09:22.980
018	pdu A:	d8:31:ff:00:00:00-0xb000 Sf	PROT : 0xb020 Sf	RxF<-	Blk	Blk	001 00:09:24.410
019	pdu B:	d8:32:ff:00:00:00-0x0000 Nr	PROT : 0xb020 Sf	Rx<--	Blk	Blk	001 00:09:25.080
000	aUp		PROT : 0xb060 Sf(DNF)	Tx-->	Fwd	Blk	001 00:09:25.410
001	bUp		PEND-G: 0x0020 Nr	Tx-->	Fwd	Blk	001 00:09:25.910
002	pdu A:	d8:31:ff:00:00:00-0x0000 Nr	PEND-G: 0x0020 Nr	Rx<--	Fwd	Blk	001 00:09:26.210
003	pdu B:	d8:31:ff:00:00:00-0x0000 Nr	PEND-G: 0x0020 Nr	Rx<--	Fwd	Blk	001 00:09:26.210
004	pdu A:	d8:31:ff:00:00:00-0x0000 Nr	PEND-G: 0x0020 Nr	Rx<--	Fwd	Blk	001 00:09:26.310
005	pdu B:	d8:31:ff:00:00:00-0x0000 Nr	PEND-G: 0x0020 Nr	Rx<--	Fwd	Blk	001 00:09:26.310
006	pdu A:	d8:31:ff:00:00:00-0x0000 Nr	PEND : 0x0020 Nr	Rx<--	Fwd	Blk	001 00:09:26.410

G.8032 Ethernet Ring Protection Multiple Ring Topology

```

007 pdu
    PEND : ----- Fwd Fwd 001 00:09:26.410
008 pdu B: d8:31:ff:00:00:00-0x0000 Nr
    PEND : Rx<-- Fwd Fwd 001 00:09:26.410
009 pdu A: d8:31:ff:00:00:00-0x00a0 Nr(RB )
    PEND : RxF<- Fwd Fwd 001 00:10:41.410
010 pdu
    IDLE : ----- Fwd Fwd 001 00:10:41.410
011 pdu B: d8:31:ff:00:00:00-0x00a0 Nr(RB )
    IDLE : Rx<-- Fwd Fwd 001 00:10:41.410
012 pdu B: d8:31:ff:00:00:00-0xb000 Sf
    IDLE : RxF<- Fwd Fwd 001 00:27:09.120
013 pdu
    PROT : ----- Fwd Fwd 001 00:27:09.120
014 aDn
    PROT : 0xb000 Sf TxF-> Blk Fwd 001 00:27:12.220

*A:PE-1#

```

Configuration of a Sub-Ring to a Major Ring with a Non-Virtual Link

The differences from the above virtual link configuration with a non-virtual link for the sub-ring are:

- The sub-ring configuration on the interconnection nodes, PE-1 and PE-3, is modified to indicate that the sub-ring is not using a virtual link, otherwise it remains the same.
- The sub-ring configuration on the sub-ring node, PE-4, is also modified to indicate that this is part of a sub-ring that is not using a virtual link. This is mandatory on all non-interconnection nodes on the sub-ring in order to ensure the propagation of R-APS messages around the sub-ring.
- The virtual link services and SAPs must be removed from PE-1, PE-2 and PE3, that is:
 - On PE-1 and PE-3, the SAPs in VPLS 2 around the major ring (configured with the parameter *eth-ring 1*) are removed.
 - The service VPLS 2 is removed completely from PE-2.

The new configuration of sub-ring 2 on PE-1 is shown below, the configuration on PE-3 is similar.

```
*A:PE-1# configure eth-ring 2
*A:PE-1>config>eth-ring# info
-----
description "Ethernet Sub-ring 2 on Ring 1"
revert-time 60
sub-ring non-virtual-link
    interconnect ring-id 1
    propagate-topology-change
exit
exit
path a 1/1/3 raps-tag 2.1
    description "Ethernet Ring : 2 Path : pathA"
    eth-cfm
        mep 7 domain 1 association 4
        ccm-enable
        control-mep
        no shutdown
    exit
exit
no shutdown
exit
no shutdown
```


The configuration of sub-ring 2 on PE-4 is shown below, note the configuration of the sub-ring non-virtual-link.

```
*A:PE-4# configure eth-ring 2
*A:PE-4>config>eth-ring# info
-----
description "Ethernet Ring : 2"
revert-time 60
rpl-node owner
sub-ring non-virtual-link
exit
path a 1/1/1 raps-tag 2.1
  description "Ethernet Ring : 2 : pathA"
  eth-cfm
    mep 8 domain 1 association 4
    ccm-enable
    control-mep
    no shutdown
  exit
exit
no shutdown
exit
path b 1/1/3 raps-tag 2.1
  description "Ethernet Ring : 2 : pathB"
  rpl-end
  eth-cfm
    mep 10 domain 1 association 5
    ccm-enable
    control-mep
    no shutdown
  exit
exit
no shutdown
exit
no shutdown
```

The SAP usage on PE-1 can be seen below with only the control and data SAPs to PE-4 now using sub-ring 2.

```
*A:PE-1# show service sap-using eth-ring
=====
Service Access Points (Ethernet Ring)
=====
```

SapId	SvcId	Eth-Ring	Path	Admin State	Oper State	Blocked	Control/Data
1/1/1:1.1	1	1	a	Up	Up	No	Ctrl
1/1/2:1.1	1	1	b	Up	Up	No	Ctrl
1/1/3:2.1	2	2	a	Up	Up	No	Ctrl
1/1/1:1.11	11	1	a	Up	Up	No	Data
1/1/2:1.11	11	1	b	Up	Up	No	Data
1/1/3:1.11	11	2	a	Up	Up	No	Data

```
-----
Number of SAPs : 6
=====
*A:PE-1#
```

Configuration of a Sub-Ring to a Major Ring with a Non-Virtual Link

The information relating to sub-ring 2 is shown below and it can be seen that this is now not using a virtual link, but that sub-ring 2 is still connected to major ring 1 and propagation is still enabled from the sub-ring to the major ring. The single ring path (a) is unblocked as the RPL is configured between PE-3 and PE-4.

```
*A:PE-1# show eth-ring 2
=====
Ethernet Ring 2 Information
=====
Description          : Ethernet Sub-ring 2 on Ring 1
Admin State          : Up                               Oper State          : Up
Node ID              : d8:30:ff:00:00:00
Guard Time           : 5 deciseconds                    RPL Node             : rplNone
Max Revert Time      : 60 seconds                        Time to Revert       : N/A
CCM Hold Down Time   : 0 centiseconds                    CCM Hold Up Time    : 20 deciseconds
Compatible Version    : 2
APS Tx PDU           : N/A
Defect Status        :

Sub-Ring Type         : nonVirtualLink                    Interconnect-ID      : 1
Topology Change       : Propagate
-----
Ethernet Ring Path Summary
-----
Path Port    Raps-Tag    Admin/Oper    Type          Fwd State
-----
a 1/1/3      2.1           Up/Up         normal        unblocked
b -          -             -/-          -             -
=====
*A:PE-1#
```

Configuration of a Sub-Ring to a VPLS Service (with a Non-Virtual Link)

Sub-rings can be connected to VPLS services, in which case a virtual link is not used and is not configurable. While similar to the ring interconnect, there are a few differences.

Flush propagation is from the sub-ring to the VPLS, in the same way as it was for the sub-ring to the major ring. The same configuration parameter is used to propagate topology changes, note that in this case LDP flush messages (flush-all-from-me) are sent into the LDP portion of the network to account for ring changes without the need to configure anything in the VPLS service.

As with other rings, until an Ethernet ring instance is attached to the VPLS service, the ring operational status is down and the forwarding status of each port is blocked. This prevents operator from creating a loop by mis-configuration.

The topology for this case is shown in [Figure 129](#). The configuration is very similar to the sub-ring with a non-virtual link described above, but ring 1 is replaced by a VPLS service using LDP signaled mesh SDPs between PE-1, PE-2 and PE-3 to create a fully meshed VPLS service. Note that either spoke or mesh SDPs using LDP could be used for the VPLS, however, only mesh SDPs have been used in this example.

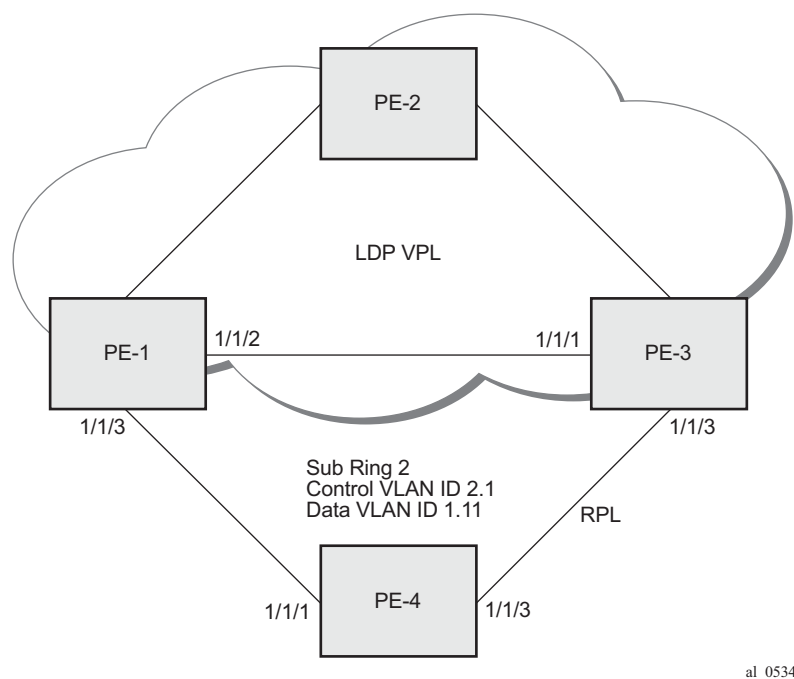


Figure 129: Sub-Ring to VPLS Topology

Configuration of a Sub-Ring to a VPLS Service (with a Non-Virtual Link)

The differences for the VPLS service connection to the configuration when the sub-ring is connected to a major ring without a virtual link are:

- The sub-ring configuration on the interconnection nodes, PE-1 and PE-3, is modified to indicate that the sub-ring is connected to a VPLS service.
- The sub-ring configuration on the sub-ring node, PE-4, is also modified to indicate that this is part of a sub-ring that is not using a virtual link. This is mandatory on all non-interconnection nodes on the sub-ring in order to ensure the propagation of R-APS messages around the sub-ring.
- The service (VPLS 1) and SAPs relating to the major ring 1 on PE-1, PE-2 and PE-3 are removed. These are replaced by routed IP interfaces configured with a routing protocol and LDP in order to signal the required MPLS labels, together with the necessary SDPs to provide interconnection at a service level.
- The data service (VPLS 11) is configured with mesh SDPs between PE-1, PE-2 and PE-3.

The configuration on PE-1 of the sub-ring 2 is as follows with the interconnect indicating a VPLS service. The configuration on PE-3 is similar.

```
*A:PE-1>config# eth-ring 2
*A:PE-1>config>eth-ring# info
-----
description "Ethernet Sub-ring 2 on Ring 1"
revert-time 60
sub-ring non-virtual-link
interconnect vpls
propagate-topology-change
exit
exit
path a 1/1/3 raps-tag 2.1
description "Ethernet Ring : 2 Path : pathA"
eth-cfm
mep 7 domain 1 association 4
ccm-enable
control-mep
no shutdown
exit
exit
no shutdown
exit
no shutdown
```

The configuration of sub-ring 2 on PE-4 is shown below, note the configuration of the sub-ring non-virtual-link.

```
*A:PE-4# configure eth-ring 2
*A:PE-4>config>eth-ring# info
-----
description "Ethernet Ring : 2"
revert-time 60
rpl-node owner
sub-ring non-virtual-link
```

```

exit
path a 1/1/1 raps-tag 2.1
  description "Ethernet Ring : 2 : pathA"
  eth-cfm
    mep 8 domain 1 association 4
    ccm-enable
    control-mep
    no shutdown
  exit
exit
no shutdown
exit
path b 1/1/3 raps-tag 2.1
  description "Ethernet Ring : 2 : pathB"
  rpl-end
  eth-cfm
    mep 10 domain 1 association 5
    ccm-enable
    control-mep
    no shutdown
  exit
exit
no shutdown
exit
no shutdown

```

The data service on PE-1 is shown below. The configuration on PE-3 is similar.

```

*A:PE-1>config>service# vpls 11
*A:PE-1>config>service>vpls# info
-----
  description "Data VPLS"
  stp
    shutdown
  exit
  sap 1/1/3:1.11 eth-ring 2 create
    stp
      shutdown
    exit
  exit
  sap 1/1/9:1 create
    description "Sample Customer Service SAP"
  exit
  mesh-sdp 2:11 create
    no shutdown
  exit
  mesh-sdp 3:11 create
    no shutdown
  exit
  no shutdown

```

The state of the sub-ring can be seen below and shows the sub-ring is not using a virtual link, is connected to a VPLS service and has propagation of topology change events enabled. As earlier, the single ring path (a) is unblocked as the RPL is configured between PE-3 and PE-4.

Configuration of a Sub-Ring to a VPLS Service (with a Non-Virtual Link)

```
*A:PE-1# show eth-ring 2
=====
Ethernet Ring 2 Information
=====
Description      : Ethernet Sub-ring 2 on Ring 1
Admin State      : Up                               Oper State      : Up
Node ID          : d8:30:ff:00:00:00
Guard Time       : 5 deciseconds                    RPL Node        : rplNone
Max Revert Time  : 60 seconds                        Time to Revert   : N/A
CCM Hold Down Time : 0 centiseconds                  CCM Hold Up Time : 20 deciseconds
Compatible Version : 2
APS Tx PDU       : N/A
Defect Status    :

Sub-Ring Type    : nonVirtualLink                    Interconnect-ID : VPLS
Topology Change  : Propagate
-----
Ethernet Ring Path Summary
-----
Path Port      Raps-Tag      Admin/Oper      Type      Fwd State
-----
a 1/1/3        2.1                Up/Up          normal    unblocked
b -            -                  -/-            -         -
=====
*A:PE-1#
```

In this case, if a topology change event occurs in the sub-ring, an LDP flush all-from-me message is sent by PE-1 and PE-3 to their LDP peers. This can be seen by enabling the following debugging for PE-1, where packets 1 and 2 are the flush messages.

```
*A:PE-1# debug router ldp peer 192.0.2.2 packet init
*A:PE-1# debug router ldp peer 192.0.2.3 packet init
*A:PE-1#
*A:PE-1# show debug
debug
    router "Base"
        ldp
            peer 192.0.2.2
                event
                exit
                packet
                    init
                exit
            exit
            peer 192.0.2.3
                event
                exit
                packet
                    init
                exit
            exit
        exit
    exit
*A:PE-1#
*A:PE-1# configure port 1/1/3 shutdown
```

G.8032 Ethernet Ring Protection Multiple Ring Topology

```
2 2014/09/20 11:28:36.01 PDT WARNING: SNMP #2004 Base 1/1/3
"Interface 1/1/3 is not operational"

3 2014/09/20 11:28:36.01 PDT MINOR: ERING #2001 Base eth-ring-2
"Eth-Ring 2 path 0 changed fwd state to blocked"

1 2014/09/20 11:28:36.01 PDT MINOR: DEBUG #2001 Base LDP
"LDP: LDP
Send Address Withdraw packet (msgId 6799) to 192.0.2.2:0
MAC Flush (All MACs learned from me)
Service FEC PWE3: ENET(5)/11 Group ID = 0 cBit = 0
"
*A:PE-1#
2 2014/09/20 11:28:36.01 PDT MINOR: DEBUG #2001 Base LDP
"LDP: LDP
Send Address Withdraw packet (msgId 183) to 192.0.2.3:0
MAC Flush (All MACs learned from me)
Service FEC PWE3: ENET(5)/11 Group ID = 0 cBit = 0
"

4 2014/09/20 11:28:36.03 PDT MAJOR: SVCMMGR #2210 Base
"Processing of an access port state change event is finished and the status of a
11 affected SAPs on port 1/1/3 has been updated."

5 2014/09/20 11:28:38.99 PDT MINOR: ETH_CFM #2001 Base
"MEP 1/4/7 highest defect is now defRemoteCCM"

3 2014/09/20 11:28:39.27 PDT MINOR: DEBUG #2001 Base LDP
"LDP: LDP
Recv Address Withdraw packet (msgId 184) from 192.0.2.3:0
"

*A:PE-1#
```

Operational Procedures

Operators may wish to configure rings with or without control over reversion. Reversion can be controlled by timers or the ring can be run without reversion allowing the operator to choose when the ring reverts. To change a ring topology, the **manual** or **force** switch command may be used to block a specified ring path. A ring will still address failures when run without reversion but will not automatically revert to the RPL when resources are restored. A **clear** command can be used to clear the manual or force state of a ring.

The following **tools** commands are available to control the state of paths on a ring.

```
tools perform eth-ring clear <ring-index>
tools perform eth-ring force <ring-index> path {a|b}
tools perform eth-ring manual <ring-index> path {a|b}
```

In the output below, path “b” of eth-ring 1 is manually blocked then cleared.

```
*A:PE-1# show eth-ring 1
=====
Ethernet Ring 1 Information
=====
Description          : Ethernet Ring 1
Admin State          : Up                Oper State          : Up
Node ID              : d8:30:ff:00:00:00
Guard Time           : 5 deciseconds    RPL Node             : rplNone
Max Revert Time      : 60 seconds        Time to Revert       : N/A
CCM Hold Down Time   : 0 centiseconds    CCM Hold Up Time     : 20 deciseconds
Compatible Version    : 2
APS Tx PDU           : N/A
Defect Status         :

Sub-Ring Type        : none
-----
Ethernet Ring Path Summary
-----
Path Port    Raps-Tag    Admin/Oper    Type          Fwd State
-----
a 1/1/1      1.1           Up/Up         normal        unblocked
b 1/1/2      1.1           Up/Up         normal        unblocked
=====
*A:PE-1#
*A:PE-1# tools perform eth-ring manual 1 path b
*A:PE-1# show eth-ring 1
=====
Ethernet Ring 1 Information
=====
Description          : Ethernet Ring 1
Admin State          : Up                Oper State          : Up
Node ID              : d8:30:ff:00:00:00
Guard Time           : 5 deciseconds    RPL Node             : rplNone
Max Revert Time      : 60 seconds        Time to Revert       : N/A
CCM Hold Down Time   : 0 centiseconds    CCM Hold Up Time     : 20 deciseconds
Compatible Version    : 2
```



```

APS Tx PDU          : Request State: 0x7
                     Sub-Code      : 0x0
                     Status        : 0x20 ( BPR )
                     Node ID       : d8:30:ff:00:00:00
Defect Status       :

Sub-Ring Type       : none
-----
Ethernet Ring Path Summary
-----
Path Port      Raps-Tag  Admin/Oper  Type      Fwd State
-----
a  1/1/1      1.1         Up/Up      normal    unblocked
b  1/1/2      1.1         Up/Up      normal    blocked
=====
*A:PE-1#
*A:PE-1# tools perform eth-ring clear 1
*A:PE-1# show eth-ring 1
=====
Ethernet Ring 1 Information
=====
Description      : Ethernet Ring 1
Admin State      : Up
Node ID          : d8:30:ff:00:00:00
Guard Time       : 5 deciseconds
Max Revert Time  : 60 seconds
CCM Hold Down Time : 0 centiseconds
Compatible Version : 2
APS Tx PDU       : N/A
Defect Status     :

Sub-Ring Type     : none
-----
Ethernet Ring Path Summary
-----
Path Port      Raps-Tag  Admin/Oper  Type      Fwd State
-----
a  1/1/1      1.1         Up/Up      normal    unblocked
b  1/1/2      1.1         Up/Up      normal    unblocked
=====
*A:PE-1#

```

Both the **manual** and **force** command block the path specified, however, the **manual** command fails if there is an existing forced switch or signal fail event in the ring, as seen below. The **force** command will block the port regardless of any existing ring state and there can be multiple force states simultaneously on a ring on different nodes.

```

*A:PE-1# tools perform eth-ring manual 1 path b
INFO: ERMGR #1001 Not permitted - The switch command is not compatible to the current state
(MS), effective priority (MS) or rpl-node type (None)

```

Conclusion

Ethernet Ring APS provides an optimal solution for designing native Ethernet services with ring topology. With sub-rings, both multiple rings and access rings increase the versatility of G.8032. G.8032 has been expanded to more of the SR/ESS platforms by allowing R-APS with slower MEPs (including CCMs intervals of 1 second). This protocol provides simple configuration, operation and guaranteed fast protection time. The implementation also has a flexible encapsulation that allows dot1q, QinQ or PBB for the ring traffic. It could be utilized on various services such as mobile backhaul, business VPN access, aggregation and core.

G.8032 Ethernet Ring Protection Single Ring Topology

In This Chapter

This section provides information about G.8032 Ethernet ring protection single ring topology.

Topics in this section include:

- [Applicability on page 800](#)
- [Overview on page 801](#)
- [Configuration on page 804](#)
- [Conclusion on page 822](#)

Applicability

This example is applicable to the 7950 XRS (as of 10.0.R4), the 7750 SR-7/12 and 7450 ESS-7/12 (as of 9.0.R1), and the 7450 ESS-6/6v with IOM3-XP or IMM and 7750 SR-c4/12 (as of 11.0.R1). It is not supported on a 7750 SR-1, 7450 ESS-1, 7710 SR, or using an IOM-2 or lower.

The configuration was tested on release 12.0.R5, and covers ring protection for a single ring. Protection for multiple ring topologies is covered in [G.8032 Ethernet Ring Protection Multiple Ring Topology on page 751](#).

Overview

G.8032 Ethernet ring protection is supported for data service SAPs within a regular VPLS service, a PBB VPLS (I/B-component) or a routed VPLS (R-VPLS). G.8032 is one of the fastest protection schemes for Ethernet networks.

ITU-T G.8032v2 specifies protection switching mechanisms and a protocol for Ethernet layer network (ETH) Ethernet rings. Ethernet rings can provide wide-area multi-point connectivity more economically due to their reduced number of links. The mechanisms and protocol defined in ITU-T G.8032v2 achieve highly reliable and stable protection and never form loops, which would negatively affect network operation and service availability. Each ring node is connected to adjacent nodes participating in the same ring using two independent paths, which use ring links (configured on ports or LAGs). A ring link is bounded by two adjacent nodes and a port for a ring link is called a ring port. The minimum number of nodes on a ring is two.

The fundamentals of this ring protection switching architecture are:

- the principle of loop avoidance and
- the utilization of learning, forwarding, and address table mechanisms defined in the ITU-T G.8032v2 Ethernet flow forwarding function (ETH_FF) (Control plane).

Loop avoidance in the ring is achieved by guaranteeing that, at any time, traffic may flow on all but one of the ring links. This particular link is called the Ring Protection Link (RPL) and under normal conditions this link is blocked, so it is not used for traffic. One designated node, the RPL Owner, is responsible to block traffic over the one designated RPL. Under a ring failure condition, the RPL Owner is responsible for unblocking the RPL, allowing the RPL to be used for traffic. The protocol ensures that even without an RPL owner defined, one link will be blocked and it operates as a “break before make protocol”, specifically the protocol guarantees that no link is restored until a different link in the ring is blocked. The other side of the RPL is configured as an RPL neighbor. An RPL neighbor blocks traffic on the link.

The event of a ring link or ring node failure results in protection switching of the traffic. This is achieved under the control of the ETH_FF functions on all ring nodes. A Ring Automatic Protection Switching (R-APS) protocol is used to coordinate the protection actions over the ring. The protection switching mechanisms and protocol supports a multi-ring/ladder network that consists of connected Ethernet rings, however, that is not covered in this example.

Ring Protection Mechanism

The Ring Protection protocol is based on the following building blocks:

- Ring status change on failure
 - Idle -> Link failure -> Protection -> Recovery -> Idle
- Ring Control State changes
 - Idle -> Protection -> Manual Switch -> Forced Switch -> Pending
- Re-use existing ETH OAM
 - Monitoring : ETH Continuity Check messages
 - Failure Notification : Y.1731 Signal Failure
- Forwarding Database MAC Flush on ring status change
- RPL (Ring Protection Link)
 - Defines blocked link in idle status

[Figure 130](#) shows a ring of six nodes, with the RPL owner on the top right. One link of the RPL owner is designated to be the RPL and will be blocked in order to prevent a loop. Schematics of the physical and logical topologies are also shown.

When an RPL owner and RPL end are configured, the associated link will be the RPL when the ring is fully operational and so be blocked by the RPL owner. If a different ring link fails then the RPL will be unblocked by the RPL owner. When the failed link recovers, it will initially be blocked by one of its adjacent nodes. The adjacent node sends an R-APS message across the ring to indicate the error is cleared and after a configurable time, if reversion is enabled, the RPL will revert to being blocked with all other links unblocked. This ensures that the ring topology is predictable when fully operational.

If a specific RPL owner is not configured, then the last link to become active will be blocked and the ring will remain in this state until another link fails. However, this operation makes the selection of the blocked link non-deterministic.

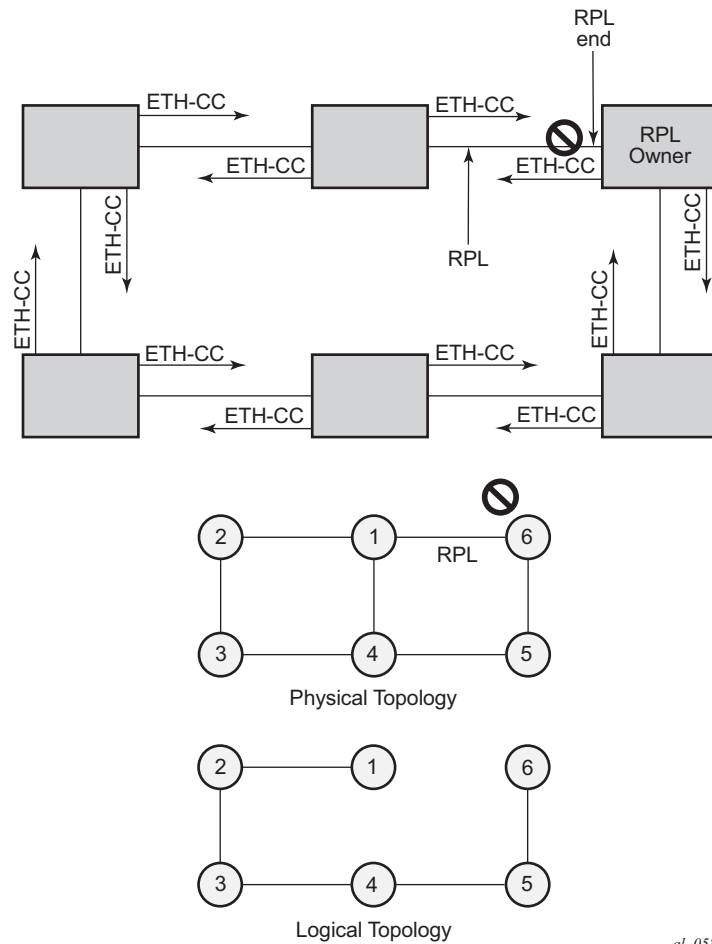
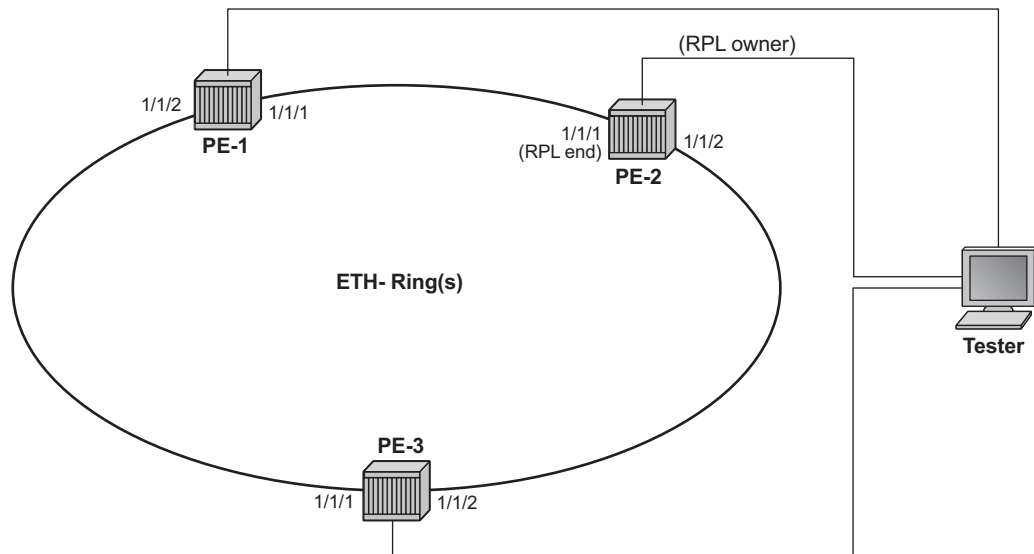


Figure 130: G.8032 Operation and Topologies

The protection protocol uses a specific control VLAN, with the associated data VLANs taking their forwarding state from the control VLAN.

Configuration

The test topology is shown in [Figure 131](#).



** Control Channel: VPLS 10, Tag 1
 ** Data Channel: VPLS 100, Tag 100

al_0589

Figure 131: Test Topology

The **eth-ring** configuration commands are shown below.

```
configure
  eth-ring <ring-index>
    ccm-hold-time { [down <down-timeout>] [up <up-timeout>] }
    compatible-version <version>
    description <description-string>
    guard-time <time>
    node-id <xx:xx:xx:xx:xx:xx or xx-xx-xx-xx-xx-xx>
    path {a|b} [ { <port-id>|<lag-id> } raps-tag <qtag>[.<qtag>] ]
    description <description-string>
    eth-cfm
      mep <mep-id> domain <md-index> association <ma-index>
      ...
    rpl-end
    shutdown
    revert-time <time>
    rpl-node {owner|nbr}
    shutdown
```


Parameters:

- *ring-index* — This is the number by which the ring is referenced, values: 1 to 128.
 - **ccm-hold-time** {[down <down-timeout>] [up <up-timeout>]}
 - down — This command specifies the timer that controls the delay between detecting that ring path is down and reporting it to the G.8032 protection module. If a non-zero value is configured, the system will wait for the time specified in the value parameter before reporting it to the G.8032 protection module. Note that this parameter applies only to ring path CCM. It does *not* apply to the ring port link state. To dampen ring port link state transitions, use the hold-time parameter from the physical member port. This is useful if the underlying path between two nodes is going across an optical system which implements its own protection.
 - up — This command specifies the timer which controls the delay between detecting that ring path is up and reporting it to the G.8032 protection module. If a non-zero value is configured, the system will wait for the time specified in the value parameter before reporting it to the G.8032 protection module. Note that this parameter applies only to ring path CCM. It does *not* apply to the member port link state. To dampen member port link state transitions, use the hold-time parameter from the physical member port.
- Values: <down-timeout> : [0..5000] in deciseconds - Default: 0
 <up-timeout> : [0..5000] in deciseconds - Default: 20
 1 centisecond = 10ms
 1 decisecond = 100ms
- *compatible version* — This command configures eth-ring compatibility version for the G.8032 state machine and messages. The default is version 2 (ITU G.8032v2) and all 7x50 systems use version 2. If there is a need to interwork with third party devices that only support version 1, this can be set to version 1 allowing the reception of version 1 PDUs. Note that version 2 is encoded as 1 in the R-APS messages. Compatibility allows the reception of version 1 (encoded as 0) R-APS PDUs but, as per the G.8032 specification, higher versions are ignored on reception. For the SR/ESS, messages are always originated with version 2. Therefore if a third party switch supported version 3 (encoded as 2) or higher interworking is also supported provided the other switch is compatible with version 2.
 - *description* <description-string> — This configures a text string, up to 80 characters, which can be used to describe the use of the eth-ring.
 - *guard-time* <time> — The forwarding method, in which R-APS messages are copied and forwarded at every Ethernet ring node, can result in a message corresponding to an old request, that is no longer relevant, being received by Ethernet ring nodes. Reception of an old R-APS message may result in erroneous ring state interpretation by some Ethernet ring nodes. The guard timer is used to prevent Ethernet ring nodes from acting upon outdated R-APS messages and prevents the possibility of forming a closed loop. Messages are not forwarded when the guard-timer is running.

Values: [1..20] in deciseconds - Default: 5
1 decisecond = 100ms

- **node-id** <xx:xx:xx:xx:xx:xx or xx-xx-xx-xx-xx-xx> — This allows the node identifier to be explicitly configured. By default the chassis MAC is used. It is not required in typical configurations.
- **path** {a|b} [{<port-id>|<lag-id>} raps-tag <qtag>[.<qtag>]] — This parameter defines the paths around the ring, of which there are two in different directions on the ring: an “a” path and a “b” path. In addition it configures the encapsulation used for the R-APS messages on the ring. These can be either single or double tagged.
 - **description** <description-string> — This configures a text string, up to 80 characters, which can be used to describe the use of the path.
 - **eth-cfm** — Configures the associated Ethernet CFM parameters.
 - **mep** <mep-id> domain <md-index> association <ma-index> — The MEP defined under the path is used for the G.8032 protocol messages, which are based on IEEE 802.1ag/Y.1731 CFM frames.
 - **rpl-end** — When configured, this path is expected to be one end of the RPL. This parameter must be configured in conjunction with the **rpl-node**.
 - **shutdown** — This command shuts down the path.
- **revert-time** <time> — This command configures the revert time for an Eth-Ring. Revert time is the time that the RPL will wait before returning to the blocked state. Configuring “no revert-time” disables reversion, effectively setting the revert-time to zero.

Values: [60..720] in seconds - Default: 300

- **rpl-node** {owner|nbr} — A node can be designated as either the owner of the RPL, in which case this node is responsible for the RPL, or the nbr, in which case this node is expected to be the neighbor to the RPL owner across the RPL. The nbr is optional and is included to be compliant with the specification. This parameter must be configured in conjunction with the **rpl-end** parameter.
- **shutdown** — This command shuts down the ring.

Prerequisites

Create following log-id on PE-2 to see major events logged to the console on PE-2.

```
configure
  log
    log-id 1
      from main
      to console
  exit
```

To configure R-APS, there should be at least 2 VPLS services for 1 Eth-Ring instance, one for the control channel and the other (or more) for data channel(s). The control channel is used for R-APS signaling while data channel is for user data traffic. The state of the data channels is inherited from the state of the control channel.

Step 0. Configuring the encapsulation for each ring port

Eth-Ring needs R-APS tags to send and receive G.8032 signaling messages. To configure a control channel, an access SAP configuration is required on each path a/b port. The SAP configuration follows that of the port and must be either *dot1q* or *qinq*, consequently the control and data packets are either single tagged or double tagged. It is also possible to have the control VPLS using single tagged frames with the data VPLSs using double tagged framed; this requires the system to be configured with the **new-qinq-untagged-sap** parameter (**configure system ethernet new-qinq-untagged-sap**), with the ring path raps-tags and control VPLS SAPs configured as qtag.0, and the data VPLSs configured as QinQ SAPs.

In this example single tags are used so the commands for ring node PE-1 are:

```
*A:PE-1# configure port 1/1/1 ethernet mode access
*A:PE-1# configure port 1/1/2 ethernet mode access
*A:PE-1# configure port 1/1/1 ethernet encap-type dot1q
*A:PE-1# configure port 1/1/2 ethernet encap-type dot1q
```

Step 1. Configuring ETH-CFM

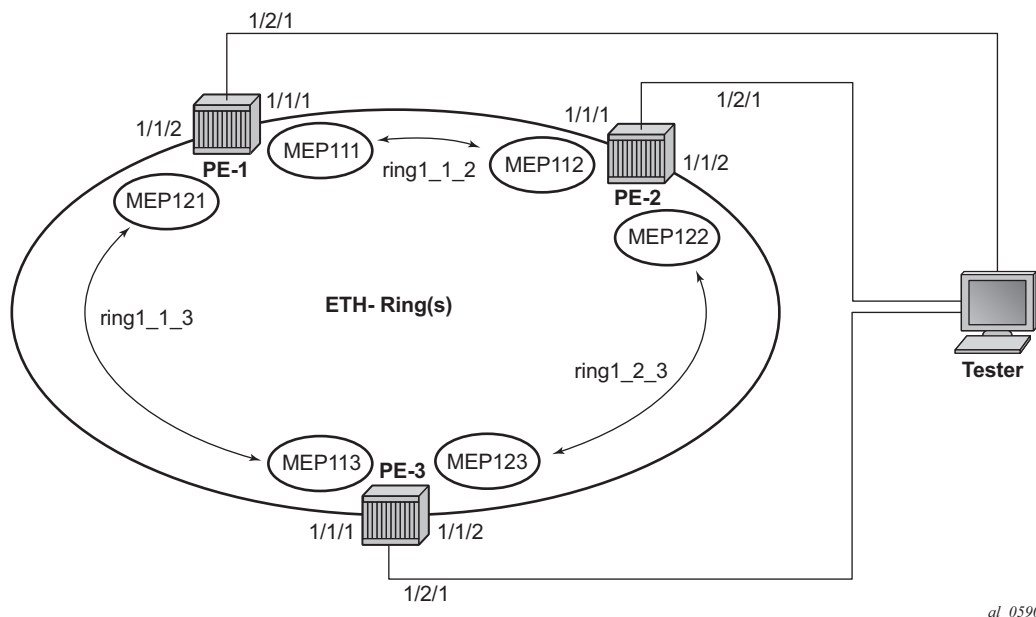
Ethernet Ring requires Eth-CFM domain(s), association(s) and MEP(s) being configured. The domain format should be none and association name should be icc-based (Y.1731). The minimum CCM interval for the 7x50 is 10ms. The eth-ring MEP requires sub-second CCM interval (10ms or 100ms) to be configured (or 1 second from 11.0.R1 or later).

Note that the MEPs used for R-APS control normally will have CCM configured on the control channel path MEPs for failure detection. Alternatively, detecting a failure of the ring may be achieved by running Ethernet in the First Mile (EFM) at the port level if CCM is not possible at 100ms or 10ms (or 1 second as of release 11.0.R1). Loss-of-signal, in conjunction with other OAM, is applicable only when the nodes are directly connected.

Prerequisites

To omit the failure detecting CCMs, it would be necessary to remove the *ccm-enable* from under the path MEPs and to remove the *remote-mepids* from under the *eth-cfm* associations on all nodes.

Figure 132 shows the Ethernet CFM configuration used here.



al_0590

Figure 132: Ethernet CFM Configuration

The configuration of each node is as follows.

PE-1:

```
*A:PE-1>config>eth-cfm# info
-----
domain 1 format none level 3
  association 1 format icc-based name "ring1_1_2"
    ccm-interval 1
    remote-mepid 112
  exit
  association 2 format icc-based name "ring1_1_3"
    ccm-interval 1
    remote-mepid 113
  exit
exit
-----
*A:PE-1>config>eth-cfm#
```

PE-2:

```
A:PE-2>config>eth-cfm# info
-----
domain 1 format none level 3
  association 1 format icc-based name "ring1_1_2"
    ccm-interval 1
    remote-mepid 111
  exit
  association 2 format icc-based name "ring1_2_3"
    ccm-interval 1
    remote-mepid 123
  exit
exit
-----
A:PE-2>config>eth-cfm#
```

PE-3:

```
A:PE-3>config>eth-cfm# info
-----
domain 1 format none level 3
  association 1 format icc-based name "ring1_1_3"
    ccm-interval 1
    remote-mepid 121
  exit
  association 2 format icc-based name "ring1_2_3"
    ccm-interval 1
    remote-mepid 122
  exit
exit
-----
A:PE-3>config>eth-cfm#
```

Step 2. Configuring Eth-Ring

Two paths should be configured to form a ring. In this example, VLAN tag 1 is used as control channel for R-APS signaling in the ring.

PE-1:

```
*A:PE-1>config>eth-ring# info
-----
      path a 1/1/1 raps-tag 1
        eth-cfm
          mep 111 domain 1 association 1
            ccm-enable
            control-mep
            no shutdown
          exit
        exit
      no shutdown
    exit
  path b 1/1/2 raps-tag 1
    eth-cfm
      mep 121 domain 1 association 2
        ccm-enable
        control-mep
        no shutdown
      exit
    exit
  no shutdown
exit
no shutdown
-----
*A:PE-1>config>eth-ring#
```

It is mandatory to configure a MEP under the path otherwise this error will be displayed.

```
*A:PE-1>config>eth-ring>path# no shutdown
INFO: ERMGR #1001 Not permitted - must configure eth-cfm MEP first
*A:PE-1>config>eth-ring>path#
```

Note that while MEPs are mandatory, enabling CCMs on the MEPs under the paths as a failure detection mechanism is optional.

PE-2:

In order to define the RPL, node PE-2 has been configured as the RPL owner and path “a” as the RPL end. The link between nodes PE-1 and PE-2 will be the RPL with node PE-2 blocking that link when the ring is fully operational.

```

A:PE-2>config>eth-ring# info
-----
      revert-time 60
      rpl-node owner
      path a 1/1/1 raps-tag 1
        rpl-end
        eth-cfm
          mep 112 domain 1 association 1
            ccm-enable
            control-mep
            no shutdown
          exit
        exit
      no shutdown
    exit
  path b 1/1/2 raps-tag 1
    eth-cfm
      mep 122 domain 1 association 2
        ccm-enable
        control-mep
        no shutdown
      exit
    exit
  no shutdown
exit
no shutdown
-----
A:PE-2>config>eth-ring#

```

It is not permitted to configure a path as an RPL end without having configured the node on this ring to be either the RPL *owner* or *nbr* otherwise the following error message is reported.

```

*A:PE-2>config>eth-ring# path a rpl-end
INFO: ERMGR #1001 Not permitted - path-type rpl-end is not consistent with eth-ring 'rpl-
node' type
*A:PE-2>config>eth-ring#

```

PE-3:

```

A:PE-3>config>eth-ring# info
-----
      path a 1/1/1 raps-tag 1
        eth-cfm
          mep 113 domain 1 association 1
            ccm-enable
            control-mep
            no shutdown
          exit
        exit
      no shutdown
    exit
  path b 1/1/2 raps-tag 1
    eth-cfm

```

Prerequisites

```

        mep 123 domain 1 association 2
        ccm-enable
        control-mep
        no shutdown
    exit
exit
no shutdown
exit
no shutdown
-----
A:PE-3>config>eth-ring#
```

Until the Ethernet Ring instance is attached to the service (VPLS in this case), the ring operational status is down and the forwarding status of each port is blocked. This prevents operator from creating a loop by mis-configuration. This state can be seen on ring node PE-1 as follows

```

*A:PE-1# show eth-ring 1
=====
Ethernet Ring 1 Information
=====
Description          : (Not Specified)
Admin State          : Up              Oper State           : Down
Node ID              : ea:4b:ff:00:00:00
Guard Time           : 5 deciseconds  RPL Node             : rplNone
Max Revert Time       : 300 seconds    Time to Revert        : N/A
CCM Hold Down Time    : 0 centiseconds CCM Hold Up Time     : 20 deciseconds
Compatible Version    : 2
APS Tx PDU           : Request State: 0xB
                     : Sub-Code       : 0x0
                     : Status         : 0x20 ( BPR )
                     : Node ID        : ea:4b:ff:00:00:00
Defect Status         :

Sub-Ring Type         : none
=====
Ethernet Ring Path Summary
=====
Path Port    Raps-Tag    Admin/Oper    Type    Fwd State
-----
a 1/1/1      1              Up/Down      normal   blocked
b 1/1/2      1              Up/Down      normal   blocked
=====
*A:PE-1#
```


Step 3. Adding eth-ring SAP to the control channel service.

Path a and b defined in the eth-ring must be added as SAPs into a VPLS service (standard VPLS in this example) using the *eth-ring* parameter. The SAP encapsulation values must match the values of the *raps-tag* configured for the associated path.

G.8032 uses the same raps-tag value on all nodes on the ring, as configured in this example. However, the 7x50 implementation relaxes this constraint by requiring the tag to match only on adjacent nodes.

PE-1:

```
*A:PE-1# configure service vpls 10 customer 1 create
*A:PE-1>config>service>vpls# info
-----
      stp
      shutdown
    exit
  sap 1/1/1:1 eth-ring 1 create
      stp
      shutdown
    exit
  exit
  sap 1/1/2:1 eth-ring 1 create
      stp
      shutdown
    exit
  exit
  no shutdown
-----
*A:PE-1>config>service>vpls#
```

PE-2:

```
*A:PE-2# /configure service vpls 10 customer 1 create
*A:PE-2>config>service>vpls# info
-----
      stp
      shutdown
    exit
  sap 1/1/1:1 eth-ring 1 create
      stp
      shutdown
    exit
  exit
  sap 1/1/2:1 eth-ring 1 create
      stp
      shutdown
    exit
  exit
  no shutdown
-----
*A:PE-2>config>service>vpls#
```

PE-3:

```
A:PE-3# configure service vpls 10 customer 1 create
A:PE-3>config>service>vpls# info
```

```
-----
      stp
        shutdown
      exit
      sap 1/1/1:1 eth-ring 1 create
        stp
          shutdown
        exit
      exit
      sap 1/1/2:1 eth-ring 1 create
        stp
          shutdown
        exit
      exit
      no shutdown
-----
A:PE-3>config>service>vpls#
```

Note that you cannot add a normal SAP or SDP in a control channel VPLS, only SAPs with an *eth-ring* parameter can be added. Trying to add a SAP without this parameter into a control channel VPLS will result in the message below being displayed.

```
A:PE-3>config>service>vpls# sap 1/2/1:1 create
MINOR: SVCMMGR #1321 Service contains an Ethernet ring control SAP
A:PE-3>config>service>vpls#
```

Now the Eth-Ring is operationally up and the RPL is blocking successfully on ring node PE-2 port 1/1/1, as expected from the RPL owner/end configuration.

An overview of all of the ring(s) can be shown using the following commands, in this case on node PE-2.

First, the ETH ring status is shown.

```
*A:PE-2# show eth-ring status
=====
Ethernet Ring (Status information)
=====
```

Ring ID	Admin State	Oper State	Path Information			MEP Information		
			Path	Tag	State	Ctrl-MEP	CC-Intvl	Defects
1	Up	Up	a - 1/1/1	1	Up	Yes	1	-----
			b - 1/1/2	1	Up	Yes	1	-----

```
=====
Ethernet Tunnel MEP Defect Legend:
R = Rdi, M = MacStatus, C = RemoteCCM, E = ErrorCCM, X = XconCCM
*A:PE-2#
```

The ring and path forwarding states is shown with following command.

```
*A:PE-2# show eth-ring
=====
Ethernet Rings (summary)
=====
```

Ring ID	Int ID	Admin State	Oper State	Paths	Summary	Path a	States b
1	-	Up	Up	a - 1/1/1	1	b - 1/1/2	1 B U

```
=====
Ethernet Ring Summary Legend:  B - Blocked    U - Unblocked
*A:PE-2#
```

Specific ring information can be shown on each node separately, as follows.

PE-1:

```
*A:PE-1# show eth-ring 1
=====
Ethernet Ring 1 Information
=====
```

Description	: (Not Specified)		
Admin State	: Up	Oper State	: Up
Node ID	: ea:4b:ff:00:00:00		
Guard Time	: 5 deciseconds	RPL Node	: rplNone
Max Revert Time	: 300 seconds	Time to Revert	: N/A
CCM Hold Down Time	: 0 centiseconds	CCM Hold Up Time	: 20 deciseconds
Compatible Version	: 2		
APS Tx PDU	: N/A		
Defect Status	:		
Sub-Ring Type	: none		

```
-----
Ethernet Ring Path Summary
-----
```

Path	Port	Raps-Tag	Admin/Oper	Type	Fwd State
a	1/1/1	1	Up/Up	normal	unblocked
b	1/1/2	1	Up/Up	normal	unblocked

```
=====
*A:PE-1#
```

PE-2:

```
*A:PE-2# show eth-ring 1
=====
Ethernet Ring 1 Information
=====
```

Description	: (Not Specified)		
Admin State	: Up	Oper State	: Up
Node ID	: ea:4c:ff:00:00:00		
Guard Time	: 5 deciseconds	RPL Node	: rplOwner

Prerequisites

```
Max Revert Time      : 60 seconds      Time to Revert      : N/A
CCM Hold Down Time   : 0 centiseconds  CCM Hold Up Time    : 20 deciseconds
Compatible Version    : 2
APS Tx PDU           : Request State: 0x0
                     Sub-Code          : 0x0
                     Status            : 0x80 ( RB )
                     Node ID           : ea:4c:ff:00:00:00
Defect Status         :
```

```
Sub-Ring Type        : none
```

```
-----
Ethernet Ring Path Summary
-----
```

Path	Port	Raps-Tag	Admin/Oper	Type	Fwd State
a	1/1/1	1	Up/Up	rplEnd	blocked
b	1/1/2	1	Up/Up	normal	unblocked

```
=====
*A:PE-2#
```

Note that node PE-2 is the RPL owner and that port 1/1/1 is the RPL end. The *revert-time* shows the configured value.

When a revert is pending, the “Time to Revert” will show the number of seconds remaining before the revert occurs, as below.

```
*A:PE-2# show eth-ring 1
```

```
=====
Ethernet Ring 1 Information
=====
```

```
Description      : (Not Specified)
Admin State       : Up                Oper State       : Up
Node ID          : ea:4c:ff:00:00:00
Guard Time       : 5 deciseconds      RPL Node         : rplOwner
Max Revert Time  : 60 seconds          Time to Revert    : 45 seconds
CCM Hold Down Time : 0 centiseconds    CCM Hold Up Time  : 20 deciseconds
Compatible Version : 2
APS Tx PDU       : N/A
Defect Status     :
```

```
Sub-Ring Type        : none
```

```
-----
Ethernet Ring Path Summary
-----
```

Path	Port	Raps-Tag	Admin/Oper	Type	Fwd State
a	1/1/1	1	Up/Up	rplEnd	unblocked
b	1/1/2	1	Up/Up	normal	unblocked

```
=====
*A:PE-2#
```

On reversion, the following console message is logged.

```
9 2014/10/03 12:54:06.84 UTC MINOR: ERING #2001 Base eth-ring-1
"Eth-Ring 1 path 0 changed fwd state to blocked"
```

PE-3:

```
*A:PE-3# show eth-ring 1
=====
Ethernet Ring 1 Information
=====
Description      : (Not Specified)
Admin State      : Up                Oper State       : Up
Node ID          : ea:4d:ff:00:00:00
Guard Time       : 5 deciseconds    RPL Node         : rplNone
Max Revert Time  : 300 seconds       Time to Revert    : N/A
CCM Hold Down Time : 0 centiseconds CCM Hold Up Time : 20 deciseconds
Compatible Version : 2
APS Tx PDU       : N/A
Defect Status     :

Sub-Ring Type     : none
-----
Ethernet Ring Path Summary
-----
Path Port      Raps-Tag   Admin/Oper   Type        Fwd State
-----
a 1/1/1        1           Up/Up        normal       unblocked
b 1/1/2        1           Up/Up        normal       unblocked
=====
*A:PE-3#
```

Finally, the details of an individual path can be shown.

```
*A:PE-2# show eth-ring 1 path b
=====
Ethernet Ring 1 Path Information
=====
Description      : (Not Specified)
Port             : 1/1/2              Raps-Tag         : 1
Admin State      : Up                Oper State       : Up
Path Type        : normal             Fwd State        : unblocked
                                           Fwd State Change : 10/03/2014 11:56:17

Last Switch Command: noCmd
APS Rx PDU       : Request State: 0x0
                  Sub-Code         : 0x0
                  Status            : 0x00 ( )
                  Node ID           : ea:4d:ff:00:00:00
=====
*A:PE-2#
```

Step 4. Configuring the user data channel VPLS service

The user data channels are created on a separate VPLS, vpls 100 in the example. Tag 100 and VPLS 100 are used here. The ring data channels must be on the same ports as the corresponding control channels configured above. The access into the data services can use SAPs and/or SDPs.

PE-1:

```
*A:PE-1# configure service vpls 100
*A:PE-1>config>service>vpls# info
-----
      stp
      shutdown
    exit
  sap 1/1/1:100 eth-ring 1 create
      stp
      shutdown
    exit
  exit
  sap 1/1/2:100 eth-ring 1 create
      stp
      shutdown
    exit
  exit
  sap 1/2/1:100 create
      stp
      shutdown
    exit
  exit
  no shutdown
-----
*A:PE-1>config>service>vpls#
```

PE-2:

```
*A:PE-2# configure service vpls 100
*A:PE-2>config>service>vpls# info
-----
      stp
      shutdown
    exit
  sap 1/1/1:100 eth-ring 1 create
      stp
      shutdown
    exit
  exit
  sap 1/1/2:100 eth-ring 1 create
      stp
      shutdown
    exit
  exit
  sap 1/2/1:100 create
      stp
      shutdown
```

```

        exit
    exit
    no shutdown
-----
*A:PE-2>config>service>vpls#

```

PE-3:

```

*A:PE-3# configure service vpls 100
*A:PE-3>config>service>vpls# info
-----
    stp
        shutdown
    exit
    sap 1/1/1:100 eth-ring 1 create
        stp
            shutdown
        exit
    exit
    sap 1/1/2:100 eth-ring 1 create
        stp
            shutdown
        exit
    exit
    sap 1/2/1:100 create
        stp
            shutdown
        exit
    exit
    no shutdown
-----
*A:PE-3>config>service>vpls#

```

All of the SAPs which are configured to use ETH rings can be shown, using PE-1 as an example.

```

*A:PE-1# show service sap-using eth-ring
=====
Service Access Points (Ethernet Ring)
=====
SapId          SvcId          Eth-Ring Path Admin Oper  Blocked Control/
                State State         Data
-----
1/1/1:1         10             1      a   Up   Up   No   Ctrl
1/1/2:1         10             1      b   Up   Up   No   Ctrl
1/1/1:100       100            1      a   Up   Up   No   Data
1/1/2:100       100            1      b   Up   Up   No   Data
-----
Number of SAPs : 4
=====
*A:PE-1#

```

To see an example of the console messages on a ring failure, when the unblocked port (1/1/2) on node PE-2 is shutdown, the following messages are displayed.

```
*A:PE-2# configure port 1/1/2 shutdown

10 2014/10/03 12:56:18.03 UTC WARNING: SNMP #2004 Base 1/1/2
"Interface 1/1/2 is not operational"

11 2014/10/03 12:56:18.03 UTC MINOR: ERING #2001 Base eth-ring-1
"Eth-Ring 1 path 1 changed fwd state to blocked"

12 2014/10/03 12:56:18.03 UTC MINOR: ERING #2001 Base eth-ring-1
"Eth-Ring 1 path 0 changed fwd state to unblocked"

13 2014/10/03 12:56:18.04 UTC MAJOR: SVCNMR #2210 Base
"Processing of an access port state change event is finished and the status of a
11 affected SAPs on port 1/1/2 has been updated."

14 2014/10/03 12:56:21.85 UTC MINOR: ETH_CFM #2001 Base
"MEP 1/2/122 highest defect is now defRemoteCCM"
```

For troubleshooting, the **tools dump eth-ring <ring-index>** command displays path information, the internal state of the control protocol, related statistics information and up to the last 16 protocol events (including messages sent and received, and the expiration of timers). An associated parameter *clear* exists, clearing the event information in this output when the command is entered. The following is an example of the output on node PE-2 with port 1/1/2 active.

```
*A:PE-2# tools dump eth-ring 1

ringId 1 (Up/Up): numPaths 2 nodeId ea:4c:ff:00:00:00
SubRing: none (interconnect ring 0, propagateTc No), Cnt 0
  path-a, port 1/1/1 (Up), tag 1.0(Up) status (Up/Up/Blk)
    cc (Dn/Up): Cnt 5/5 tm 000 17:43:30.030/000 17:46:05.690
    state: Cnt 23 B/F 000 17:54:59.030/000 17:52:31.220, flag: 0x0
  path-b, port 1/1/2 (Up), tag 1.0(Up) status (Up/Up/Fwd)
    cc (Dn/Up): Cnt 5/5 tm 000 17:52:35.040/000 17:53:56.900
    state: Cnt 8 B/F 000 17:52:31.220/000 17:53:59.890, flag: 0x0
FsmState= IDLE, Rpl = Owner, revert = 60 s, guard = 5 ds
Defects =
Running Timers = PduReTx
lastTxPdu = 0x0080 Nr(RB )
path-a Rpl, RxId(I)= ea:4d:ff:00:00:00, rx= v1-0x0020 Nr, cmd= None
path-b Normal, RxId(I)= ea:4d:ff:00:00:00, rx= v1-0x0020 Nr, cmd= None
DebugInfo: aPathSts 7, bPathSts 7, pm (set/cfr) 0/0, txFlush 0
RxRaps: ok 67 nok 0 self 3447, TmrExp - wtr 11(1), grd 7, wtb 0
Flush: cnt 31 (16/15/0) tm 000 17:54:59.030-000 17:54:59.030 Out/Ack 0/1
RxRawRaps: aPath 12397 bPath 12544 vPath 0
Now: 000 17:55:40.030 , softReset: No - noTx 0

Seq Event RxInfo(Path: NodeId-Bytes)
      state:TxInfo (Bytes)          Dir  pA  pB          Time
=== =====
010  aUp
      PEND-G: 0x0000 Nr              Tx--> Blk Fwd 000 17:46:08.230
011  pdu B: ea:4d:ff:00:00:00-0x0000 Nr
```


G.8032 Ethernet Ring Protection Single Ring Topology

```

012 pdu          PEND : 0x0000 Nr          Rx<-- Blk Fwd 000 17:46:12.890
          PEND :          ----- Fwd Fwd 000 17:46:12.890
013 pdu A: ea:4d:ff:00:00:00-0x0000 Nr
          PEND :          Rx<-- Fwd Fwd 000 17:46:12.890
014 xWtr          IDLE : 0x0080 Nr(RB )      TxF-> Blk Fwd 000 17:47:12.030
015 pdu B: ea:4d:ff:00:00:00-0xb000 Sf
          IDLE : 0x0080 Nr(RB )      RxF<- Blk Fwd 000 17:49:20.650
016 pdu          PROT :          ----- Fwd Fwd 000 17:49:20.650
017 pdu B: ea:4d:ff:00:00:00-0x0000 Nr
          PROT :          Rx<-- Fwd Fwd 000 17:49:20.690
018 pdu          PEND :          ----- Fwd Fwd 000 17:49:20.690
019 pdu A: ea:4d:ff:00:00:00-0x0000 Nr
          PEND :          Rx<-- Fwd Fwd 000 17:49:20.690
000 xWtr          IDLE : 0x0080 Nr(RB )      TxF-> Blk Fwd 000 17:50:20.030
001 bDn          PROT : 0xb020 Sf          TxF-> Fwd Blk 000 17:52:31.220
002 pdu A: ea:4d:ff:00:00:00-0xb020 Sf
          PROT : 0xb020 Sf          RxF<- Fwd Blk 000 17:52:34.900
003 pdu B: ea:4d:ff:00:00:00-0x0020 Nr
          PROT : 0xb020 Sf          Rx<-- Fwd Blk 000 17:53:58.090
004 pdu A: ea:4d:ff:00:00:00-0x0020 Nr
          PROT : 0xb020 Sf          Rx<-- Fwd Blk 000 17:53:58.090
005 bUp          PEND-G: 0x0020 Nr          Tx--> Fwd Blk 000 17:53:58.930
006 pdu B: ea:4d:ff:00:00:00-0x0020 Nr
          PEND : 0x0020 Nr          Rx<-- Fwd Blk 000 17:53:59.890
007 pdu          PEND :          ----- Fwd Fwd 000 17:53:59.890
008 pdu A: ea:4d:ff:00:00:00-0x0020 Nr
          PEND :          Rx<-- Fwd Fwd 000 17:53:59.890
009 xWtr          IDLE : 0x0080 Nr(RB )      TxF-> Blk Fwd 000 17:54:59.030

```

*A:PE-2#

Conclusion

Ethernet Ring APS provides optimal solution for designing native Ethernet services with ring topology. This protocol provides simple configuration, operation and guaranteed fast protection time. 7x50 also has a flexible encapsulation that allows dot1Q, qinq or PBB for the ring traffic. It could be utilized for various services such as mobile backhaul, business VPN access, aggregation and core.

Services: Layer 2 and EVPN

In This Section

This section provides configuration information for the following topics:

- [BGP Multi-Homing for VPLS Networks on page 825](#)
- [BGP VPLS on page 865](#)
- [BGP Virtual Private Wire Services on page 905](#)
- [EVPN for MPLS Tunnels on page 937](#)
- [EVPN for PBB over MPLS \(PBB-EVPN\) on page 991](#)
- [EVPN for VXLAN Tunnels \(Layer 2\) on page 1033](#)
- [EVPN for VXLAN Tunnels \(Layer 3\) on page 326](#)
- [Inter-AS Model C for VLL on page 1103](#)
- [LDP VPLS using BGP-Auto Discovery on page 1127](#)
- [Multi-Chassis Endpoint for VPLS Active/Standby Pseudowire on page 1159](#)
- [Multi-Segment Pseudowire Routing on page 1189](#)
- [PBB-Epipe on page 1243](#)
- [PBB-VPLS on page 1269](#)
- [Shortest Path Bridging for MAC on page 1311](#)

BGP Multi-Homing for VPLS Networks

In This Chapter

This section describes BGP Multi-Homing (BGP-MH) for VPLS network configurations.

Topics in this section include:

- [Applicability on page 826](#)
- [Summary on page 827](#)
- [Overview on page 829](#)
- [Configuration on page 831](#)
- [Conclusion on page 864](#)

Applicability

This section is applicable to all of the 7750 SR series (SR-7, SR-12, SR-c4 and SR-c12), as well as 7450 ESS (ESS-7 and ESS-12) series in mixed mode. It was tested on release 13.0.R1.

Summary

The SR/ESS portfolio supports the use of Border Gateway Protocol Multi-Homing for VPLS (hereafter called BGP-MH). BGP-MH is described in draft-ietf-l2vpn-vpls-multihoming, *BGP based Multi-homing in Virtual Private LAN Service*, and provides a network-based resiliency mechanism (no interaction from the PEs Provider Edge router — to MTU/CEs Multi-Tenant Unit/ Customer Equipment) that can be applied on access Service Access Points (SAPs) or network (pseudowires) topologies. The BGP-MH procedures will run between the PEs and will provide a loop-free topology from the network perspective (only one logical active path will be provided per VPLS among all the objects SAPs or pseudowires which are part of the same Multi-Homing site).

Each multi-homing site connected to two or more peers is represented by a site-id (2-bytes long) which is encoded in the BGP MH Network Layer Reachability Information (NLRI). The BGP peer holding the active path for a particular multi-homing site will be named as the Designated Forwarder (DF), whereas the rest of the BGP peers participating in the BGP MH process for that site will be named as non-DF and will block the traffic (in both directions) for all the objects belonging to that multi-homing site.

BGP MH uses the following rules to determine which PE is the DF for a particular multi-homing site:

1. A BGP MH NLRI with D flag = 0 (multi-homing object up) always takes precedence over a BGP MH NLRI with D flag = 1 (multi-homing object down). If there is a tie, then:
2. The BGP MH NLRI with the highest BGP LP (Local Preference) wins. If there is a tie, then:
3. The BGP MH NLRI issued from the PE with the lowest PE ID (system address) wins.

The main advantages of using BGP-MH as opposed to other resiliency mechanisms for VPLS are:

- Flexibility: BGP-MH uses a common mechanism for access and core resiliency. The designer has the flexibility of using BGP-MH to control the active/standby status of SAPs, spoke SDPs, Split Horizon Groups (SHGs) or even mesh SDP bindings.
- The standard protocol is based on BGP, a standard, scalable and well-known protocol.
- Specific benefits at the access:
 - It is network-based, independent of the customer CE and, as such, it does not need any customer interaction to determine the active path. Consequently the operator will spend less effort on provisioning and will minimize both operation costs and security risks (in particular, this removes the requirement for spanning tree interaction between the PE and CE).
 - Easy load balancing per service (no service fate-sharing) on physical links.

- Specific benefits in the core:
 - It is a network-based mechanism, independent of the MTU resiliency capabilities and it does not need MTU interaction, therefore operational advantages are achieved as a result of the use of BGP-MH: less provisioning is required and there will be minimal risks of loops. In addition, simpler MTUs can be used.
 - Easy load balancing per service (no service fate-sharing) on physical links.
 - Less control plane overhead: there is no need for an additional protocol running the pseudowire redundancy when BGP is already used in the core of the network. BGP-MH just adds a separate NLRI in the L2-VPN family (AFI=25, SAFI=65).

The objective of this section is to provide the required guidelines to configure and troubleshoot BGP-MH for VPLS

Knowledge of the LDP/BGP VPLS (RFC 4762, *Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling*, and RFC 4761, *Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling*) architecture and functionality is assumed throughout this document. For further information refer to the relevant Alcatel-Lucent documentation.

Overview

The following network setup will be used throughout the rest of the section.

Note:

- IGP — ISIS, Level 2 on all routers; area 49.0001
- RSVP-TE for transport tunnels
- FRR in the core
- No protection at the access.

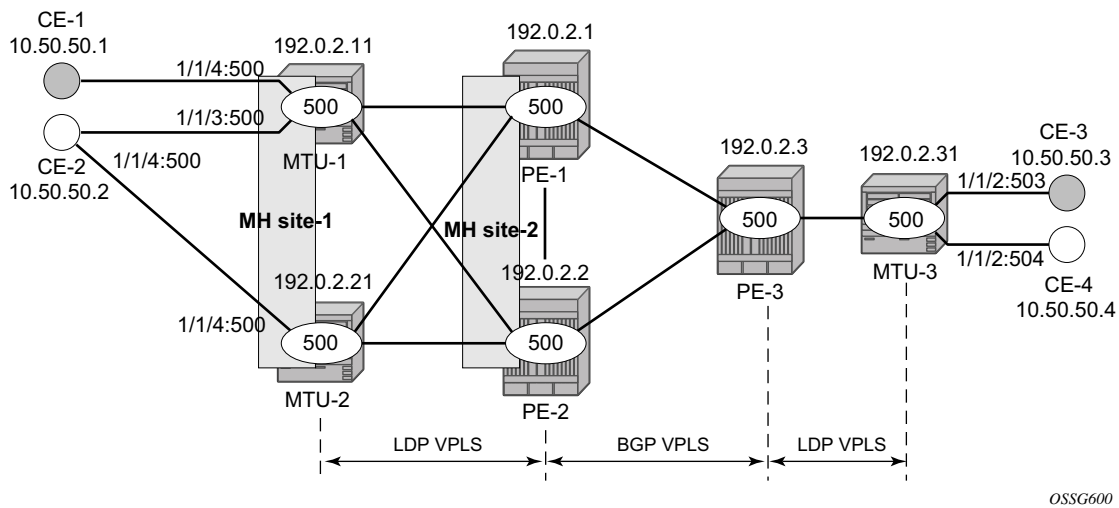


Figure 133: Network Topology

The setup consists of a three core nodes (PE-1, PE-2 and PE-3) and three Multi-Tenant Unit (MTU) nodes connected to the core. The VPLS service 500 is configured on all the six nodes with the following characteristics:

The VPLS service 500 is configured on all the six nodes with the following characteristics:

- The core VPLS instances are connected by a full mesh of BGP-signaled pseudowires (that is, pseudowires among PE-1, PE-2 and PE-3 will be signaled by BGP VPLS).
- As depicted in the [Figure 133](#), the MTUs are connected to the BGP VPLS core by TLDP pseudowires. While MTU-3 is connected to PE-3 by a single pseudowire, MTU-1 and

MTU-2 are dual-homed to PE-1 and PE-2. The following resiliency mechanisms are used on the dual-homed MTUs:

- MTU-1 is dual-connected to PE-1 and PE-2 by an active/standby pseudowire (A/S pseudowire hereafter).
- MTU-2 is dual-connected to PE-1 and PE-2 by two active pseudowires, one of them being blocked by BGP MH running between PE-1 and PE-2. The PE-1 and PE-2 pseudowires, set up from MTU-2, will be part of the BGP MH site MH-site-2.
- MTU-1 and MTU-2 are running BGP MH, being SHG site-1 and sap 1/1/4:500 on MTU-2 part of the same BGP MH site, MH-site-1.
- The CEs are connected to the network in the following way:
 - CE-1, CE-3 and CE-4 are single-connected to the network
 - CE-2 is dual connected to MTU-1 and MTU-2.
 - CE-1 and CE-2 are part of the split-horizon-group (SHG) site-1(SAPs 1/1/4:500 and 1/1/3:500 on MTU-1). Assume that CE-1 and CE-2 have a backdoor link between them so that when MTU-2 is elected as DF, CE1 does not get isolated. This configuration high-lights the use of a SHG within a site configuration.

For each BGP MH site, MH-site-1 and MH-site-2, the BGP MH process will elect a DF, blocking the site objects for the non-DF nodes. In other words, based on the specific configuration explained throughout the section:

- For MH-site-1, MTU-1 will be elected as the DF. The non-DF-MTU-2 will block the SAP 1/1/4:500.
- For MH-site-2, PE-1 will be elected as the DF. The non-DF PE-1 will block the spoke-SDP to MTU-2.

Configuration

This section describes all the relevant configuration tasks for the setup shown in [Figure 133](#). Note that the appropriate associated IP/MPLS configuration is out of the scope of this section. In this particular example the following protocols will be configured beforehand:

- ISIS-TE as IGP with all the interfaces being level-2 (OSPF-TE could have been used instead).
- RSVP-TE as the MPLS protocol to signal the transport tunnels (LDP could have been used instead).
- LSPs between core PEs will be Fast Re-Route protected (facility bypass tunnels) whereas LSP tunnels between MTUs and PEs will not be protected¹.

Once the IP/MPLS infrastructure is up and running, the specific service configuration including the support for BGP MH can begin.

Global BGP Configuration

BGP is used in this configuration guide for these purposes:

- a.** Auto-discovery and signaling of the pseudowires in the core, as per RFC 4761.
- b.** Exchange of multi-homing site NLRIs and redundancy handling from MTU-2 to the core.
- c.** Exchange of multi-homing site NLRIs and redundancy handling at the access for CE-1/CE-2.

A BGP route reflector (RR), PE-3, is used for the reflection of BGP updates corresponding to the above uses **a** and **b**.

A direct peering is established between MTU-1 and MTU-2 for use **c**. Note that the same RR could have been used for the three cases, however, like in this example, the designer may choose to have a direct BGP peering between access devices. The reasons for this are:

- By having a direct BGP peering between MTU-1 and MTU-2, the BGP updates do not have to travel back and forth.

1. Note that the designer can choose whether to protect access link failures by means of MPLS FRR or A/S pseudowire or BGP MH. While FRR provides a faster convergence (around 50ms) and stability (it does not impact on the service layer, hence, link failures do not trigger MAC flush and flooding) some interim inefficiencies can be introduced compared to A/S pseudowire or BGP MH.

Global BGP Configuration

- On MTU-1 and MTU-2, BGP is exclusively used for multi-homing, therefore there will not be more BGP peers for either MTUs and a RR adds nothing in terms of control plane scalability.

On all nodes, the autonomous-system number must be configured.

```
configure router autonomous-system 65000
```

In this example, the router-id is equal to the system address.

```
*A:PE-1# configure router router-id 192.0.2.1
```

```
*A:MTU-1# configure router router-id 192.0.2.11
```

The following CLI output shows the global BGP configuration required on MTU-1. Note that the 192.0.2.21 address will be replaced by the corresponding peer or the RR system address for PE-1 and PE-2.

```
*A:MTU-1# configure router bgp
    family 12-vpn
    router-id 192.0.2.11
    rapid-withdrawal
    rapid-update 12-vpn
    group "Multi-Homing"
        neighbor 192.0.2.21
        type internal
    exit
exit
no shutdown
```

In this example, PE-3 is the BGP RR, therefore its BGP configuration will contain a cluster with all its peers included (PE-1 and PE-2):

```
*A:PE-3# configure router bgp
    family 12-vpn
    router-id 192.0.2.3
    rapid-withdrawal
    rapid-update 12-vpn
    group "internal"
        cluster 1.1.1.1
        neighbor 192.0.2.1
        type internal
    exit
    neighbor 192.0.2.2
    type internal
    exit
exit
no shutdown
```

The relevant BGP commands for BGP-MH are in bold. Some considerations about those:

- It is required to specify **family l2-vpn** in the BGP configuration. That statement will allow the BGP peers to agree on the support for the family AFI=25 (Layer 2 VPN), SAFI=65 (VPLS). This family is used for BGP VPLS as well as for BGP MH and BGP AD.
- The **rapid-update l2-vpn** statement allows BGP MH to send BGP updates immediately after detecting link failures, without having to wait for the Minimum Route Advertisement Interval (MRAI) to send the updates in batches. This statement is required to guarantee a fast convergence for BGP MH.
- Optionally, rapid-withdrawal can also be added. Note that, in the context of BGP MH, this command is only useful if a particular multi-homing site is cleared². In that case, a BGP withdrawal is sent immediately without having to wait for the MRAI.

2. This means removing the BGP-MH site or even to remove the whole VPLS service.

Service Level Configuration

Once the IP/MPLS infrastructure is configured, including BGP, this section shows the configuration required at service level (VPLS 500). The focus is on the nodes involved on BGP MH, that is, MTU-1, MTU-2, PE-1 and PE-2. These nodes are highlighted in [Figure 134](#).

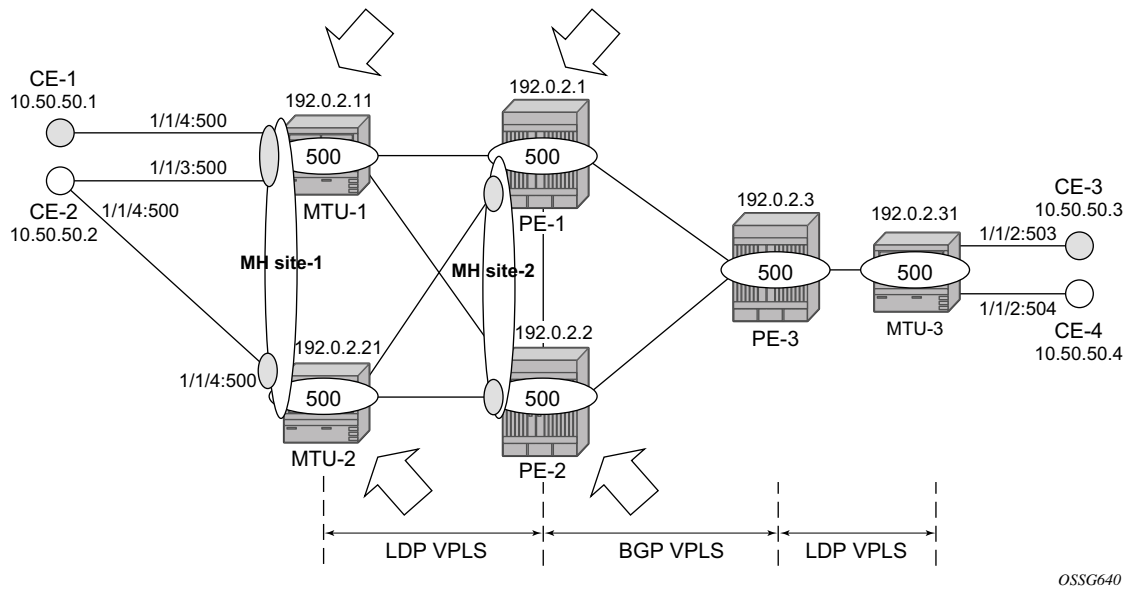


Figure 134: Nodes Involved in BGP MH

Core PE Service Configuration

The following CLI excerpt shows the service level configuration on PE-1.

```
*A:PE-1# configure service
-----
sdp 111 mpls create
  far-end 192.0.2.11
  lsp "LSP-PE-1-MTU-1"
  path-mtu 8000
  keep-alive
  shutdown
exit
no shutdown
exit
sdp 121 mpls create
  far-end 192.0.2.21
  lsp "LSP-PE-1-MTU-2"
  path-mtu 8000
  keep-alive
  shutdown
exit
no shutdown
exit
sdp 1212 mpls create
  description "SDP to transport BGP-signaled PWs"
  signaling bgp
  far-end 192.0.2.2
  lsp "LSP-PE-1-PE-2"
  path-mtu 8000
  keep-alive
  shutdown
exit
no shutdown
exit
sdp 1313 mpls create
  description "SDP to transport BGP-signaled PWs"
  signaling bgp
  far-end 192.0.2.3
  lsp "LSP-PE-1-PE-3"
  path-mtu 8000
  keep-alive
  shutdown
exit
no shutdown
exit
customer 1 create
  description "Default customer"
exit
pw-template 500 use-provisioned-sdp create
exit
vpls 500 customer 1 create
  bgp
    route-distinguisher 65000:501
    vsi-export "vsi500_export"
    vsi-import "vsi500_import"
    pw-template-binding 500 split-horizon-group "CORE"
```

Core PE Service Configuration

```
        exit
    exit
    bgp-vpls
        max-ve-id 65535
        ve-name "501"
        ve-id 501
    exit
    no shutdown
exit
stp
    shutdown
exit
site "MH-site-2" create
    site-id 2
    spoke-sdp 121:500
    no shutdown
exit
spoke-sdp 111:500 create
    no shutdown
exit
spoke-sdp 121:500 create
    no shutdown
exit
no shutdown
exit
```


The following are general comments about the configuration of service 500:

- As seen in the CLI output above for PE-1, there are four provisioned SDPs that the service VPLS 500 will use in this example. SDP 111 and SDP 121 are tunnels over which the TLDP FEC128 pseudowires for service 500 will be carried (according to RFC 4762), whereas SDP 1212 and SDP 1313 are the tunnels for the core BGP pseudowires (based on RFC 4761).
- The BGP context provides the general service BGP configuration that will be used by BGP VPLS and BGP MH:
 - Route distinguisher (notation chosen is based on <AS_number:500 + node_id>)
 - VSI export policies are used to add the export route-targets included in all the BGP updates sent to the BGP peers.
 - VSI import policies are used to control the NLRIs accepted in the RIB, normally based on the route-targets.
 - Both, VSI-export and VSI-import policies can be used to modify attributes like the Local-Preference (LP) that will be used to influence the BGP MH Designated Forwarder (DF) election (LP is the second rule in the BGP MH election process, as previously discussed). The use of these policies will be described later in the section.
 - The **pw-template-binding** command maps the previously defined pw-template 500 to the **split-horizon-group CORE**. In this way, all the BGP-signaled pseudowires will be part of this split-horizon-group. Although not shown in this example, the **pw-template-binding** command can also be used to instantiate pseudowires within different split-horizon-groups, based on different import route-targets³.

```
*A:PE-1# configure service vpls 500 bgp pw-template-binding ?
  - pw-template-binding <policy-id> [split-horizon-group <group-name>]
                                     [import-rt {ext-community,...(upto 5 max)}]
  - no pw-template-binding <policy-id>

---snip---

*A:PE-1#
```

- The BGP-signaled pseudowires (from PE-1 to PE-2 and PE-3) are set up according to the configuration in the BGP context. Beside those pseudowires, the VPLS 500 also has two more pseudowires signaled by TLDP: spoke-sdp 111:500 (to MTU-1) and spoke-sdp 121:500 (to MTU-2).

3. Detailed BGP-VPLS configuration is out of the scope of this configuration guide. For more information refer to the Alcatel-Lucent SR OS documentation.

The general BGP MH configuration parameters for a particular multi-homing site are shown in the following output:

```
*A:PE-1# configure service vpls ?
  - no vpls <service-id>
  - vpls <service-id> [customer <customer-id>] [create] [vpn <vpn-id>] [m-vpls] [b-vpls|i-
vpls]
    [etree]

---snip---

*A:PE-1# configure service vpls 500 site ?
  - no site <name>
  - site <name> [create]

<name>                : [32 chars max]

[no] boot-timer        - Configure/Override site boot-timer
    failed-thresho*    - Configure threshold for the site to be declared down
[no] mesh-sdp-bindi*   - Enable/Disable application to all Mesh-SDP
[no] monitor-oper-g*   - Configure an Operational-Group to monitor
[no] sap               - Configure a SAP for the site
[no] shutdown          - Administratively enable/disable the site
[no] site-activatio*   - Configure/Override site activation timer
[no] site-id           - Configure site identifier
[no] site-min-down-*   - Configure minimum down timer for the site
[no] split-horizon-*   - Configure a split-horizon-group
[no] spoke-sdp         - Configure a spoke-SDP
```

Where:

- The site-name is defined by a string of up to 32 characters.
- The site-id is an integer that identifies the multi-homing site and is encoded in the BGP MH NLRI. This ID must be the same one used on the peer node where the same multi-homing site is connected to. That is, MH-site-2 must use the same site-id in PE-1 and PE-2 (value = 2 in the PE-1 site configuration).
- Out of the four potential objects in a site, spoke SDP, SAP, SHG and mesh SDP binding only one can be used at the time on a particular site. To add more than just one sap/spoke-sdp to the same site, a split-horizon-group composed of the sap/spoke-sdp objects must be used in the site configuration. Otherwise, only one object (spoke SDP, SAP, SHG and mesh SDP binding) is allowed per site. A CLI log message warns the operator of such fact:

```
*A:PE-1>config>service>vpls>site# mesh-sdp-binding
MINOR: SVCNMR #5855 only one object is allowed per site
```

- The **failed-threshold** command defines how many objects should be down for the site to be declared down. This command is obviously only valid for multi-object sites (split-

horizon-groups and mesh-sdp-bindings). By default, all the objects in a site must be down for the site to be declared as operationally down.

```
*A:PE-1>config>service>vpls>site# failed-threshold ?
- failed-threshold <[1..1000]>
- failed-threshold all
```

- The **boot-timer** specifies for how long the service manager waits after a node reboot before running the MH Procedures. The boot-timer value should be configured to allow for the BGP sessions to come up and for the NLRI information to be refreshed/exchanged. In environments with the default BGP MRAI (30 seconds) it is highly recommended to increase this value (for instance, 120 seconds for a normal configuration). The **boot-timer** is only important when a node comes back up and would become the DF. Default value: 10 seconds.

```
*A:PE-1>config>service>vpls>site# boot-timer ?
- boot-timer <seconds>
- no boot-timer

<seconds> : [0..600]
```

- The **site-activation-timer** command defines the amount of time the service manager will keep the local objects in standby (in the absence of BGP updates from remote PEs) before running the DF election algorithm to decide whether the site should be unblocked. The timer is started when one of the following events occurs only if the site is operationally up:

- Manual site activation using the **no shutdown** command at the site-id level or at member object(s) level (SAP(s) or pseudowire(s))
- Site activation after a failure

The BGP MH election procedures will be resumed upon expiration of this timer or the arrival of a BGP MH update for the multi-homing site. Default value: 2 seconds.

- When a BGP MH site goes down, it may be preferred that it stays down for a minimum time. This is configurable by the **site-min-down-timer**. When set to zero, this timer is disabled.

```
*A:PE-1>config>service>vpls>site# site-min-down-timer
- no site-min-down-timer
- site-min-down-timer <seconds>

<seconds> : [0..100]
```

- The **boot-timer**, **site-activation-timer** and **site-min-down-timer** commands can be provisioned at service level or at global level. The service level settings have precedence and override the global configuration. The **no** form of the commands at global level, sets the value back to the default values. The **no** form of the commands at service level, makes the timers inherit the global values.

```
*A:PE-1# configure redundancy bgp-multi-homing
- bgp-multi-homing
```

```
[no] boot-timer      - Configure BGP multi-homing boot-timer
[no] site-activation - Configure BGP multi-homing site activation timer
[no] site-min-down-* - Configure minimum down timer for the site
```

- The **shutdown** command controls the admin state of the site. Note that each site has three possible states:
 - Admin state — controlled by the shutdown command.
 - Operational state — controlled by the operational status of the individual site objects.
 - Designated-Forwarder (DF) state — controlled by the BGP MH election algorithm.

The following CLI output shows the three states for a BGP MH site:

```
*A:MTU-2# show service id 500 site "MH-site-1"
=====
Site Information
=====
Site Name           : MH-site-1
-----
Site Id             : 1
Dest                : sap:1/1/4:500      Mesh-SDP Bind      : no
Admin Status        : Enabled            Oper Status        : up
Designated Fwdr     : No
DF UpTime           : 0d 00:00:00        DF Chg Cnt        : 1
Boot Timer          : default             Timer Remaining    : 0d 00:00:00
Site Activation Timer: default             Timer Remaining    : 0d 00:00:00
Min Down Timer       : default             Timer Remaining    : 0d 00:00:00
Failed Threshold     : default(all)
Monitor Oper Grp     : (none)
=====
*A:MTU-2#
```

For this example, configure the site MH-site-2 in PE-1, where the site-id is 2 and the object in the site is spoke-sdp 121:500 (pseudowire established from PE-1 to MTU-2).

The following CLI output shows the service configuration for PE-2. Note that the site-id is 2, that is, the same value configured in PE-1. The object defined in PE-2's site is spoke-sdp 221:500 (pseudowire established from PE-2 to MTU-2).

```
*A:PE-2# configure service
      sdp 211 mpls create
        far-end 192.0.2.11
        lsp "LSP-PE-2-MTU-1"
        path-mtu 8000
        keep-alive
        shutdown
      exit
      no shutdown
    exit
    sdp 221 mpls create
      far-end 192.0.2.21
      lsp "LSP-PE-2-MTU-2"
      path-mtu 8000
      keep-alive
```

```

        shutdown
    exit
    no shutdown
exit
sdp 2121 mpls create
    description "SDP to transport BGP-signaled PWs"
    signaling bgp
    far-end 192.0.2.1
    lsp "LSP-PE-2-PE-1"
    path-mtu 8000
    keep-alive
        shutdown
    exit
    no shutdown
exit
sdp 2323 mpls create
    description "SDP to transport BGP-signaled PWs"
    signaling bgp
    far-end 192.0.2.3
    lsp "LSP-PE-2-PE-3"
    path-mtu 8000
    keep-alive
        shutdown
    exit
    no shutdown
exit
customer 1 create
    description "Default customer"
exit
pw-template 500 use-provisioned-sdp create
exit
vpls 500 customer 1 create
    bgp
        route-distinguisher 65000:502
        vsi-export "vsi500_export"
        vsi-import "vsi500_import"
        pw-template-binding 500 split-horizon-group "CORE"
    exit
    exit
    bgp-vpls
        max-ve-id 65535
        ve-name "502"
        ve-id 502
    exit
    no shutdown
    exit
    stp
        shutdown
    exit
    site "MH-site-2" create
        site-id 2
        spoke-sdp 221:500
        no shutdown
    exit
    spoke-sdp 211:500 create
        no shutdown
    exit
    spoke-sdp 221:500 create
        no shutdown

```

Core PE Service Configuration

```
exit
no shutdown
exit
```

MTU Service Configuration

The following CLI output shows the service level configuration on MTU-1.

```
*A:MTU-1# configure service
      sdp 111 mpls create
        far-end 192.0.2.1
        lsp "LSP-MTU-1-PE-1"
        path-mtu 8000
        keep-alive
        shutdown
      exit
      no shutdown
    exit
  sdp 112 mpls create
    far-end 192.0.2.2
    lsp "LSP-MTU-1-PE-2"
    path-mtu 8000
    keep-alive
    shutdown
  exit
  no shutdown
exit
customer 1 create
  description "Default customer"
exit
vpls 500 customer 1 create
  split-horizon-group "site-1" create
  exit
  bgp
    route-distinguisher 65000:511
    route-target export target:65000:500 import target:65000:500
  exit
  stp
    shutdown
  exit
  site "MH-site-1" create
    site-id 1
    split-horizon-group site-1
    no shutdown
  exit
  endpoint "CORE" create
    no suppress-standby-signaling
  exit
  sap 1/1/3:500 split-horizon-group "site-1" create
    eth-cfm
      mep 511 domain 1 association 1 direction down
      fault-propagation-enable use-if-tlv
      ccm-enable
      no shutdown
    exit
  exit
exit
sap 1/1/4:500 split-horizon-group "site-1" create
exit
spoke-sdp 111:500 endpoint "CORE" create
stp
```

MTU Service Configuration

```
        shutdown
    exit
    precedence primary
    no shutdown
exit
spoke-sdp 112:500 endpoint "CORE" create
    stp
        shutdown
    exit
    no shutdown
exit
no shutdown
exit
```

The MTU-1 node is configured with the following characteristics:

- The BGP context provides the general BGP parameters for service 500 in MTU-1. Note that the route-target command is now used instead of the vsi-import and vsi-export commands. The intent in this example is to configure only the export and import route-targets. There is no need to modify any other attribute. If the local preference is to be modified (to influence the DF election), a vsi-policy must be configured.
- An A/S pseudowire configuration is used to control the pseudowire redundancy towards the core.
- The multi-homing site, MH-site-1 has a site-id = 1 and a split-horizon-group as an object. The split-horizon-group site-1 is composed of sap 1/1/3:500 and sap 1/1/4:500. As previously discussed, the site will not be declared operationally down until the two SAPs belonging to the site are down. This behavior can be changed by the **failed-threshold** command (for instance, in order to bring the site down when only one object has failed even though the second SAP is still up).
- Note that, as an example, a Y.1731 MEP with fault-propagation has been defined in SAP 1/1/3:500. As discussed later in the section, this MEP will signal the status of the SAP (as a result of the BGP MH process) to CE-2.

The service configuration in MTU-2 is shown below.

```
*A:MTU-2# configure service
  sdp 211 mpls create
    far-end 192.0.2.1
    lsp "LSP-MTU-2-PE-1"
    path-mtu 8000
    keep-alive
    shutdown
  exit
  no shutdown
exit
sdp 212 mpls create
  far-end 192.0.2.2
  lsp "LSP-MTU-2-PE-2"
  path-mtu 8000
  keep-alive
  shutdown
exit
  no shutdown
exit
customer 1 create
  description "Default customer"
exit
vpls 500 customer 1 create
  bgp
    route-distinguisher 65000:521
    route-target export target:65000:500 import target:65000:500
  exit
  stp
    shutdown
  exit
  site "MH-site-1" create
    site-id 1
    sap 1/1/4:500
    no shutdown
  exit
  sap 1/1/4:500 create
  exit
  spoke-sdp 211:500 create
    no shutdown
  exit
  spoke-sdp 212:500 create
    no shutdown
  exit
  no shutdown
exit
```

Influencing the Designated Forwarder (DF) Decision

As previously explained, assuming that the sites on the two nodes taking part of the same multi-homing site are both up, the two tie-breakers for electing the DF are (in this order):

1. Highest LP
2. Lowest PE ID

The LP by default is 100 in all the routers. Under normal circumstances, if the LP in any router is not changed, MTU-1 will be elected the DF for MH-site-1, whereas PE-1 will be the DF for MH-site-2. Assume in this section that this behavior is changed for MH-site-2 to make PE-2 the DF. Since changing the system address (to make PE-2's ID the lower of the two IDs) is usually not an easy task to accomplish, a vsi-export policy in PE-2 is added to modify the LP with which the MH-site-2 NLRI is announced to PE-1. That policy will change the LP to 150 and as such, since 150 is greater than the default 100 in PE-1, PE-2 will be elected as the DF for MH-site-2. The configuration of the policy is outlined below:

```
*A:PE-2# admin display-config

---snip---

#-----
echo "Service Configuration"
#-----
    service

---snip---

    vpls 500 customer 1 create
        bgp
            route-distinguisher 65000:502
            vsi-export "vsi500_export"
            vsi-import "vsi500_import"
            pw-template-binding 500 split-horizon-group "CORE"
            exit
        exit

---snip---

*A:PE-2# configure router policy-options
begin
policy-statement "vsi500_export"
    entry 10
        action accept
            community add "comm_core"
            local-preference 150
        exit
    exit
exit
policy-statement "vsi500_import"
```

```

        entry 10
        from
            community "comm_core"
            family l2-vpn
        exit
        action accept
        exit
    exit
    default-action reject
exit
commit
exit all

```

In PE-1, simply import and export the same route-target without modifying the LP. The configuration is shown below.

```

*A:PE-1# admin display-config

---snip---
#-----
echo "Service Configuration"
#-----
    service

---snip---

    vpls 500 customer 1 create
        bgp
            route-distinguisher 65000:501
            vsi-export "vsi500_export"
            vsi-import "vsi500_import"
            pw-template-binding 500 split-horizon-group "CORE"
            exit
        exit

---snip---
*A:PE-1# configure router policy-options
begin
community "comm_core" members "target:65000:500"
policy-statement "vsi500_export"
    entry 10
        action accept
        community add "comm_core"
    exit
exit
policy-statement "vsi500_import"
    entry 10
        from
            community "comm_core"
            family l2-vpn
        exit
        action accept
        exit
    exit
    default-action reject
exit

```

commit

Note that the policy is applied at service 500 level, which means that the LP changes for all the potential multi-homing sites configured under service 500. Therefore, load balancing can be achieved on a per-service basis, but not within the same service.

Another important remark is that these policies are applied on the VPLS 500 for all the potential BGP applications, that is, BGP VPLS, BGP MH and BGP AD. In the example, the LP for the PE-2 BGP updates for BGP MH and BGP VPLS will be set to 150 (note that this however has no impact on BGP VPLS since a given PE cannot receive two BGP VPLS NLRI with the same VE-ID, such as a different VE-ID per PE within the same VPLS is required).

Black-Hole Avoidance

The 7x50 supports the appropriate MAC flush mechanisms for BGP MH, regardless of the protocol being used for the pseudowire signaling:

- LDP VPLS — The PE that contains the old DF site (the site that just experienced a DF→non-DF transition) always sends a LDP MAC **flush-all-from-me** to all LDP pseudowires in the VPLS, including the LDP pseudowires associated with the new DF site. No specific configuration is required.
- BGP VPLS — The remote BGP VPLS PEs interpret the F bit transitions from 1 to 0 as an implicit MAC flush-all-from-me indication. If a BGP update with the flag F=0 is received from the previous DF PE, the remote PEs perform MAC flush-all-from-me, flushing all the MACs associated with the pseudowire to the old DF PE. No specific configuration is required.

Double flushing will not happen as it is expected that between any pair of PEs there will exist only one type of pseudowires, either BGP or LDP pseudowire, but not both types.

In the example, assuming MTU-1 and PE-1 are the DF nodes:

- When MH-site-1 is brought operationally down on MTU-1 (so by default, the two SAPs must go down unless the **failed-threshold** parameter is changed so that the site is down when only one SAP is brought down), MTU-1 will issue a **flush-all-from-me** message.
- When MH-site-2 is brought operationally down on PE-1, a BGP update with F=0 and D=1 is issued by PE-1. PE-2 and PE-3 will receive the update and will flush the MAC addresses learned on the pseudowire to PE-1.

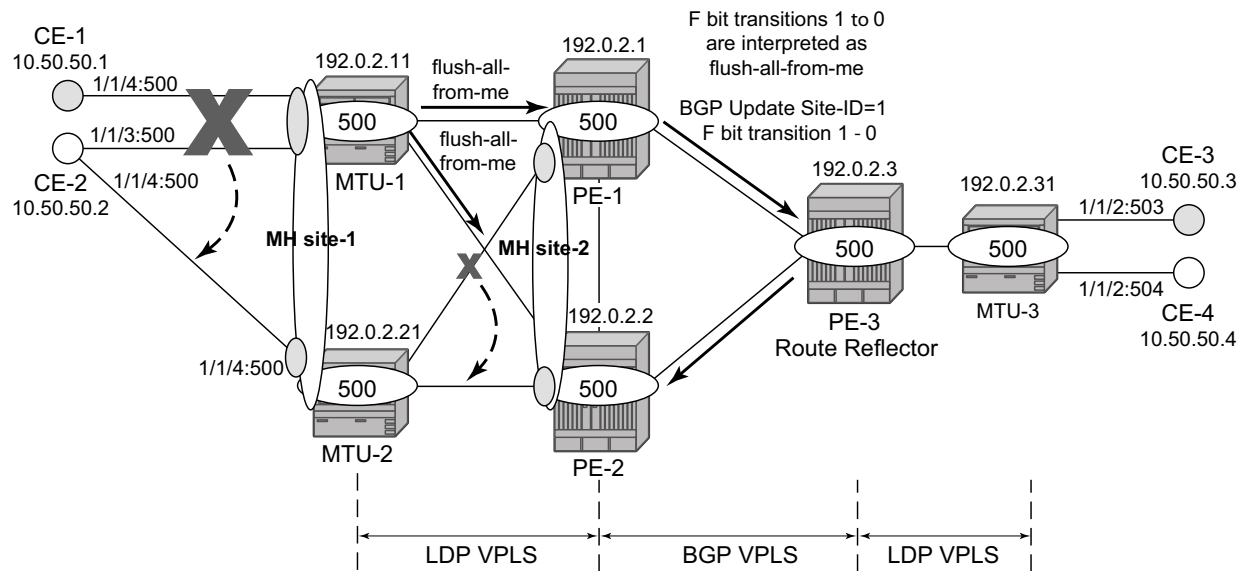


Figure 135: MAC Flush for BGP MH

Node failures implicitly trigger a MAC flush on the remote nodes, since the TLDP/BGP session to the failed node goes down.

Access CE/PE Signaling

BGP MH works at service level, therefore no physical ports are torn down on the non-DF but rather the objects are brought down operationally, while the physical port will stay up and used for any other services existing on that port. Due to this reason, there is a need for signaling the standby status of an object to the remote PE or CE.

- Access PEs running BGP MH on spoke SDPs and elected non-DF, will signal pseudowire standby status (0x20) to the other end. If no pseudowire status is supported on the remote MTU, a label withdrawal is performed⁴. If there is more than one spoke SDP on the site (part of the same SHG), the signaling is sent for all the pseudowires of the site.
- Multi-homed CEs connected through SAPs to the PEs running BGP MH, are signaled by the PEs using Y.1731 CFM, either by stopping the transmission of CCMs or by sending CCMs with isDown (interface status down encoding in the interface status TLV).

In this example, down MEPs on MTU-1 SAP 1/1/3:500, SAP 1/1/4:500 and MTU-2 SAP 1/1/4:500 are configured. [Figure 136](#) shows only the MEPs on MTU-1 SAP 1/1/3:500 and CE-2. Upon failure on the MTU-1 site MH-site-1 the MEP 1 will start sending CCMs with interface status down.

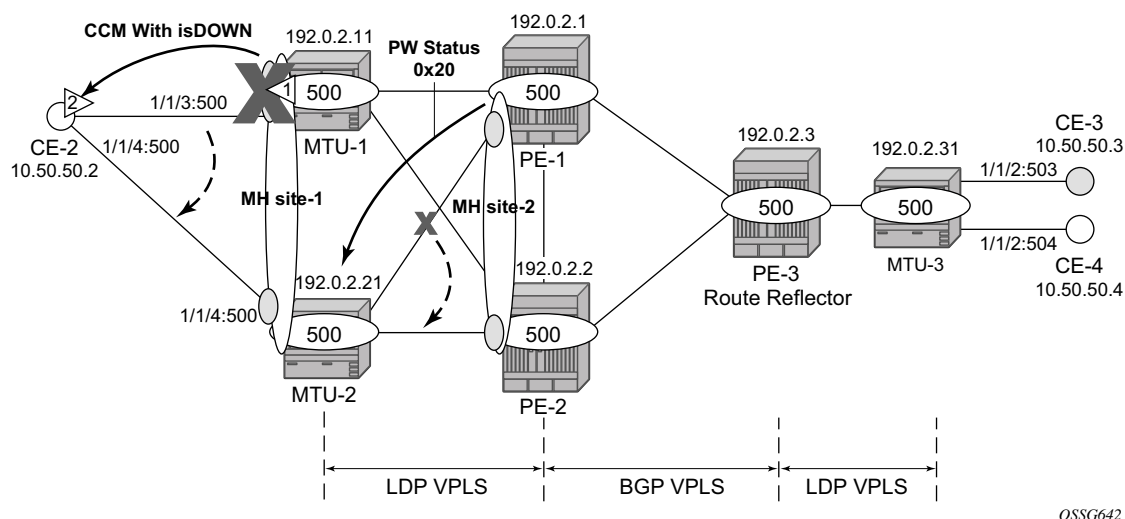


Figure 136: Access PE/CE Signaling

4. Note that the **configure service vpls x spoke-sdp y:z no pw-status-signaling** parameter allows to send a TLDP label-withdrawal instead of pseudowire status bits, even though the peer supports pseudowire status.

The CFM configuration required at SAP 1/1/3:500 is outlined below⁵. Down MEPs will be configured on CE-2 and MTU-2 SAPs in the same way, but in a different association. Note that **fault-propagation-enable use-if-tlv** must be added. In case the CE does not understand the CCM interface status TLV, the **fault-propagation-enable suspend-ccm** option can be enabled instead. This will stop the transmission of CCMs upon site failures.

```
*A:MTU-1# configure eth-cfm
    domain 1 format none level 3
    association 1 format icc-based name "vpls000000500"
        bridge-identifier 500
        exit
        ccm-interval 1
        remote-mepid 5023
    exit
exit

*A:MTU-1# configure service

---snip---
    vpls 500 customer 1 create
        split-horizon-group "site-1" create
        exit
        bgp
            route-distinguisher 65000:511
            route-target export target:65000:500 import target:65000:500
        exit
        stp
            shutdown
        exit
        site "MH-site-1" create
            site-id 1
            split-horizon-group site-1
            no shutdown
        exit
        endpoint "CORE" create
            no suppress-standby-signaling
        exit
        sap 1/1/3:500 split-horizon-group "site-1" create
            eth-cfm
                mep 511 domain 1 association 1 direction down
                fault-propagation-enable use-if-tlv
                ccm-enable
                no shutdown
            exit
        exit
    exit

---snip---
```

5. Detailed configuration guidelines for Y.1731 are out of the scope of this configuration guide.

If CE-2 is a service router, upon receiving a CCM with isDown, an alarm will be triggered and the SAP will be brought down:

```
173 2015/03/27 14:32:53.78 UTC WARNING: SNMP #2004 vprn502 int-CE-2-MTU-1
"Interface int-CE-2-MTU-1 is not operational"

172 2015/03/27 14:32:53.78 UTC MINOR: SVCMGR #2203 vprn502
"Status of SAP 1/2/3:500 in service 502 (customer 1) changed to admin=up oper=down
flags=OamDownMEPFault "

171 2015/03/27 14:32:53.78 UTC MINOR: SVCMGR #2108 vprn502
"Status of interface int-CE-2-MTU-1 in service 502 (customer 1) changed to admin=up
oper=down"

170 2015/03/27 14:32:53.78 UTC MINOR: ETH_CFM #2001 Base
"MEP 1/1/5023 highest defect is now defRemoteCCM "

On MTU-1, the status of the SAP can be checked:
*A:MTU-1# show service id 500 sap 1/1/3:500
```

```
=====
Service Access Points(SAP)
=====
Service Id      : 500
SAP             : 1/1/3:500          Encap             : q-tag
Description     : (Not Specified)
Admin State    : Up                  Oper State       : Down
Flags          : PortOperDown
                OamDownMEPFault
Multi Svc Site : None
Last Status Change : 03/27/2015 14:32:53
Last Mgmt Change  : 03/26/2015 13:57:20
=====
*A:MTU-1#
```

As also depicted in [Figure 136](#), PE-1 will signal pseudowire status standby (code 0x20) when PE-1 goes to non-DF state for MH-site-2 MTU-2 will receive that signaling and, based on the **ignore-standby-signaling** parameter, will decide whether to send the broadcast, unknown unicast and multicast (BUM) traffic to PE-1. In case MTU-2 uses in its configuration **ignore-standby-signaling**, it will be sending BUM traffic on both pseudowires at the same time (which is not normally desired), ignoring the pseudowire status bits. The following output shows the MTU-2 spoke **sdp** receiving the pseudowire status signaling. Note that although the spoke SDP stays operationally up, the peer Pw Bits field shows **pwFwdingStandby** and MTU-2 will not send any traffic if the **ignore-standby-signaling** parameter is disabled.

```
A:MTU-2# show service id 500 sdp 211:500 detail
=====
Service Destination Point (Sdp Id : 211:500) Details
=====
-----
Sdp Id 211:500 - (192.0.2.1)
-----
```



```

Description      : (Not Specified)
SDP Id           : 211:500
Spoke Descr      : (Not Specified)
Split Horiz Grp  : (Not Specified)
Etree Root Leaf Tag: Disabled
VC Type          : Ether
Admin Path MTU   : 8000
Delivery         : MPLS
Far End          : 192.0.2.1
Tunnel Far End   : n/a
Hash Label       : Disabled
Oper Hash Label  : Disabled

Admin State      : Up

---snip---

Endpoint         : N/A
PW Status Sig    : Enabled
Force Vlan-Vc    : Disabled
Class Fwding State : Down
Flags           : None
Time to RetryReset : never
Mac Move         : Blockable
Local Pw Bits    : None
Peer Pw Bits     : pwFwdingStandby

---snip---

```

```

Type              : Spoke

Etree Leaf AC    : Disabled
VC Tag           : n/a
Oper Path MTU    : 8000

LSP Types        : RSVP
Hash Lbl Sig Cap : Disabled

Oper State       : Up

Precedence       : 4
Force Qinq-Vc    : Disabled

Retries Left     : 3
Blockable Level  : Tertiary

```

Operational Groups for BGP-MH

Operational groups (“oper-groups” in the CLI) introduce the capability of grouping objects into a generic group object and associating its status to other service endpoints (pseudowires, SAPs, IP interfaces) located in the same or in different service instances. The operational group status is derived from the status of the individual components using certain rules specific to the application using the concept. A number of other service entities, the monitoring objects, can be configured to monitor the operational group status and to drive their own status based on the oper-group status. In other words, if the operational group goes down, the monitoring objects will be brought down. When one of the objects included in the oper-group comes up, the entire group will also come up, and therefore so will the monitoring objects.

This concept can be used to enhance the BGP-MH solution for avoiding black-holes on the PE selected as the Designated Forwarder (DF), if the rest of the VPLS endpoints fail (pseudowire spoke(s)/pseudowire mesh and/or SAP(s)). [Figure 137](#) illustrates the use of oper-groups together with BGP-MH. On PE-1 (and PE-2) all of the BGP-VPLS pseudowires in the core are configured under the same **oper-group group-1**. MH-site-2 is configured as a monitoring object. When the two BGP-VPLS pseudowires go down, **oper-group group-1** will be brought down, therefore MH-site-2 on PE-1 will go down as well (PE-2 will become DF and PE-1 will signal standby to MTU-2).

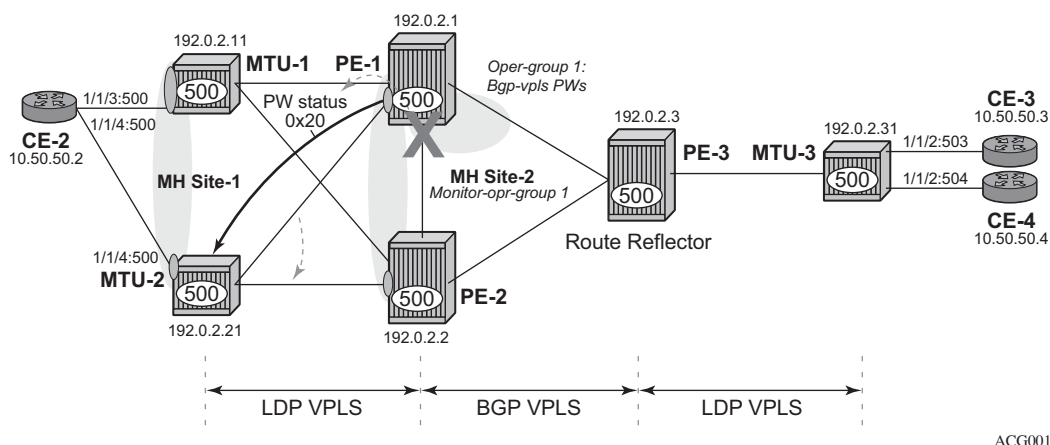


Figure 137: Oper-Groups and BGP-MH

In the example above, this feature provides a solution to avoid a black-hole when PE-1 loses its connectivity to the core.

Operational groups are configured in two steps:

1. Identify a set of objects whose forwarding state should be considered as a whole group then group them under an operational group (in this case **oper-group group-1**, which is configured in the **bgp pw-template-bind** context).
2. Associate other existing objects (clients) with the oper-group using the **monitor-group** command (configured, in this case, in the **site MH-site-2**).

The following CLI excerpt shows the commands required (**oper-group**, **monitor-oper-group**).

```
*A:PE-1# configure service
      oper-group "group-1" create
      back
      vpls 500
        bgp
          pw-template-binding 500 split-horizon-group "CORE"
          oper-group "group-1"
        exit
      exit
      site "MH-site-2"
        monitor-oper-group "group-1"
      exit all
*A:PE-1#
```

When all the BGP-VPLS pseudowires go down, **oper-group group-1** will go down and therefore the monitoring object, **site MH-site-2**, will also go down and PE-2 will then be elected as DF. The log 99 gives information about this sequence of events:

```
*A:PE-1# configure service sdp 1212 shutdown
*A:PE-1# configure service sdp 1313 shutdown

*A:PE-1# show log log-id 99

238 2015/04/01 12:47:35.54 UTC WARNING: SVCMGR #2531 Base BGP-MH
"Service-id 500 site MH-site-2 is not the designated-forwarder"

236 2015/04/01 12:47:35.54 UTC MINOR: SVCMGR #2306 Base
"Status of SDP Bind 121:500 in service 500 (customer 1) changed to admin=up oper=down
flags="

234 2015/04/01 12:47:35.54 UTC MINOR: SVCMGR #2542 Base
"Oper-group group-1 changed status to down"
```

PE-2 becomes the DF.

```
*A:PE-2# show service id 500 site detail
=====
Site Information
=====
Site Name           : MH-site-2
-----
Site Id             : 2
Dest                : sdp:221:500      Mesh-SDP Bind    : no
```

Operational Groups for BGP-MH

Admin Status	: Enabled	Oper Status	: up
Designated Fwdr	: Yes		
DF UpTime	: 0d 00:03:02	DF Chg Cnt	: 4
Boot Timer	: default	Timer Remaining	: 0d 00:00:00
Site Activation Timer	: default	Timer Remaining	: 0d 00:00:00
Min Down Timer	: default	Timer Remaining	: 0d 00:00:00
Failed Threshold	: default(all)		
Monitor Oper Grp	: group-1		

Number of Sites : 1

=====
*A:PE-2#

Note that the process reverts when at least one BGP-VPLS pseudowire comes back up.

Show Commands and Debugging Options

The main command to find out the status of a given site is the **show service id x site** command. A **detail** modifier is available:

```
*A:MTU-2# show service id 500 site
=====
VPLS Sites
=====
Site                Site-Id  Dest                Mesh-SDP  Admin  Oper  Fwdr
-----
MH-site-1           1       sap:1/1/4:500      no        Enabled up    No
-----
Number of Sites : 1
-----
*A:MTU-2#
*A:MTU-2# show service id 500 site detail
=====
Site Information
=====
Site Name           : MH-site-1
-----
Site Id              : 1
Dest                 : sap:1/1/4:500      Mesh-SDP Bind      : no
Admin Status         : Enabled              Oper Status        : up
Designated Fwdr      : No
DF UpTime             : 0d 00:00:00      DF Chg Cnt         : 1
Boot Timer            : default              Timer Remaining    : 0d 00:00:00
Site Activation Timer : default              Timer Remaining    : 0d 00:00:00
Min Down Timer        : default              Timer Remaining    : 0d 00:00:00
Failed Threshold      : default(all)
Monitor Oper Grp      : (none)
-----
Number of Sites : 1
=====
*A:MTU-2#
```

Note that the **detail** view of the command displays information about the BGP MH timers. The values are only shown if the global values are overridden by specific ones at service level (and will be tagged with **Ovr** if they have been configured at service level). The **Timer Remaining** field reflects the count down from the boot/site activation timers down to the moment when this router tries to become DF again. Again, this is only shown when the global timers have been overridden by the ones at service level.

It is important to note that the objects on the non-DF site will be brought down operationally and flagged with **StandByForMHProtocol**.

```
*A:MTU-2# show service id 500 sap 1/1/4:500
=====
Service Access Points (SAP)
=====
Service Id          : 500
```

Show Commands and Debugging Options

```
SAP                : 1/1/4:500                Encap                : q-tag
Description        : (Not Specified)
Admin State        : Up                      Oper State                : Down
Flags              : StandByForMHPProtocol
Multi Svc Site     : None
Last Status Change : 03/27/2015 12:11:58
Last Mgmt Change   : 03/27/2015 12:04:45
=====
*A:MTU-2#
*A:PE-2# show service id 500 sdp 221:500 detail
=====
Service Destination Point (Sdp Id : 221:500) Details
=====
-----
Sdp Id 221:500  -(192.0.2.21)
-----
Description      : (Not Specified)
SDP Id           : 221:500                Type                      : Spoke
---snip---
Admin State      : Up                      Oper State                : Down
---snip---
Flags            : StandbyForMHPProtocol
---snip---
```

The BGP MH routes in the RIB, RIB-IN and RIB-OUT can be shown by using the corresponding **show router bgp routes** and **show router bgp neighbor x.x.x.x received-routes|advertised-routes** commands. Note that the BGP MH routes are only shown when the operator uses the **l2-vpn family** modifier. Should the operator want to filter only the BGP MH routes out of the l2-vpn routes, the **multi-homing** filter has to be added to the **show router bgp routes** commands.

```
*A:PE-3# show router bgp routes l2-vpn multi-homing siteid 1 detail
=====
BGP Router ID:192.0.2.3      AS:65000      Local AS:65000
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
              l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP L2VPN-MULTIHOME Routes
=====
Original Attributes

Route Type      : MultiHome
Route Dist.     : 65000:511
Site Id         : 1
Nexthop         : 192.0.2.11
From            : 192.0.2.11
Res. Nexthop    : n/a
Local Pref.     : 100                      Interface Name : NotAvailable
```

```

Aggregator AS : None                      Aggregator      : None
Atomic Aggr.  : Not Atomic                 MED              : 0
AIGP Metric   : None
Connector     : None
Community     : target:65000:500
               l2-vpn/vrf-imp:Encap=19: Flags=-DF: MTU=0: PREF=0
Cluster       : No Cluster Members
Originator Id : None                      Peer Router Id  : 192.0.2.11
Flags         : Used Valid Best IGP
Route Source  : Internal
AS-Path       : No As-Path
Route Tag     : 0
Neighbor-AS   : N/A
Orig Validation: N/A
Source Class  : 0                          Dest Class      : 0
Add Paths Send : Default
Last Modified  : 01h09m26s

```

---snip---

Original Attributes

```

Route Type      : MultiHome
Route Dist.     : 65000:521
Site Id         : 1
Nexthop         : 192.0.2.21
From            : 192.0.2.21
Res. Nexthop    : n/a
Local Pref.     : 100                      Interface Name  : NotAvailable
Aggregator AS   : None                      Aggregator      : None
Atomic Aggr.    : Not Atomic                 MED              : 0
AIGP Metric     : None
Connector       : None
Community       : target:65000:500
               l2-vpn/vrf-imp:Encap=19: Flags=none: MTU=0: PREF=0
Cluster         : No Cluster Members
Originator Id   : None                      Peer Router Id  : 192.0.2.21
Flags           : Used Valid Best IGP
Route Source    : Internal
AS-Path         : No As-Path
Route Tag       : 0
Neighbor-AS     : N/A
Orig Validation: N/A
Source Class    : 0                          Dest Class      : 0
Add Paths Send  : Default
Last Modified   : 01h10m03s

```

---snip---

The following shows new command added to show the Layer 2 BGP routes:

```

*A:PE-1# show service l2-route-table
- l2-route-table [detail] [bgp-ad] [multi-homing] [bgp-vpls] [bgp-vpws] [all-routes]

<detail>                : keyword - display detailed information

*A:PE-1# show service l2-route-table multi-homing

```

Show Commands and Debugging Options

```
=====
Services: L2 Multi-Homing Route Information - Summary
=====
Svc Id      L2-Routes (RD-Prefix)      Next Hop      SiteId      State      DF
-----
500          65000:511                  192.0.2.11    1            up(0)      set
500          65000:521                  192.0.2.21    1            up(0)      clear
500          65000:502                  192.0.2.2     2            up(0)      clear
-----
No. of L2 Multi-Homing Route Entries: 3
=====
*A:PE-1#
```

Finally, in terms of debugging, the recommendation would be to check the following CLI sources:

- **log-id 99** — Provides information about the site object changes and DF changes.
- **debug router bgp update** — Shows the BGP updates for BGP MH, including the sent and received BGP MH NLRIs and flags.

```
*A:MTU-1# debug router bgp update
```

- **debug router ldp** commands — Provides information about the pseudowire status bits being signaled as well as the MAC flush messages.

```
*A:MTU-1# debug router ldp peer 192.0.2.1 packet init detail
*A:MTU-1# debug router ldp peer 192.0.2.1 packet label detail
```

As an example, log-id 99 and debug output is displayed below after shutting down MH-site-1 on MTU-1:

```
*A:MTU-1# configure service vpls 500 sap 1/1/4:500 shutdown
*A:MTU-1# configure service vpls 500 sap 1/1/3:500 shutdown

*A:MTU-1# show log log-id 99
=====
Event Log 99
=====
Description : Default System Log
Memory Log contents [size=500 next event=91 (not wrapped)]

---snip---

89 2015/03/27 13:39:04.82 UTC WARNING: SVCNMR #2531 Base BGP-MH
"Service-id 500 site MH-site-1 is not the designated-forwarder"

88 2015/03/27 13:39:04.82 UTC MINOR: SVCNMR #2203 Base
"Status of SAP 1/1/4:500 in service 500 (customer 1) changed to admin=down
oper=down flags=SapAdminDown "
```

Log 2 has been configured to log BGP updates and LDP commands.

```
*A:MTU-1# show log log-id 2
=====
Event Log 2
```



```

=====
Description : (Not Specified)
Memory Log contents [size=100 next event=11 (not wrapped)]

9 2015/03/27 13:39:04.82 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 86
  Flag: 0x90 Type: 14 Len: 28 Multiprotocol Reachable NLRI:
    Address Family L2VPN
    NextHop len 4 NextHop 192.0.2.11
    [MH] site-id: 1, RD 65000:511
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0x80 Type: 9 Len: 4 Originator ID: 192.0.2.11
  Flag: 0x80 Type: 10 Len: 4 Cluster ID:
    1.1.1.1
  Flag: 0xc0 Type: 16 Len: 16 Extended Community:
    target:65000:500
    12-vpn/vrf-imp:Encap=19: Flags=D: MTU=0: PREF=0
"

8 2015/03/27 13:39:04.82 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 72
  Flag: 0x90 Type: 14 Len: 28 Multiprotocol Reachable NLRI:
    Address Family L2VPN
    NextHop len 4 NextHop 192.0.2.11
    [MH] site-id: 1, RD 65000:511
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 16 Extended Community:
    target:65000:500
    12-vpn/vrf-imp:Encap=19: Flags=D: MTU=0: PREF=0
"

7 2015/03/27 13:39:04.82 UTC MINOR: DEBUG #2001 Base LDP
"LDP: LDP
Send Address Withdraw packet (msgId 671) to 192.0.2.1:0
Protocol version = 1
MAC Flush (All MACs learned from me)
Service FEC PWE3: ENET(5)/500 Group ID = 0 cBit = 0
"

```

Note that assuming all the recommended tools are enabled, a DF to non-DF transition can be shown as well as the corresponding MAC flush messages and related BGP processing.

If MH-site-2 is torn down on PE-1, the **debug router bgp update** command would allow us to see two BGP updates from PE-1:

Show Commands and Debugging Options

- A BGP MH update for site-id 2 with flag D set (since the site is down).
- A BGP VPLS update for veid=501 and flag D set. This is due to the fact that there are no more active objects on the VPLS, besides the BGP pseudowires.

```
*A:PE-1# configure service vpls 500 spoke-sdp 111:500 shutdown
```

```
*A:PE-1# configure service vpls 500 spoke-sdp 121:500 shutdown
```

```
*A:PE-1# show log log-id 2
```

```
=====
Event Log 2
=====
```

```
Description : (Not Specified)
```

```
Memory Log contents [size=100 next event=141 (wrapped)]
```

```
137 2015/04/01 13:28:25.08 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
```

```
"Peer 1: 192.0.2.3: UPDATE
```

```
Peer 1: 192.0.2.3 - Received BGP UPDATE:
```

```
Withdrawn Length = 0
```

```
Total Path Attr Length = 86
```

```
Flag: 0x90 Type: 14 Len: 28 Multiprotocol Reachable NLRI:
```

```
Address Family L2VPN
```

```
NextHop len 4 NextHop 192.0.2.2
```

```
[MH] site-id: 2, RD 65000:502
```

```
Flag: 0x40 Type: 1 Len: 1 Origin: 0
```

```
Flag: 0x40 Type: 2 Len: 0 AS Path:
```

```
Flag: 0x80 Type: 4 Len: 4 MED: 0
```

```
Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
```

```
Flag: 0x80 Type: 9 Len: 4 Originator ID: 192.0.2.2
```

```
Flag: 0x80 Type: 10 Len: 4 Cluster ID:
```

```
1.1.1.1
```

```
Flag: 0xc0 Type: 16 Len: 16 Extended Community:
```

```
target:65000:500
```

```
12-vpn/vrf-imp:Encap=19: Flags=D: MTU=0: PREF=0
```

```
"
```

```
136 2015/04/01 13:28:25.08 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
```

```
"Peer 1: 192.0.2.3: UPDATE
```

```
Peer 1: 192.0.2.3 - Received BGP UPDATE:
```

```
Withdrawn Length = 0
```

```
Total Path Attr Length = 86
```

```
Flag: 0x90 Type: 14 Len: 28 Multiprotocol Reachable NLRI:
```

```
Address Family L2VPN
```

```
NextHop len 4 NextHop 192.0.2.1
```

```
[VPLS/VPWS] preflen 17, veid: 501, vbo: 497, vbs: 8, label-base: 131050, RD 65000:501
```

```
Flag: 0x40 Type: 1 Len: 1 Origin: 0
```

```
Flag: 0x40 Type: 2 Len: 0 AS Path:
```

```
Flag: 0x80 Type: 4 Len: 4 MED: 0
```

```
Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
```

```
Flag: 0x80 Type: 9 Len: 4 Originator ID: 192.0.2.1
```

```
Flag: 0x80 Type: 10 Len: 4 Cluster ID:
```

```
1.1.1.1
```

```
Flag: 0xc0 Type: 16 Len: 16 Extended Community:
```

```
target:65000:500
```

```
12-vpn/vrf-imp:Encap=19: Flags=D: MTU=1514: PREF=0
```

The D flag, sent along with the BGP VPLS update for veid 501, would be seen on the remote core PEs as though it was a pseudowire status fault (although there is no TLDP running in the core).

```
*A:PE-2# show service id 500 all | match Flag
Flags          : None
Flags          : None
Flags          : PWPeerFaultStatusBits
Flags          : None
```

When oper-groups are configured (as previously shown), the following **show** command helps to find the operational dependencies between monitoring objects and group objects.

```
*A:PE-1# show service oper-group "group-1" detail
=====
Service Oper Group Information
=====
Oper Group      : group-1
Creation Origin : manual                      Oper Status: up
Hold DownTime   : 0 secs                      Hold UpTime: 4 secs
Members         : 2                          Monitoring  : 1
=====

=====
Member SDP-Binds for OperGroup: group-1
=====
SdpId           SvcId      Type IP address      Adm    Opr
-----
1212:4294967295  500        Bgp* 192.0.2.2      Up     Up
1313:4294967292  500        Bgp* 192.0.2.3      Up     Up
-----
SDP Entries found: 2
=====
* indicates that the corresponding row element may have been truncated.

=====
Monitoring Sites for OperGroup: group-1
=====
SvcId   Site           Site-Id  Dest                Admin  Oper  Fwdr
-----
500     MH-site-2       2        sdp:121:500        Enabled up    Yes
-----
Site Entries found: 1
=====
*A:PE-1#
```

Conclusion

SR OS supports a wide range of service resiliency options as well as the best-of-breed system level HA and MPLS mechanisms for the access and the core. BGP MH for VPLS completes the service resiliency toolset by adding a mechanism that has some good advantages over the alternative solutions:

- BGP MH provides a common resiliency mechanism for attachment circuits (SAPs), pseudowires (spoke SDPs), split horizon groups and mesh bindings
- BGP MH is a network-based technique which does not need interaction to the CE or MTU to which it is providing redundancy to.

The examples used in this section illustrate the configuration of BGP MH for access CEs and MTUs. Show and debug commands have also been suggested so that the operator can verify and troubleshoot the BGP MH procedures.

In This Chapter

This section describes advanced BGP VPLS configurations.

Topics in this section include:

- [Applicability on page 866](#)
- [Summary on page 867](#)
- [Overview on page 868](#)
- [Configuration on page 870](#)
- [Conclusion on page 903](#)

Applicability

This example is applicable to all of the 7450 ESS and 7750 SR series and was tested on Release 13.0.R3. There are no pre-requisites for this configuration.

Summary

There are currently two IETF standards for the provisioning of Virtual Private LAN Services (VPLS). RFC 4762, *Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling*, describes Label Distribution Protocol (LDP) VPLS, where VPLS pseudowires are signaled using LDP between VPLS Provider Edge (PE) routers, either configured manually or auto-discovered using BGP.

RFC 4761, *Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling*, describes the use of Border Gateway Protocol (BGP) for both the auto-discovery of VPLS PEs and signaling of pseudowires between such PEs.

The purpose of this section is to describe the configuration and troubleshooting for BGP-VPLS.

Knowledge of BGP-VPLS RFC 4761 architecture and functionality is assumed throughout this chapter, as well as knowledge of Multi-Protocol BGP.

Overview

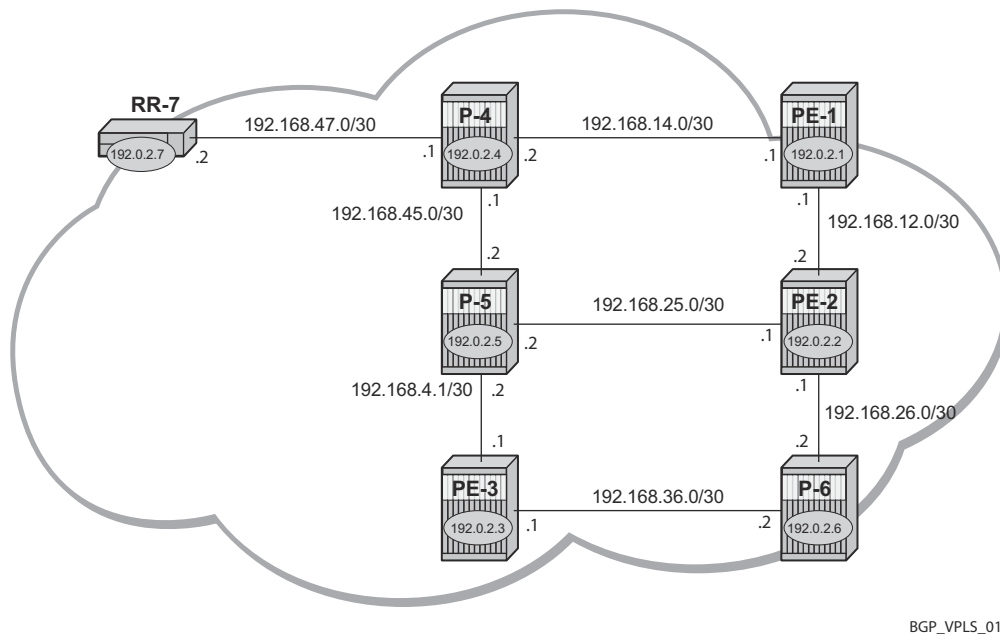


Figure 138: Network Topology

The network topology is displayed in [Figure 138](#). The configuration uses seven 7750/7450 Service Router (SR) nodes located in the same Autonomous System (AS). There are three Provider Edge (PE) routers, and RR-7 will act as a Route Reflector (RR) for the AS. The PE routers are all VPLS-aware, the Provider (P) routers are VPLS unaware and do not take part in the BGP process.

The following configuration tasks should be completed as a pre-requisite:

- ISIS or OSPF on each of the network interfaces between the PE/P routers and RR.
- MPLS should be configured on all interfaces between PE routers and P routers. MPLS is not required between P-4 and RR-7.
- LDP should be configured on interfaces between PE and P routers. It is not required between P-4 and the RR-7.
- The RSVP protocol should be enabled.

BGP VPLS

In this architecture a VPLS instance is a collection of local VPLS instances present on a number of PEs in a provider network. In this context, any VPLS-aware PE is also known as a VPLS Edge (VE) device.

The PEs communicate with each other at the control plane level by means of BGP updates containing BGP-VPLS Network Layer Reachability Information (NLRI). Each update contains enough information for a PE to determine the presence of other local VPLS instances on peering PEs and to set-up pseudowire connectivity for data flow between peers containing a local VPLS within the same VPLS instance. Therefore, auto-discovery and pseudowire signaling are achieved using a single BGP update message.

Each PE within a VPLS instance is identified by a VPLS Edge identifier (ve-id) and the presence of a VPLS instance is determined using the exchange of standard BGP extended community route targets between PEs.

Each PE will advertise, via the route reflectors, the presence of each VPLS instance to all other PEs, along with a block of multiplexer labels that can be used to communicate between such instances plus a BGP next hop that determines a labelled transport tunnel between PEs.

Each VPLS instance is configured with import and export route target extended communities for topology control, along with VE identification.

Configuration

The first step is to configure an MP-iBGP session between each of the PEs and the RR.

The configuration for PE-1 is as follows:

```
*A:PE-1# configure router
      bgp
        group "INTERNAL"
          family l2-vpn
          type internal
          neighbor 192.0.2.7
          exit
        exit
      no shutdown
    exit
```

The BGP configuration for the other PE nodes is identically the same. The IP addresses can be derived from [Figure 138](#).

The configuration for RR-7 is as follows:

```
*A:RR-7# configure router
      bgp
        cluster 1.1.1.1
        group "RR-INTERNAL"
          family l2-vpn
          type internal
          neighbor 192.0.2.1
          exit
          neighbor 192.0.2.2
          exit
          neighbor 192.0.2.3
          exit
        exit
      no shutdown
    exit
```

On PE-1, verify that the BGP session with RR-7 is established with address family l2-vpn capability negotiated:

```
*A:PE-1# show router bgp neighbor 192.0.2.7
=====
BGP Neighbor
=====
-----
Peer           : 192.0.2.7
Description    : (Not Specified)
Group          : INTERNAL
-----
Peer AS        : 65536           Peer Port      : 50439
Peer Address   : 192.0.2.7
Local AS       : 65536           Local Port     : 179
```

```

Local Address      : 192.0.2.1
Peer Type          : Internal
State              : Established      Last State          : Established
Last Event         : recvKeepAlive
Last Error         : Cease (Connection Collision Resolution)
Local Family       : L2-VPN
Remote Family      : L2-VPN
Hold Time          : 90                Keep Alive          : 30
Min Hold Time      : 0
Active Hold Time   : 90                Active Keep Alive    : 30
Cluster Id         : None
Preference         : 170                Num of Update Flaps  : 0

```

```
---snip---
```

```
Neighbors : 1
```

```

=====
* indicates that the corresponding row element may have been truncated.
*A:PE-1#

```

On RR-7, show that BGP sessions with each PE are established, and have a negotiated the l2-vpn address family capability.

```
*A:RR-7# show router bgp summary
```

```

=====
BGP Router ID:192.0.2.7      AS:65536      Local AS:65536
=====
BGP Admin State      : Up      BGP Oper State      : Up
Total Peer Groups    : 1      Total Peers          : 3
Total BGP Paths       : 7      Total Path Memory    : 1288

```

```
---snip---
```

```
BGP Summary
```

```

=====
Neighbor
Description
          AS PktRcvd InQ Up/Down  State|Rcv/Act/Sent (Addr Family)
          PktSent OutQ
-----
192.0.2.1
          65536      4    0 00h00m48s 0/0/0 (L2VPN)
          4          0
192.0.2.2
          65536      4    0 00h00m48s 0/0/0 (L2VPN)
          4          0
192.0.2.3
          65536      4    0 00h00m48s 0/0/0 (L2VPN)
          4          0
-----

```

```
*A:RR-7#
```

Configuration

Configure a full mesh of RSVP-TE LSPs between PE routers.

The MPLS interface and LSP configuration for PE-1 are:

```
*A:PE-1# configure router
mpls
  interface "system"
  exit
  interface "int-PE-1-PE-2"
  exit
  interface "int-PE-1-P-4"
  exit
  path "loose"
  no shutdown
  exit
  lsp "LSP-PE-1-PE-2"
  to 192.0.2.2
  primary "loose"
  exit
  no shutdown
  exit
  lsp "LSP-PE-1-PE-3"
  to 192.0.2.3
  primary "loose"
  exit
  no shutdown
  exit
  no shutdown
exit
```

The MPLS and LSP configuration for PE-2 and PE-3 are similar to that of PE-1 with the appropriate interfaces and LSP names configured.

BGP VPLS PE Configuration

Pseudowire Templates

Pseudowire templates are used by BGP to dynamically instantiate SDP bindings, for a given service, to signal the egress service de-multiplexer labels used by remote PEs to reach the local PE.

The template determines the signaling parameters of the pseudowire, control word presence, MAC-pinning, filters etc., plus other usage characteristics such as split horizon groups.

The MPLS transport tunnel between PEs can be signaled using LDP or RSVP-TE.

LDP based pseudowires can be automatically instantiated. RSVP-TE based SDPs have to be pre-provisioned.

Pseudowire Templates for Auto-SDP Creation Using LDP

In order to use an LDP transport tunnel for data flow between PEs, it is necessary for link layer LDP to be configured between all PEs/Ps, so that a transport label for each PE's system interface is available.

```
*A:PE-1# configure router
      ldp
        interface-parameters
          interface "int-PE-1-PE-2"
          exit
          interface "int-PE-1-P-4"
          exit
        exit
      exit
```

Using this mechanism SDPs can be auto-instantiated with SDP-ids starting at the higher end of the SDP numbering range, such as 17407. Any subsequent SDPs created use SDP-ids decrementing from this value.

A pseudowire template is required containing a split horizon group. Each SDP created with this template is contained within a split horizon group so that traffic cannot be forwarded between them.

```
*A:PE-1# configure service
      pw-template 1 create
        split-horizon-group "VPLS-SHG"
        exit
      exit
```

The pseudowire template also has the following options available when used for BGP-VPLS:

```
*A:PE-1# configure service pw-template
---snip---
[no] controlword
---snip---
[no] force-vlan-vc-forwarding
---snip---
vc-type {ether | vlan}
---snip---
```

- The control word will determine whether the C flag is set in the Layer 2 extended community and, therefore, if a control word is used in the pseudowire.
- The encap type in the Layer 2 extended community is always 19 (VPLS encap), therefore, the vc-type will always be **ether** regardless of the configured value on the vc-type.
- The **force-vlan-vc-forwarding** command will add a tag (equivalent to **vc-type vlan**) and will allow for customer QoS transparency (dot1p+Drop Eligibility (DE) bits).

Pseudowire Templates for Provisioned SDPs using RSVP-TE

To use an RSVP-TE tunnel as transport between PEs, it is necessary to bind the RSVP-TE LSP between PEs to an SDP.

SDP creation from PE-1 to PE-2 is shown below:

```
*A:PE-1# configure service
      sdp 12 mpls create
        description "SDP-PE-1-PE-2_RSVP_BGP"
        signaling bgp
        far-end 192.0.2.2
        lsp "LSP-PE-1-PE-2"
        no shutdown
      exit
```

Note that the **signaling bgp** parameter is required for BGP-VPLS to be able to use this SDP. Conversely, SDPs that are bound to RSVP-based LSPs with signaling set to the default value of **tldp** will not be used as SDPs within BGP-VPLS.

BGP VPLS Using Auto-Provisioned SDPs

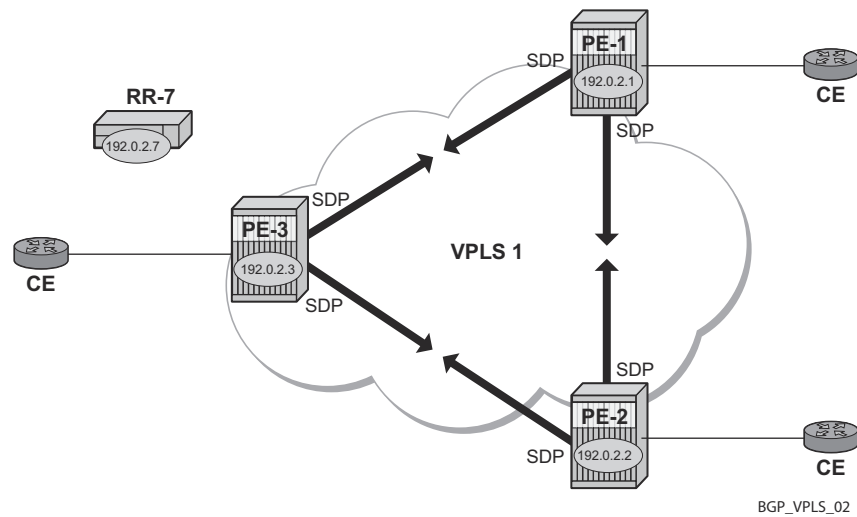


Figure 139: BGP VPLS Using Auto-Provisioned SDPs

Figure 139 shows a VPLS instance where SDPs are auto-provisioned. In this case, the transport tunnels are LDP signaled.

The following output shows the configuration required on PE-1 for a BGP-VPLS service using a pseudowire template configured for auto-provisioning of SDPs.

```
A:PE-1# configure service
      vpls 1 customer 1 create
        bgp
          route-distinguisher 65536:1
          route-target export target:65536:1 import target:65536:1
          pw-template-binding 1
        exit
      exit
      bgp-vpls
        max-ve-id 10
        ve-name "PE-1"
        ve-id 1
      exit
      no shutdown
    exit
    service-name "VPLS1_PE-1"
    sap 1/1/4:1.0 create
  exit
  no shutdown
exit
```

The **bgp** context specifies parameters which are valid for all of the VPLS BGP applications, such as BGP-multi-homing, BGP-auto-discovery as well as BGP-VPLS.

Within the **bgp** context, parameters are configured that are used by neighboring PEs to determine membership of a given VPLS instance, such as the auto-discovery of PEs containing the same VPLS instance; the route-distinguisher is configured, along with the route target extended communities.

Route target communities are used to determine membership of a given VPLS instance. Note that the import route target at the BGP level is mandatory. The pseudowire template bind is then applied by the service manager on the received routes matching the route target value.

Also within the **bgp-vpls** context, the signaling parameters are configured. These determine the service labels required for the data plane of the VPLS instance.

The VPLS edge ID (ve-id) is a numerical value assigned to each PE within a VPLS instance. This value should be unique for a given VPLS instance, no two PEs within the same instance should have the same ve-id values.

Note that a more specific route target can be applied to a pseudowire template in order to define a specific pseudowire topology, rather than only a full mesh, using the command within the **bgp** context:

pw-template *template-id* [**split-horizon-group** *groupname*] [**import-rt** *import-rt-value* (up to 5 max)]

It is also worth noting that changes to the import policies are not taken once the pseudowire has been setup (changes on route-target are refreshed though). Pseudowire templates can be re-evaluated with the command **tools perform service eval-pw-template**. The **eval-pw-template** command checks whether all the bindings using this pseudowire template policy are still meant to use this policy.

If the policy has changed and **allow-service-impact** is true, then the old binding is removed and it is re-added with the new template.

VE-ID and BGP Label Allocations

The choice of ve-id is crucial in ensuring efficient allocation of de-multiplexer labels. The most efficient choice is for ve-ids to be allocated starting at 1 and incrementing for each PE as the following section explains.

The **max-ve-id** *value* determines the range of the ve-id value that can be configured. If a PE receives a BGP-VPLS update containing a ve-id with a greater value than the configured **max-ve-id**, then the update is dropped and no service labels are installed for this ve-id.

The **max-ve-id** command also checks the locally-configured ve-id, and prevents a higher value from being used.

Each PE allocates blocks of labels per VPLS instance to remote PEs, in increments of eight labels. It achieves this by advertising three parameters in a BGP update message,

- A label base (LB) which is the lowest label in the block
- A VE Block size (VBS) which is always eight labels, and cannot be changed
- A VE base offset (VBO).

This defines a block of labels in the range (LB, LB+1, ..., LB+VBS-1).

As an example, if the label base (LB) = 262128, then the range for the block is 262128 to 262135, which is exactly eight labels, as per the block size. (The last label in the block is calculated as $262128+8-1 = 262135$)

The label allocated by the PE to each remote PE within the VPLS is chosen from this block and is determined by its ve-id. In this way, each remote PE has a unique de-multiplexer label for that VPLS.

To reduce label wastage, contiguous ve-ids in the range (N..N+7) per VPLS should be chosen, where $N > 0$.

Assuming a collection of PEs with contiguous ve-ids, the following labels will be chosen by PEs from the label block allocated by PE-1 which has a ve-id =1.

Table 5: VE-IDs and Labels

VE-ID	Label
2	262129
3	262130
4	262131
5	262132

Table 5: VE-IDs and Labels (Continued)

VE-ID	Label
6	262133
7	262434
8	262135

This shows that the label allocated to a given PE is (LB+veid-1). The “1” is the VE block offset (VBO).

This means that the label allocated to a PE router within the VPLS can now be written as (LB + veid - VBO), which means that (ve-id - VBO) calculation must always be at least zero and be less than the block size, which is always 8.

For $\text{ve-id} \leq 8$, a label will be allocated from this block.

For the next block of 8 ve-ids (ve-id 9 to ve-id 16) a new block of 8 labels must be allocated, so a new BGP update is sent, with a new label base, and a block offset of 9.

[Table 6](#) shows how the choice of ve-ids can affect the number of label blocks allocated, and hence the number of labels:

Table 6: VE-IDs and Number of Labels

VE-ID	Block Offset	Labels Allocated
1-8	1	8
9-16	9	8
17-24	17	8
25-32	25	8
33-40	33	8
41-48	41	8
49-56	49	8

This shows that the most efficient use of labels occurs when the ve-ids for a set of PEs are chosen from the same block offset.

Note that if ve-ids are chosen that map to different block offsets, then each PE will have to send multiple BGP updates to signal service labels. Each PE sends label blocks in BGP updates to each of its BGP neighbors for all label blocks in which at least one ve-id has been seen by this PE (it

does not advertise label blocks which do not contain an active ve-id, where active ve-id means the ve-id of this PE or any other PE in this VPLS).

The **max-ve-id** must be configured first, and determines the maximum value of the ve-id that can be configured within the PE. The ve-id value cannot be higher than this within the PE configuration, $\text{ve-id} \leq \text{max-ve-id}$. Similarly, if the ve-id within a received NLRI is higher than the **max-ve-id** value, it will not be accepted as valid consequently the max-ve-id configured on all PEs must be greater than or equal to any ve-id used in the VPLS.

Only one ve-id value can be configured. If the ve-id value is changed, BGP withdraws the NLRI and sends a route-refresh.

Note that if the same ve-id is used in different PEs for the same VPLS, a Designated Forwarder election takes place.

Executing the **shutdown** command triggers an MP-UNREACH-NLRI from the PE to all BGP peers.

The **no shutdown** command triggers an MP-REACH-NLRI to the same peers.

PE-2 Service Creation

On PE-2 create a VPLS service using pseudowire template 1. In order to make the label allocation more efficient, PE-2 has been allocated a ve-id value of 2. For completeness, the pseudowire template is also shown.

```
*A:PE-2# configure service
      pw-template 1 create
        split-horizon-group "VPLS-SHG"
      exit
    exit
  vpls 1 customer 1 create
    bgp
      route-distinguisher 65536:1
      route-target export target:65536:1 import target:65536:1
      pw-template-binding 1
    exit
  exit
  bgp-vpls
    max-ve-id 10
    ve-name "PE-2"
    ve-id 2
  exit
  no shutdown
exit
service-name "VPLS1_PE-2"
sap 1/1/4:1.0 create
exit
no shutdown
exit
```

Note that the **max-ve-id** *value* is set to 10 to allow an increase in the number of PEs that could be a part of this VPLS instance.

PE-3 Service Creation

Create a VPLS instance on PE-3, using a ve-id value of 3.

```
*A:PE-3# configure service
pw-template 1 create
    split-horizon-group "VPLS-SHG"
    exit
exit
vpls 1 customer 1 create
    bgp
        route-distinguisher 65536:1
        route-target export target:65536:1 import target:65536:1
        pw-template-binding 1
    exit
    exit
    bgp-vpls
        max-ve-id 10
        ve-name "PE-3"
        ve-id 3
    exit
    no shutdown
exit
service-name "VPLS1_PE-3"
sap 1/1/4:1.0 create
exit
no shutdown
exit
```

PE-1 Service Operation Verification

Verify that the BGP-VPLS site is enabled on PE-1.

```
*A:PE-1# show service id 1 bgp-vpls
=====
BGP VPLS Information
=====
Max Ve Id      : 10                Admin State    : Enabled
VE Name        : PE-1              VE Id         : 1
PW Tmpl used   : 1
=====
*A:PE-1#
```

Verify that the service is operationally up on PE-1.

```
*A:PE-1# show service id 1 base
=====
Service Basic Information
=====
Service Id      : 1                Vpn Id         : 0
Service Type    : VPLS
Name            : VPLS1_PE-1
Description     : (Not Specified)
Customer Id     : 1                Creation Origin : manual
Last Status Change: 07/13/2015 12:06:05
Last Mgmt Change : 07/13/2015 12:07:05
Etree Mode     : Disabled
Admin State     : Up               Oper State      : Up
MTU             : 1514             Def. Mesh VC Id : 1
SAP Count       : 1               SDP Bind Count  : 2
Snd Flush on Fail : Disabled       Host Conn Verify : Disabled
Propagate MacFlush: Disabled       Per Svc Hashing  : Disabled
Allow IP Intf Bind: Disabled
Def. Gateway IP : None
Def. Gateway MAC : None
Temp Flood Time : Disabled         Temp Flood      : Inactive
Temp Flood Chg Cnt: 0
VSD Domain      : <none>
SPI load-balance : Disabled

-----
Service Access & Destination Points
-----
Identifier                               Type      AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/4:1.0                           qinq      1522    1522    Up    Up
sdp:17406:4294967294 SB(192.0.2.2)      BgpVpls   0        1556    Up    Up
sdp:17407:4294967295 SB(192.0.2.3)      BgpVpls   0        1556    Up    Up
=====
*A:PE-1#
```

The SAP and SDPs are all operationally up. Note that the SB flags signify Spoke and BGP.

Further verification can be seen below where the ingress labels for PE-2 and PE-3, the labels allocated by PE-1, can be seen.

```
*A:PE-1# show service id 1 sdp
=====
Services: Service Destination Points
=====
SdpId          Type Far End addr      Adm    Opr      I.Lbl      E.Lbl
-----
17406:4294967294 Bgp* 192.0.2.2      Up     Up       262129     262126
17407:4294967295 Bgp* 192.0.2.3      Up     Up       262130     262128
-----
Number of SDPs : 2
-----
* indicates that the corresponding row element may have been truncated.
*A:PE-1#
```

As can be seen from the following output, a BGP-VPLS NLRI update is sent to the route reflector (192.0.2.7) and is received by each PE.

The following debug trace from PE-1 shows the BGP NLRI update for VPLS 1 sent by PE-1 to the route reflector.

```
1 2015/07/13 12:06:09.16 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.7
"Peer 1: 192.0.2.7: UPDATE
Peer 1: 192.0.2.7 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 72
  Flag: 0x90 Type: 14 Len: 28 Multiprotocol Reachable NLRI:
    Address Family L2VPN
    NextHop len 4 NextHop 192.0.2.1
    [VPLS/VPWS] preflen 17, veid: 1, vbo: 1, vbs: 8, label-base: 262128,
    RD 65536:1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 16 Extended Community:
    target:65536:1
    12-vpn/vrf-imp:Encap=19: Flags=none: MTU=1514: PREF=0
"
```

Note the presence of the control flags within the extended community which indicates the status of the VPLS instance.

New control flags have been introduced to allow support for BGP multi-homing, D indicates that all attachment circuits are Down, or the VPLS is shutdown. The flags are used in BGP Multi-homing when determining which PEs are designated forwarders.

When flags=none, then all attachment circuits are up. In the example above no flags are present, but should all SAPs become operationally down, then the following debug would be seen:

BGP VPLS PE Configuration

```
5 2015/07/13 12:09:46.16 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.7
"Peer 1: 192.0.2.7: UPDATE
Peer 1: 192.0.2.7 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 72
  Flag: 0x90 Type: 14 Len: 28 Multiprotocol Reachable NLRI:
    Address Family L2VPN
    NextHop len 4 NextHop 192.0.2.1
    [VPLS/VPWS] preflen 17, veid: 1, vbo: 1, vbs: 8, label-base: 262128, RD 65536:1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 16 Extended Community:
    target:65536:1
    l2-vpn/vrf-imp:Encap=19: Flags=D: MTU=1514: PREF=0
```

The BGP VPLS signaling parameters are also present, namely the ve-id of the PE within the VPLS instance, the VBO and VBS, and the label base. The target indicates the VPLS instance, which must be matched against the import route targets of the receiving PEs.

The signaling parameters can be seen within the BGP update with following command:

```
*A:PE-1# show router bgp routes l2-vpn rd 65536:1 hunt
=====
  BGP Router ID:192.0.2.1      AS:65536      Local AS:65536
=====
---snip---

-----
RIB Out Entries
-----
Route Type      : VPLS
Route Dist.     : 65536:1
VeId            : 1                      Block Size      : 8
Base Offset     : 1                      Label Base      : 262128
Nexthop         : 192.0.2.1
To              : 192.0.2.7
Res. Nexthop    : n/a
Local Pref.     : 100
Aggregator AS   : None                  Interface Name  : NotAvailable
Atomic Aggr.    : Not Atomic            Aggregator      : None
AIGP Metric     : None                  MED            : 0
Connector       : None
Community       : target:65536:1
                  l2-vpn/vrf-imp:Encap=19: Flags=none: MTU=1514: PREF=0
Cluster         : No Cluster Members
Originator Id   : None                  Peer Router Id  : 192.0.2.7
Origin          : IGP
AS-Path         : No As-Path
Route Tag       : 0
Neighbor-AS     : N/A
Orig Validation : N/A
Source Class    : 0                      Dest Class      : 0
-----
```



```
Routes : 4
```

```
=====
*A:PE-1#
```

In this configuration example, PE-1 (192.0.2.1) with ve-id =1 has sent an update with base offset (VBO) =1, block size (VBS) = 8 and label base 262128. This means that labels 262128 (LB) to 262135 (LB+VBS-1) are available as de-multiplexer labels, egress labels to be used to reach PE-1 for VPLS 1.

PE-2 receives this update from PE-1. This is seen as a valid VPLS BGP route from PE-1 through the route reflector with nexthop 192.0.2.1.

```
*A:PE-2# show router bgp routes l2-vpn rd 65536:1
```

```
=====
BGP Router ID:192.0.2.2      AS:65536      Local AS:65536
=====
```

```
Legend -
```

```
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
                l - leaked
```

```
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
```

```
=====
BGP L2VPN Routes
=====
```

Flag	RouteType	Prefix	MED
	RD	SiteId	Label
	Nexthop	VeId	LocalPref
	As-Path	BaseOffset	vplsLabelBase
u*>i	VPLS	-	0
	65536:1	-	-
	192.0.2.1	1	100
	No As-Path	1	262128
i	VPLS	-	0
	65536:1	-	-
	192.0.2.2	2	100
	No As-Path	1	262126
u*>i	VPLS	-	0
	65536:1	-	-
	192.0.2.3	3	100
	No As-Path	1	262128

```
-----
Routes : 3
```

```
=====
*A:PE-2#
```

PE-2 uses this information in conjunction with its own ve-id to calculate the egress label towards PE-1, using the condition $VBO \leq \text{ve-id} < (VBO + VBS)$.

The ve-id of PE-2 is in the Label Block covered by VBO =1, thus,

BGP VPLS PE Configuration

Label calculation = label base + local ve-id - Base offset
= 262128 + 2 - 1
Egress label used = 262129

This is verified using the following command on PE-2 where the egress label toward PE-1 (192.0.2.1) is 262129.

```
*A:PE-2# show service id 1 sdp
=====
Services: Service Destination Points
=====
SdpId          Type Far End addr    Adm    Opr      I.Lbl    E.Lbl
-----
17406:4294967294 Bgp* 192.0.2.3      Up     Up       262128   262129
17407:4294967295 Bgp* 192.0.2.1      Up     Up       262126   262129
-----
Number of SDPs : 2
-----
* indicates that the corresponding row element may have been truncated.
*A:PE-2#
```

PE-3 also receives this update from PE-1 by the route reflector. This is seen as a valid VPLS BGP route from PE-1 with nexthop 192.0.2.1.

```
*A:PE-3# show router bgp routes l2-vpn rd 65536:1
=====
BGP Router ID:192.0.2.3      AS:65536      Local AS:65536
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP L2VPN Routes
=====
Flag RouteType      Prefix      MED
      RD            SiteId      Label
      Nexthop       VeId        LocalPref
      As-Path       BaseOffset  vplsLabelBase
-----
u*>i VPLS              -            -            0
      65536:1        -            -            -
      192.0.2.1      1            8            100
      No As-Path     1            262128
u*>i VPLS              -            -            0
      65536:1        -            -            -
      192.0.2.2      2            8            100
      No As-Path     1            262126
i    VPLS              -            -            0
      65536:1        -            -            -
      192.0.2.3      3            8            100
      No As-Path     1            262128
-----
```

```
Routes : 3
```

```
=====
```

```
*A:PE-3#
```

The ve-id of PE-3 is also in the label block covered by block offset VBO =1.

Label calculation = label base + local ve-id - VBO

= 262128 + 3 - 1

Egress label used = 262130

This is verified using the following command on PE-3 where egress label towards 192.0.2.1 is 262130.

```
*A:PE-3# show service id 1 sdp
```

```
=====
```

```
Services: Service Destination Points
```

```
=====
```

SdpId	Type	Far End addr	Adm	Opr	I.Lbl	E.Lbl
17406:4294967294	Bgp*	192.0.2.2	Up	Up	262129	262128
17407:4294967295	Bgp*	192.0.2.1	Up	Up	262128	262130

```
Number of SDPs : 2
```

```
=====
```

```
* indicates that the corresponding row element may have been truncated.
```

```
*A:PE-3#
```

PE-2 Service Operation Verification

For completeness, verify the service is operationally up on PE-2.

```
*A:PE-2# show service id 1 base
=====
Service Basic Information
=====
Service Id       : 1                Vpn Id       : 0
Service Type     : VPLS
Name             : VPLS1_PE-2
Description      : (Not Specified)
Customer Id      : 1                Creation Origin : manual
Last Status Change: 07/13/2015 12:07:34
Last Mgmt Change : 07/13/2015 12:08:20
Etree Mode      : Disabled
Admin State      : Up               Oper State     : Up
MTU              : 1514             Def. Mesh VC Id : 1
SAP Count        : 1               SDP Bind Count  : 2
Snd Flush on Fail : Disabled        Host Conn Verify : Disabled
Propagate MacFlush: Disabled        Per Svc Hashing  : Disabled
Allow IP Intf Bind: Disabled
Def. Gateway IP   : None
Def. Gateway MAC  : None
Temp Flood Time   : Disabled        Temp Flood      : Inactive
Temp Flood Chg Cnt: 0
VSD Domain        : <none>
SPI load-balance  : Disabled
=====

Service Access & Destination Points
=====
Identifier                               Type      AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/4:1.0                           qinq      1522    1522    Up    Up
sdp:17406:4294967294 SB(192.0.2.3)      BgpVpls   0        1556    Up    Up
sdp:17407:4294967295 SB(192.0.2.1)      BgpVpls   0        1556    Up    Up
=====
*A:PE-2#
```

PE-2 De-Multiplexer Label Calculation

In the same way that PE-1 allocates a label base (LB), block size (VBS), and base offset (VBO), PE-2 also allocates the same parameters for PE-1 and PE-3 to calculate the egress service label required to reach PE-2.

```
*A:PE-2# show router bgp routes l2-vpn rd 65536:1 hunt
=====
BGP Router ID:192.0.2.2          AS:65536          Local AS:65536
=====
BGP L2VPN Routes
=====

---snip---

RIB Out Entries
-----
Route Type      : VPLS
Route Dist.     : 65536:1
VeId            : 2                      Block Size      : 8
Base Offset     : 1                      Label Base       : 262126
Nexthop         : 192.0.2.2
To              : 192.0.2.7
Res. Nexthop    : n/a
Local Pref.     : 100
Aggregator AS   : None                  Interface Name   : NotAvailable
Atomic Aggr.    : Not Atomic            Aggregator      : None
AIGP Metric     : None                  MED             : 0
Connector       : None
Community       : target:65536:1
                  l2-vpn/vrf-imp:Encap=19: Flags=none: MTU=1514: PREF=0
Cluster         : No Cluster Members
Originator Id   : None                  Peer Router Id   : 192.0.2.7
Origin          : IGP
AS-Path         : No As-Path
Route Tag       : 0
Neighbor-AS     : N/A
Orig Validation : N/A
Source Class    : 0                      Dest Class       : 0
-----
Routes : 4
=====
*A:PE-2#
```

This is verified using the following command on PE-1 to show the egress label towards PE-2 (192.0.2.2) where the egress label towards PE-2 = 262126 + 1 - 1 = 262126.

```
*A:PE-1# show service id 1 sdp
=====
Services: Service Destination Points
=====
SdpId          Type Far End addr   Adm    Opr      I.Lbl    E.Lbl
-----
17406:4294967294 Bgp* 192.0.2.2    Up     Up       262129   262126
17407:4294967295 Bgp* 192.0.2.3    Up     Up       262130   262128
=====
```

BGP VPLS PE Configuration

```
Number of SDPs : 2
```

```
=====
* indicates that the corresponding row element may have been truncated.
*A:PE-1#
```

This is also verified using the following command on PE-3 to show the egress label towards PE-2 (192.0.2.2) where the egress label towards PE - 2 = 262126 + 3 - 1 = 262128.

```
*A:PE-3# show service id 1 sdp
```

```
=====
Services: Service Destination Points
```

```
=====
SdpId              Type Far End addr   Adm    Opr      I.Lbl    E.Lbl
-----
17406:4294967294 Bgp* 192.0.2.2    Up     Up       262129   262128
17407:4294967295 Bgp* 192.0.2.1    Up     Up       262128   262130
=====
```

```
Number of SDPs : 2
```

```
=====
* indicates that the corresponding row element may have been truncated.
*A:PE-3#
```

PE-3 Service Operation Verification

Verify that the service is operationally up on PE-3:

```
*A:PE-3# show service id 1 base
```

```
=====
Service Basic Information
```

```
=====
Service Id       : 1                      Vpn Id          : 0
Service Type     : VPLS
Name             : VPLS1_PE-3
Description      : (Not Specified)
Customer Id      : 1                      Creation Origin  : manual
Last Status Change: 07/13/2015 12:08:54
Last Mgmt Change  : 07/13/2015 12:09:36
Etree Mode       : Disabled
Admin State      : Up                     Oper State       : Up
MTU              : 1514                   Def. Mesh VC Id  : 1
SAP Count        : 1                     SDP Bind Count   : 2
Snd Flush on Fail : Disabled              Host Conn Verify : Disabled
Propagate MacFlush: Disabled              Per Svc Hashing  : Disabled
Allow IP Intf Bind: Disabled
Def. Gateway IP   : None
Def. Gateway MAC  : None
Temp Flood Time   : Disabled              Temp Flood       : Inactive
Temp Flood Chg Cnt: 0
VSD Domain        : <none>
SPI load-balance  : Disabled
```

Service Access & Destination Points

Identifier	Type	AdmMTU	OprMTU	Adm	Opr
sap:1/1/4:1.0	qinq	1522	1522	Up	Up
sdp:17406:4294967294 SB(192.0.2.2)	BgpVpls	0	1556	Up	Up
sdp:17407:4294967295 SB(192.0.2.1)	BgpVpls	0	1556	Up	Up

*A:PE-3#

*A:PE-3# show service id 1 sdp

Services: Service Destination Points

SdpId	Type	Far End addr	Adm	Opr	I.Lbl	E.Lbl
17406:4294967294	Bgp*	192.0.2.2	Up	Up	262129	262128
17407:4294967295	Bgp*	192.0.2.1	Up	Up	262128	262130

Number of SDPs : 2

* indicates that the corresponding row element may have been truncated.

*A:PE-3#

PE-3 De-Multiplexer Label Verification

PE-3 also allocates the required parameters for PE-1 and PE-2 to calculate the egress service label required to reach PE-3.

This is verified using the following command on PE-1 to show the egress label towards PE-3 (192.0.2.3) (262128) where egress label towards PE-2 = 262126. The Label Base equals 262128 on PE-3 and 262126 on PE-2.

*A:PE-1# show service id 1 sdp

Services: Service Destination Points

SdpId	Type	Far End addr	Adm	Opr	I.Lbl	E.Lbl
17406:4294967294	Bgp*	192.0.2.2	Up	Up	262129	262126
17407:4294967295	Bgp*	192.0.2.3	Up	Up	262130	262128

Number of SDPs : 2

* indicates that the corresponding row element may have been truncated.

*A:PE-1#

This is also verified using the following command on PE-2 to show the egress label towards PE-3 (192.0.2.3) which is using auto-provisioned SDP 17406.

*A:PE-2# show service id 1 sdp

BGP VPLS PE Configuration

```
=====
Services: Service Destination Points
=====
SdpId          Type Far End addr    Adm    Opr      I.Lbl    E.Lbl
-----
17406:4294967294 Bgp* 192.0.2.3    Up     Up       262128   262129
17407:4294967295 Bgp* 192.0.2.1    Up     Up       262126   262129
-----
Number of SDPs : 2
-----
* indicates that the corresponding row element may have been truncated.
*A:PE-2#
```

This example has shown that for VPLS instance with 3 PEs, not all labels allocated by a PE will be used by remote PEs as de-multiplexer service labels. There will be some wastage of label space, so there is a necessity to choose ve-ids that keep this waste to a minimum.

The next example will show an even more wasteful use of labels by using a random choice of ve-ids.

BGP VPLS Using Pre-Provisioned SDP

It is possible to configure BGP-VPLS instances that use RSVP-TE transport tunnels. In this case, the SDP must be created with the MPLS LSPs mapped and with signaling set to BGP, as the service labels are signaled using BGP. The pseudowire template configured within the BGP-VPLS instance must use the **use-provisioned-sdp** keyword.

This example also examines the effect of using ve-ids that are not all within the same contiguous block.

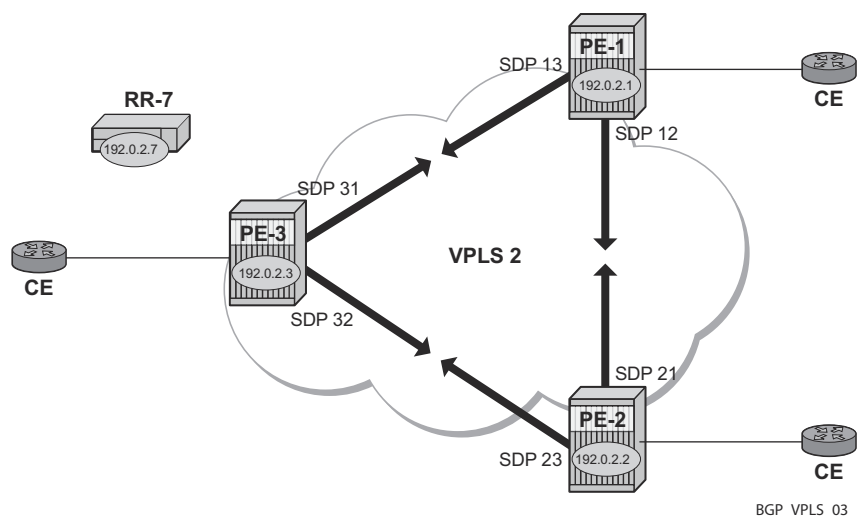


Figure 140: BGP VPLS Using Pre-Provisioned SDP

Figure 140 shows an example of a VPLS instance where SDPs are pre-provisioned with RSVP-TE signaled transport tunnels.

SDPs on PE-1

```
*A:PE-1# configure service
sdp 12 mpls create
description "SDP-PE-1-PE-2_RSVP_BGP"
signaling bgp
far-end 192.0.2.2
lsp "LSP-PE-1-PE-2"
no shutdown
exit
sdp 13 mpls create
description "SDP-PE-1-PE-3_RSVP_BGP"
signaling bgp
far-end 192.0.2.3
lsp "LSP-PE-1-PE-3"
no shutdown
```

BGP VPLS PE Configuration

```
exit
```

SDPs on PE-2

```
*A:PE-2# configure service
sdp 21 mpls create
description "SDP-PE-2-PE-1_RSVP_BGP"
signaling bgp
far-end 192.0.2.1
lsp "LSP-PE-2-PE-1"
no shutdown
exit
sdp 23 mpls create
description "SDP-PE-2-PE-3_RSVP_BGP"
signaling bgp
far-end 192.0.2.3
lsp "LSP-PE-2-PE-3"
no shutdown
exit
```

SDPs on PE-3

```
*A:PE-3# configure service
sdp 31 mpls create
description "SDP-PE-3-PE-1_RSVP_BGP"
signaling bgp
far-end 192.0.2.1
lsp "LSP-PE-3-PE-1"
no shutdown
exit
sdp 32 mpls create
description "SDP-PE-3-PE-2_RSVP_BGP"
signaling bgp
far-end 192.0.2.2
lsp "LSP-PE-3-PE-2"
no shutdown
exit
```

Note that pre-provisioned BGP-SDPs can also be used with BGP-VPLS. For reference, they are configured as follows:

```
*A:PE-3# configure service
sdp 332 mpls create
signaling bgp
far-end 192.0.2.2
no shutdown
exit
```

To create an SDP within a service that uses the RSVP transport tunnel, a pseudowire template is required that has the **use-provisioned-sdp** parameter set.

Once again, a split horizon group is included to prevent forwarding between pseudowires.

The pseudowire template must be provisioned on all PEs and looks like:

```
*A:PE-1# configure service
      pw-template 2 use-provisioned-sdp create
        split-horizon-group "VPLS-SHG"
      exit
    exit
```

The following output shows the configuration required for a BGP-VPLS service using a pseudowire template configured for using pre-provisioned RSVP-TE SDPs.

```
*A:PE-1# configure service
      vpls 2 customer 1 create
        bgp
          route-distinguisher 65536:2
          route-target export target:65536:2 import target:65536:2
          pw-template-binding 2
        exit
      exit
    bgp-vpls
      max-ve-id 100
      ve-name "PE-1"
      ve-id 1
    exit
    no shutdown
  exit
  sap 1/1/4:2.0 create
  exit
  no shutdown
exit
```

Note that the route distinguisher and route target extended community values for VPLS 2 are different from the ones in VPLS 1. The ve-id value for PE-1 can be the same as the one in VPLS 1, but these must be different within the same VPLS instance on the other PEs — PE-2 should not have ve-id = 1.

Similarly, on PE-2 the configuration example shows where the ve-id value is 20:

```
*A:PE-2# configure service
      vpls 2 customer 1 create
        bgp
          route-distinguisher 65536:2
          route-target export target:65536:2 import target:65536:2
          pw-template-binding 2
        exit
      exit
    bgp-vpls
      max-ve-id 100
      ve-name "PE-2"
      ve-id 20
    exit
    no shutdown
  exit
  sap 1/1/4:2.0 create
```

BGP VPLS PE Configuration

```
exit
no shutdown
exit
```

and on PE-3:

```
*A:PE-3# configure service
vpls 2 customer 1 create
  bgp
    route-distinguisher 65536:2
    route-target export target:65536:2 import target:65536:2
    pw-template-binding 2
  exit
exit
bgp-vpls
  max-ve-id 100
  ve-name "PE-3"
  ve-id 3
  exit
  no shutdown
exit
sap 1/1/4:2.0 create
exit
no shutdown
exit
```

Verify that the service is operationally up on PE-1.

```
*A:PE-1# show service id 2 base
=====
Service Basic Information
=====
Service Id       : 2                Vpn Id           : 0
Service Type     : VPLS
Name             : (Not Specified)
Description      : (Not Specified)
Customer Id      : 1                Creation Origin   : manual
Last Status Change: 07/13/2015 12:16:28
Last Mgmt Change : 07/13/2015 12:17:11
Etree Mode       : Disabled
Admin State      : Up               Oper State        : Up
MTU              : 1514             Def. Mesh VC Id   : 2
SAP Count        : 1               SDP Bind Count    : 2
Snd Flush on Fail : Disabled        Host Conn Verify   : Disabled
Propagate MacFlush: Disabled        Per Svc Hashing    : Disabled
Allow IP Intf Bind: Disabled
Def. Gateway IP   : None
Def. Gateway MAC  : None
Temp Flood Time   : Disabled        Temp Flood         : Inactive
Temp Flood Chg Cnt: 0
VSD Domain        : <none>
SPI load-balance  : Disabled
=====
Service Access & Destination Points
=====
Identifier                                     Type          AdmMTU  OprMTU  Adm  Opr
```

```

-----
sap:1/1/4:2.0                qinq          1522    1522    Up    Up
sdp:12:4294967293 S(192.0.2.2) BgpVpls       0      1556    Up    Up
sdp:13:4294967292 S(192.0.2.3) BgpVpls       0      1556    Up    Up
=====
*A:PE-1#

```

Note that the SDP-ids are the pre-provisioned SDPs.

For completeness, verify the service is operationally up on PE-2

```

*A:PE-2# show service id 2 base
=====
Service Basic Information
=====
Service Id      : 2                Vpn Id          : 0
Service Type    : VPLS
Name            : (Not Specified)
Description     : (Not Specified)
Customer Id     : 1                Creation Origin  : manual
Last Status Change: 07/13/2015 12:17:58
Last Mgmt Change  : 07/13/2015 12:18:53
Etree Mode     : Disabled
Admin State     : Up              Oper State       : Up
MTU             : 1514           Def. Mesh VC Id  : 2
SAP Count       : 1             SDP Bind Count   : 2
Snd Flush on Fail : Disabled     Host Conn Verify : Disabled
Propagate MacFlush: Disabled     Per Svc Hashing  : Disabled
Allow IP Intf Bind: Disabled
Def. Gateway IP : None
Def. Gateway MAC : None
Temp Flood Time : Disabled       Temp Flood       : Inactive
Temp Flood Chg Cnt: 0
VSD Domain     : <none>
SPI load-balance : Disabled
-----
Service Access & Destination Points
-----
Identifier                Type          AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/4:2.0             qinq          1522    1522    Up    Up
sdp:21:4294967293 S(192.0.2.1) BgpVpls       0      1556    Up    Up
sdp:23:4294967292 S(192.0.2.3) BgpVpls       0      1556    Up    Up
=====
*A:PE-2#

```

Verify service is operational on PE-3:

```

*A:PE-3# show service id 2 base
=====
Service Basic Information
=====
Service Id      : 2                Vpn Id          : 0
Service Type    : VPLS
Name            : (Not Specified)
Description     : (Not Specified)

```

BGP VPLS PE Configuration

```
Customer Id       : 1                      Creation Origin  : manual
Last Status Change: 07/13/2015 12:19:18
Last Mgmt Change  : 07/13/2015 12:19:19
Etree Mode       : Disabled
Admin State      : Up                      Oper State       : Up
MTU              : 1514                    Def. Mesh VC Id  : 2
SAP Count        : 1                      SDP Bind Count   : 2
Snd Flush on Fail: Disabled                Host Conn Verify : Disabled
Propagate MacFlush: Disabled              Per Svc Hashing  : Disabled
Allow IP Intf Bind: Disabled
Def. Gateway IP   : None
Def. Gateway MAC  : None
Temp Flood Time   : Disabled                Temp Flood       : Inactive
Temp Flood Chg Cnt: 0
VSD Domain       : <none>
SPI load-balance  : Disabled
```

----- Service Access & Destination Points

Identifier	Type	AdmMTU	OprMTU	Adm	Opr
sap:1/1/4:2.0	qinq	1522	1522	Up	Up
sdp:31:4294967293 S(192.0.2.1)	BgpVpls	0	1556	Up	Up
sdp:32:4294967292 S(192.0.2.2)	BgpVpls	0	1556	Up	Up

=====

*A:PE-3#

PE-1 De-Multiplexer Label Calculation

In the case of VPLS 1, all ve-ids are in the range of a single label block. In the case of VPLS 2, the ve-ids are in different blocks, for example, the ve-id 20 is in a different block to ve-ids 1 and 2.

As the label allocation is block-dependent, multiple labels blocks must be advertised by each PE to encompass this.

Consider PE-1's BGP update NLRIs.

```
*A:PE-1# show router bgp routes l2-vpn rd 65536:2 hunt
=====
BGP Router ID:192.0.2.1          AS:65536          Local AS:65536
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP L2VPN Routes
=====
---snip---
-----
RIB Out Entries
-----
Route Type      : VPLS
Route Dist.     : 65536:2
VeId            : 1                               Block Size      : 8
Base Offset     : 1                               Label Base      : 262120
Nexthop         : 192.0.2.1
To              : 192.0.2.7
Res. Nexthop    : n/a
Local Pref.     : 100                             Interface Name  : NotAvailable
Aggregator AS   : None                           Aggregator     : None
Atomic Aggr.    : Not Atomic                      MED            : 0
AIGP Metric     : None
Connector       : None
Community       : target:65536:2
                  l2-vpn/vrf-imp:Encap=19: Flags=none: MTU=1514: PREF=0
Cluster         : No Cluster Members
Originator Id   : None                           Peer Router Id  : 192.0.2.7
Origin          : IGP
AS-Path         : No As-Path
Route Tag       : 0
Neighbor-AS     : N/A
Orig Validation : N/A
Source Class    : 0                               Dest Class      : 0

Route Type      : VPLS
Route Dist.     : 65536:2
VeId            : 1                               Block Size      : 8
Base Offset     : 17                             Label Base      : 262112
Nexthop         : 192.0.2.1
To              : 192.0.2.7
Res. Nexthop    : n/a
Local Pref.     : 100                             Interface Name  : NotAvailable
Aggregator AS   : None                           Aggregator     : None
```

BGP VPLS PE Configuration

```
Atomic Aggr.   : Not Atomic           MED           : 0
AIGP Metric    : None
Connector      : None
Community      : target:65536:2
                12-vpn/vrf-imp:Encap=19: Flags=none: MTU=1514: PREF=0
Cluster        : No Cluster Members
Originator Id  : None                 Peer Router Id : 192.0.2.7
Origin         : IGP
AS-Path        : No As-Path
Route Tag      : 0
Neighbor-AS    : N/A
Orig Validation: N/A
Source Class   : 0                   Dest Class    : 0
```

```
-----
Routes : 8
=====
```

```
*A:PE-1#
```

Two NLRIs updates are sent to the route reflector, with the following label parameters:

1. LB = 262120, VBS = 8, VBO = 1
2. LB = 262112, VBS = 8, VBO = 17

PE-2 has a ve-id of 20. Applying the condition $VBO \leq \text{ve-id} < (VBO+VBS)$

Update 1: LB = 262120, VBS = 8, VBO = 1
VBO \leq ve-id for ve-id = 20 is TRUE
ve-id $< (VBO+VBS)$ for ve-id = 20 is FALSE.
PE-2 cannot choose a label from this block.

Update 2: LB = 262112, VBS = 8, VBO = 17
VBO \leq ve-id for ve-id = 20 is TRUE
ve-id $< (VBO+VBS)$ for ve-id = 20 is TRUE.
PE-2 chooses label $262112 + 20 - 17 = 262115$ (LB + veid - VBO)

The egress label chosen is verified by examining the egress label towards PE-1 (192.0.2.1) on PE-2.

```
*A:PE-2# show service id 2 sdp
```

```
=====
Services: Service Destination Points
```

```
=====
SdpId          Type Far End addr   Adm    Opr      I.Lbl    E.Lbl
-----
21:4294967293  Bgp* 192.0.2.1    Up     Up       262110   262115
23:4294967292  Bgp* 192.0.2.3    Up     Up       262112   262115
=====
```

```
Number of SDPs : 2
```

```
=====
* indicates that the corresponding row element may have been truncated.
```

```
*A:PE-2#
```


PE-3 has a ve-id of 3. Applying the condition $VBO \leq \text{ve-id} < (VBO + VBS)$

Update 1: LB = 262120, VBS = 8, VBO = 1
 $VBO \leq \text{ve-id}$ for ve-id = 3 is TRUE
 $\text{ve-id} < (VBO + VBS)$ for ve-id = 3 is TRUE.
 PE-3 chooses label $262120 + 3 - 1 = 262122$ (LB + veid - VBO)

Update 2: LB = 262112, VBS = 8, VBO = 17
 $VBO \leq \text{ve-id}$ for ve-id = 3 is FALSE
 $\text{ve-id} < (VBO + VBS)$ for ve-id = 3 is TRUE.
 PE-3 cannot choose a label from this block.

The egress label chosen is verified by examining the egress label towards PE-1 (192.0.2.1) on PE-3.

```
*A:PE-3# show service id 2 sdp
=====
Services: Service Destination Points
=====
SdpId          Type Far End addr   Adm   Opr       I.Lbl      E.Lbl
-----
31:4294967293  Bgp* 192.0.2.1      Up    Up        262120     262122
32:4294967292  Bgp* 192.0.2.2      Up    Up        262115     262112
-----
Number of SDPs : 2
-----
* indicates that the corresponding row element may have been truncated.
*A:PE-3#
```

To illustrate the allocation of label blocks by a PE, against the actual use of the same labels, consider the following. When BGP updates from each PE signal the multiplexer labels in blocks of eight, the allocated label values are added to the in-use pool. First check what label range can be allocated dynamically.

```
*A:PE-1# show router mpls-labels label-range
=====
Label Ranges
=====
Label Type      Start Label End Label   Aging      Available  Total
-----
Static          32          18431      -          18400     18400
Dynamic         18432       524287     0          505824    505856
  Seg-Route      0           0          -           0         505856
=====
*A:PE-1#
```

Verify which labels in the dynamic range are in use. The label pool of PE-1 can be verified as per the following output which shows labels used along with the associated protocol:

```
*A:PE-1# show router mpls-labels label 18432 524287 in-use
```

BGP VPLS PE Configuration

```
=====
MPLS Labels from 18432 to 524287 (In-use)
=====
Label                Label Type          Label Owner
-----
262112                dynamic             BGP
262113                dynamic             BGP
262114                dynamic             BGP
262115                dynamic             BGP
262116                dynamic             BGP
262117                dynamic             BGP
262118                dynamic             BGP
262119                dynamic             BGP
262120                dynamic             BGP
262121                dynamic             BGP
262122                dynamic             BGP
262123                dynamic             BGP
262124                dynamic             BGP
262125                dynamic             BGP
262126                dynamic             BGP
262127                dynamic             BGP
262128                dynamic             BGP
262129                dynamic             BGP
262130                dynamic             BGP
262131                dynamic             BGP
262132                dynamic             BGP
262133                dynamic             BGP
262134                dynamic             BGP
262135                dynamic             BGP
262136                dynamic             RSVP
262137                dynamic             RSVP
262138                dynamic             ILDP
262139                dynamic             ILDP
262140                dynamic             ILDP
262141                dynamic             ILDP
262142                dynamic             ILDP
262143                dynamic             ILDP
-----
In-use labels (Owner: All) in specified range : 32
In-use labels in entire range                 : 32
=====
*A:PE-1#
```

This shows that 24 labels have been allocated for use by BGP. Of this number, 16 labels have been allocated for use by PEs within VPLS 2 to communicate with PE-1, the blocks with label base 262112 and with label base 262120.

There are only two neighboring PEs within this VPLS instance, so only two labels will ever be used in the data plane for traffic destined to PE-1. These are 262115 and 262122. The remaining labels have no PE with the associated ve-id that can use them.

Once again, this case emphasizes that to reduce label wastage, contiguous ve-ids in the range (N..N+7) per VPLS should be chosen, where N>0.

Conclusion

BGP-VPLS allows the delivery of Layer 2 VPN services to customers where BGP is commonly used. The examples presented in this chapter show the configuration of BGP-VPLS together with the associated show outputs which can be used for verification and troubleshooting.

BGP Virtual Private Wire Services

In This Chapter

This section describes BGP Virtual Private Wire Service (VPWS) configurations.

Topics in this section include:

- [Applicability on page 906](#)
- [Overview on page 907](#)
- [Configuration on page 911](#)
- [Conclusion on page 936](#)

Applicability

This chapter is applicable to all of the 7750 SR, 7450 ESS and 7950 XRS series and was tested on Release 13.0.R3. There are no prerequisites for this configuration.

Introduction

There are currently two IETF standards for the provisioning of Virtual Private Wire Services (VPWS). RFC 4447, *Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)*, describes Label Distribution Protocol (LDP) VPWS, where VPWS pseudowires are signaled using LDP between Provider Edge (PE) Routers.

RFC 6624, *Layer 2 Virtual Private Networks Using BGP for Auto-Discovery and Signaling*, describes the use of Border Gateway Protocol (BGP) for signaling of pseudo-wires between such PEs.

The purpose of this chapter is to describe the configuration and troubleshooting for BGP VPWS.

Overview

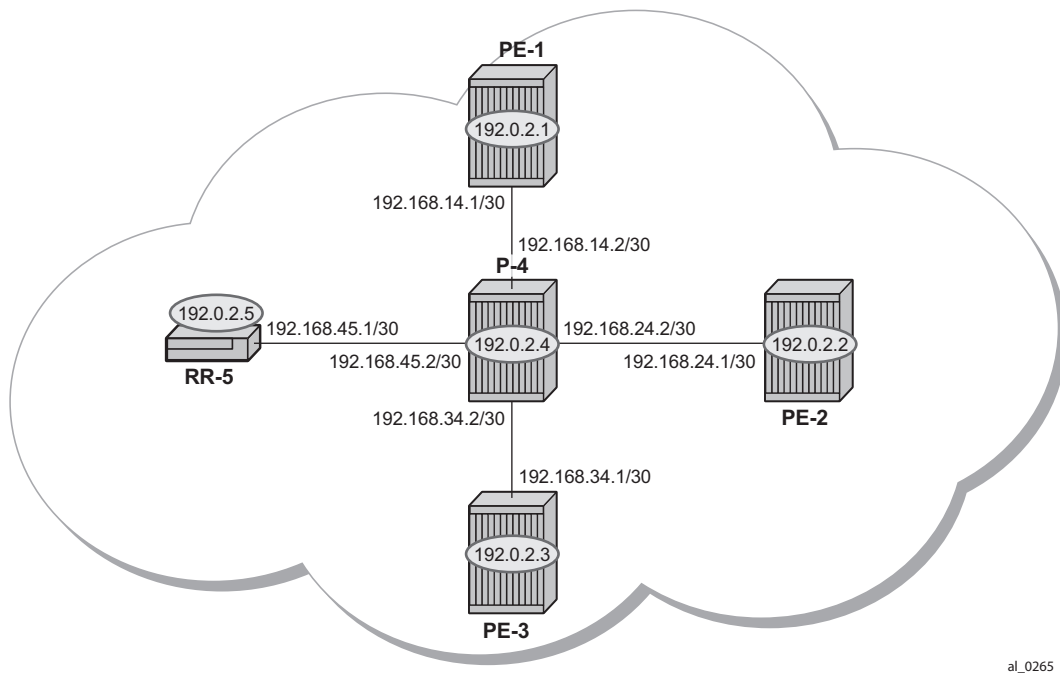


Figure 141: Network Topology

The network topology is displayed in [Figure 141](#). The setup uses five Service Router (SR) nodes located in the same Autonomous System (AS). There are three PE routers connected to a single P router and a Route Reflector (RR-5) for the AS. The Provider Edge routers are all BGP VPWS aware. The Provider (P) router is BGP VPWS unaware and also does not take part in the BGP process.

The following configuration tasks should be completed as a prerequisite:

- IS-IS or OSPF should be configured on each of the network interfaces between the PE/P routers and route reflector.
- MPLS should be configured on all interfaces between PE routers and P routers. It is not required between P-4 and RR-5.
- LDP should be configured on interfaces between PE and P routers. It is not required between P-4 and the RR-5.
- RSVP protocol should be configured on interfaces between PE and P routers. It is not required between P-4 and the RR-5.

BGP VPWS

In this architecture, a VPWS is a collection of two (or three in case of redundancy) BGP VPWS service instances present on different PEs in a provider network.

The PEs communicate with each other at the control plane level by means of BGP updates containing BGP VPWS Network Layer Reachability Information (NLRI). Each update contains enough information for a PE to determine the presence of other BGP VPWS instances on peering PEs and to set-up pseudowire connectivity for data flow between peers containing the same BGP VPWS service. Therefore, auto-discovery and pseudowire signaling is achieved using a single BGP update message.

Each PE with a BGP VPWS instance is identified by a VPWS Edge Identifier (VE-ID) and the presence of other BGP VPWS instances is determined using the exchange of standard BGP extended community route targets between PEs.

Each PE will advertise, via the route reflector, the presence of its BGP VPWS instance to all other PEs, along with a block of multiplexer labels (for BGP VPWS there is just one label per block) that can be used to communicate between each instance, plus a BGP next-hop that determines a labeled transport tunnel to be used between PEs.

Each BGP VPWS instance is configured with import and export route target extended communities for topology control, along with VE identification.

The following examples show the configuration of four BGP VPWS scenarios.

- Single homed BGP VPWS
 - using auto-provisioned SDPs
 - using pre-provisioned SDPs
- Dual homed BGP VPWS
 - with single pseudowire
 - with active/standby pseudowire

Configuration Tasks

The first step is to configure an MP-iBGP session between each of the PEs and the Route Reflector.

The configuration for PE-1 is as follows:

```
configure router autonomous-system 65536

configure router
    bgp
        group "INTERNAL"
            family l2-vpn
            type internal
            neighbor 192.0.2.5
            exit
        exit
    no shutdown
    exit
exit
```

The configuration for the other PE nodes is exactly the same. The IP addresses can be derived from [Figure 141](#).

The configuration for the Route Reflector (RR-5) is:

```
configure router autonomous-system 65536

configure router
    bgp
        group "INTERNAL"
            family l2-vpn
            type internal
            cluster 1.1.1.1
            neighbor 192.0.2.1
            exit
            neighbor 192.0.2.2
            exit
            neighbor 192.0.2.3
            exit
        exit
    no shutdown
    exit
exit
```

On RR-5, show that BGP sessions with each PE are established and have a negotiated l2-vpn address family capability.

```
*A:RR-5# show router bgp summary
=====
BGP Router ID:192.0.2.5      AS:65536      Local AS:65536
=====
BGP Admin State      : Up      BGP Oper State      : Up
Total Peer Groups    : 1      Total Peers          : 3
Total BGP Paths       : 5      Total Path Memory    : 920
Total IPv4 Remote Rts : 0      Total IPv4 Rem. Active Rts : 0
Total McIPv4 Remote Rts : 0    Total McIPv4 Rem. Active Rts : 0
Total McIPv6 Remote Rts : 0    Total McIPv6 Rem. Active Rts : 0
Total IPv6 Remote Rts : 0      Total IPv6 Rem. Active Rts : 0
Total IPv4 Backup Rts : 0      Total IPv6 Backup Rts  : 0

Total Supressed Rts   : 0      Total Hist. Rts      : 0
Total Decay Rts       : 0

Total VPN Peer Groups : 0      Total VPN Peers      : 0
Total VPN Local Rts   : 0
Total VPN-IPv4 Rem. Rts : 0    Total VPN-IPv4 Rem. Act. Rts : 0
Total VPN-IPv6 Rem. Rts : 0    Total VPN-IPv6 Rem. Act. Rts : 0
Total VPN-IPv4 Bkup Rts : 0    Total VPN-IPv6 Bkup Rts  : 0

Total VPN Supp. Rts   : 0      Total VPN Hist. Rts  : 0
Total VPN Decay Rts   : 0

Total L2-VPN Rem. Rts : 0      Total L2VPN Rem. Act. Rts : 0
Total MVPN-IPv4 Rem Rts : 0    Total MVPN-IPv4 Rem Act Rts : 0
Total MDT-SAFI Rem Rts : 0      Total MDT-SAFI Rem Act Rts : 0
Total MSPW Rem Rts    : 0      Total MSPW Rem Act Rts    : 0
Total RouteTgt Rem Rts : 0      Total RouteTgt Rem Act Rts : 0
Total McVpnIPv4 Rem Rts : 0     Total McVpnIPv4 Rem Act Rts : 0
Total MVPN-IPv6 Rem Rts : 0     Total MVPN-IPv6 Rem Act Rts : 0
Total EVPN Rem Rts     : 0      Total EVPN Rem Act Rts    : 0
Total FlowIpv4 Rem Rts : 0      Total FlowIpv4 Rem Act Rts : 0
Total FlowIpv6 Rem Rts : 0      Total FlowIpv6 Rem Act Rts : 0
=====
BGP Summary
=====
Neighbor
Description
AS PktRcvd InQ Up/Down State|Rcv/Act/Sent (Addr Family)
PktSent OutQ
-----
192.0.2.1
65536 3 0 00h00m21s 0/0/0 (L2VPN)
3 0
192.0.2.2
65536 3 0 00h00m21s 0/0/0 (L2VPN)
3 0
192.0.2.3
65536 3 0 00h00m21s 0/0/0 (L2VPN)
3 0
-----
*A:RR-5#
```

Configuration

Pseudowire Templates

BGP VPWS utilizes pseudowire (PW) templates to dynamically instantiate SDP bindings for a given service to signal the egress service de-multiplexer labels used by remote PEs to reach the local PE.

The template determines the signaling parameters of the pseudowire, such as vc-type, vlan-vc-tag, hash-label, filters, etc. The following parameters are recognized by BGP VPWS; the remainder is ignored.

The following commands are supported parameters:

```
configure
service
  [no] pw-template policy-id [use-provisioned-sdp] [create]
  accounting-policy acct-policy-id
  no accounting-policy
  [no] collect-stats
  egress
    filter ipv6 ipv6-filter-id
    filter ip ip-filter-id
    filter mac mac-filter-id
    no filter [ip ip-filter-id] [mac mac-filter-id] [ipv6 ipv6-filter-id]
    qos network-policy-id port-redirect-group queue-group-name [instance instance-id]
    no qos
  [no] force-vlan-vc-forwarding
  hash-label [signal-capability]
  no hash-label
  ingress
    filter ipv6 ipv6-filter-id
    filter ip ip-filter-id
    filter mac mac-filter-id
    no filter [ip ip-filter-id] [mac mac-filter-id] [ipv6 ipv6-filter-id]
    qos network-policy-id fp-redirect-group queue-group-name instance instance-id
    no qos
  [no] sdp-exclude group-name
  [no] sdp-include group-name
  vc-type {ether | vlan}
  vlan-vc-tag 0..4094
  no vlan-vc-tag
```

Note that:

- The encapsulation type in the Layer-2 extended community is either 4 (Ethernet VLAN tagged mode) or 5 (Ethernet raw mode), depending on the vc-type parameter.
- The **force-vlan-vc-forwarding** function will add a tag (equivalent to vc-type vlan) and will allow for customer QoS transparency (dot1p + Drop Eligibility (DE) bits).

The MPLS transport tunnel between PEs can be signaled using LDP or RSVP-TE.

LDP-based SDPs can be automatically instantiated or pre-provisioned. RSVP-TE-based SDPs have to be pre-provisioned. If pre-provisioned pseudowires should be used, the pw-template must be created with the **use-provisioned-sdp** parameter.

```
*A:PE-1# configure service pw-template
- [no] pw-template <policy-id> [use-provisioned-sdp] [create]
```

Pseudowire Templates for Auto-SDP Creation using LDP

In order to use an LDP transport tunnel for data flow between PEs, it is necessary for link layer LDP to be configured between all PEs/Ps so that a transport label for each PE's system interface is available. For example, on PE-1:

```
configure router
  ldp
    interface-parameters
      interface "int-PE-1-P-4"
    exit
  exit
```

Using this mechanism, SDPs can be auto-instantiated with SDP-ids starting at the higher end of the SDP numbering range, such as 17407. Any subsequent SDPs created use SDP-ids decrementing from this value.

A pseudowire template is required. The example below is created using the default values:

```
configure service
  pw-template 1 create
  exit
```

Pseudowire Templates for Provisioned SDPs using RSVP-TE

RSVP-TE LSPs need to be created between the PE routers on which provisioned SDPs will be used as prerequisite.

The MPLS interface and LSP configuration for PE-1 are:

```
configure router
  mpls
    interface "system"
  exit
  interface "int-PE-1-P-4"
```

```

exit
path "dyn"
    no shutdown
exit
lsp "LSP-PE-1-PE-2"
    to 192.0.2.2
    primary "dyn"
    exit
    no shutdown
exit
lsp "LSP-PE-1-PE-3"
    to 192.0.2.3
    primary "dyn"
    exit
    no shutdown
exit
no shutdown

```

The MPLS and LSP configuration for PE-2 are similar to that of PE-1 with the appropriate interfaces and LSP names configured.

To use an RSVP-TE tunnel as transport between PEs, it is necessary to bind the RSVP-TE LSP between PEs to an SDP.

The SDP creation on PE-1 towards PE-2 is shown below; similar SDPs are required on each PE to the remote PEs in the service where provisioned SDPs are to be used (only on PE-2 in the following example).

```

configure service
    sdp 12 mpls create
        description "SDP-PE-1-PE-2_RSVP_BGP"
        signaling bgp
        far-end 192.0.2.2
        lsp "LSP-PE-1-PE-2"
        no shutdown
    exit

```

Note that the **signaling bgp** parameter is required. BGP VPWS instances using BGP VPWS signaling are able to use these SDPs. Conversely, SDPs that are bound to RSVP-based LSPs with signaling set to the default value of “tldp” will not be used as SDPs within BGP VPWS.

Single Homed BGP VPWS using Auto-Provisioned SDPs

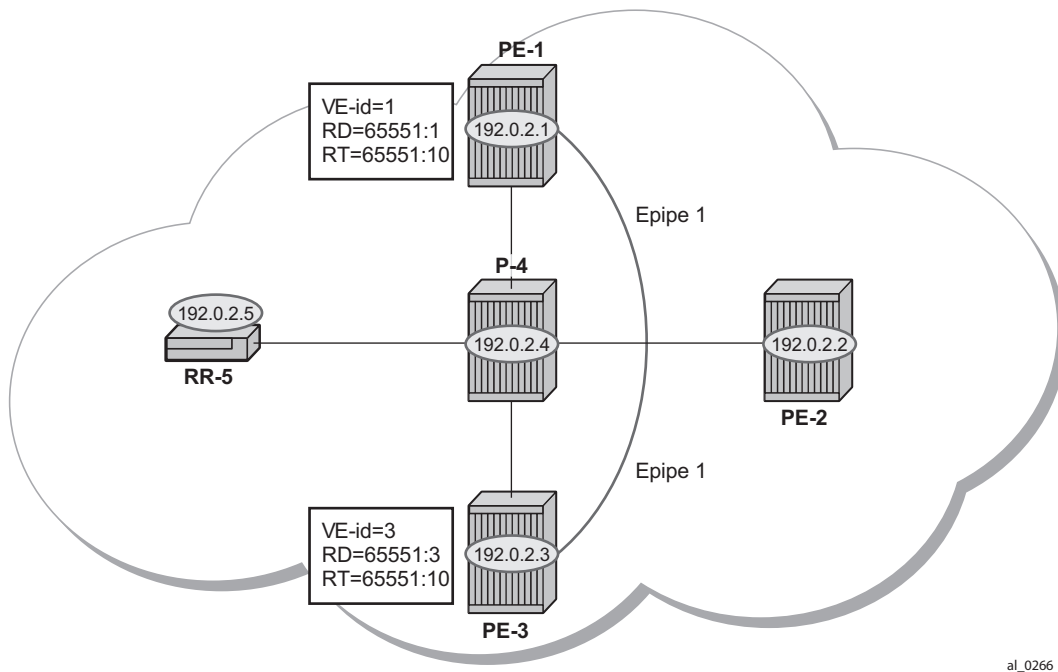


Figure 142: Single Homed BGP VPWS using Auto-Provisioned SDPs

Figure 142 shows a schematic of a single homed BGP VPWS between PE-1 and PE-3 where SDPs are auto-provisioned. In this case, the transport tunnels are LDP signaled.

The following shows the configuration required on PE-1 for a BGP VPWS service using a pseudowire template configured for auto-provisioning of SDPs.

```
*A:PE-1# configure service
pw-template 1 create
vc-type vlan
exit
epipe 1 customer 1 create
bgp
route-distinguisher 65551:1
route-target export target:65551:10 import target:65551:10
pw-template-binding 1
exit
exit
bgp-vpws
ve-name "PE-1"
ve-id 1
exit
remote-ve-name "PE-3"
```

```

        ve-id 3
    exit
    no shutdown
exit
sap 1/1/4:100 create
exit
    no shutdown
exit

```

The **bgp** context specifies parameters that are required for BGP VPWS.

Within the **bgp** context, parameters are configured that are used by the neighboring PEs to determine the membership of a given BGP VPWS; in other words, the auto-discovery of PEs in the same BGP VPWS, the route-distinguisher is configured, along with the route target extended communities. Route target communities are used to determine membership of a given BGP VPWS. Note that the import and export route targets at the BGP level are mandatory. The pw-template binding is then applied and its parameters are used for both the routes sent by this PE and the received routes matching the route target value.

Within the **bgp-vpws** context, the signaling parameters are also configured. These determine the service labels required for the data plane of the VPWS instance.

The VPWS Edge ID (VE-ID) is a numerical value assigned to each PE within a BGP VPWS. This value must be unique for a given BGP VPWS, with the exception of multi-homed scenarios, where two dual-homed PEs can have the same VE-ID and are distinguishable by the site preference (or by the tie breaking rules from the multi-homing draft RFC).

It is also worth noting that changes to the pseudowire template are not taken into account once the pseudowire has been set up (changes of route-target are refreshed though). PW-templates can be re-evaluated with the **tools perform service eval-pw-template** command. The **eval-pw-template** checks if all of the bindings using this pw-template policy are still meant to be used this policy. If the template has changed and allow-service-impact is TRUE, then the old binding is removed and it is re-added using the new template.

```

*A:PE-1# tools perform service eval-pw-template 1
eval-pw-template succeeded for Svc 1 Tx L2 ExtComm, Policy 1
eval-pw-template succeeded for Svc 1 17407:4294967295 Policy 1
*A:PE-1#

```

VE-ID and BGP Label Allocations

For a point-to-point VPWS, there are only two members within the BGP VPWS service, so only one label entry is required by each remote service. For dual-homed scenarios, there are two labels for the redundant site, one from each dual-homed PE.

Each PE allocates a label per BGP VPWS instance for the remote PEs, so it signals blocks with one label. It achieves this by advertising three parameters in a BGP update message.

- A Label Base (LB) which is the lowest label in the block.
- A VE Block size (VBS) which is always 1 and cannot be changed.
- A VE Base Offset (VBO) corresponding to the first label in the label block.

PE-3 Service Creation

On PE-3 create a BGP VPWS service using pseudowire template 1. PE-3 has been allocated a VE-ID of 3. For completeness, the pw-template is also shown.

```
*A:PE-3# configure service
      pw-template 1 create
        vc-type vlan
      exit
      epipe 1 customer 1 create
        bgp
          route-distinguisher 65551:3
          route-target export target:65551:10 import target:65551:10
          pw-template-binding 1
        exit
      exit
      bgp-vpws
        ve-name "PE-3"
        ve-id 3
      exit
      remote-ve-name "PE-1"
        ve-id 1
      exit
      no shutdown
    exit
    sap 1/1/4:101 create
    exit
    no shutdown
  exit
```


PE-1 Service Operation Verification

Verify that the BGP VPWS service is enabled on PE-1.

```
*A:PE-1# show service id 1 bgp-vpws
=====
BGP VPWS Information
=====
Admin State           : Enabled
VE Name               : PE-1           VE Id               : 1
PW Tmpl used          : 1

Remote-Ve Information
-----
Remote VE Name        : PE-3           Remote VE Id         : 3
=====
*A:PE-1#
```

Verify the BGP information used by the BGP VPWS service on PE-1.

```
*A:PE-1# show service id 1 bgp
=====
BGP Information
=====
Route Dist            : 65551:1
Oper Route Dist       : 65551:1
Oper RD Type          : configured
Rte-Target Import     : 65551:10       Rte-Target Export: 65551:10

PW-Template Id        : 1
BFD Template          : None
BFD-Enabled           : no             BFD-Encap           : ipv4
Import Rte-Tgt        : None
=====
*A:PE-1#
```

Verify that the service is operationally up on PE-1.

```
*A:PE-1# show service id 1 base
=====
Service Basic Information
=====
Service Id           : 1               Vpn Id              : 0
Service Type         : Epipe
Name                 : (Not Specified)
Description           : (Not Specified)
Customer Id          : 1               Creation Origin      : manual
Last Status Change   : 07/06/2015 13:30:00
Last Mgmt Change     : 07/06/2015 13:30:00
Test Service         : No
Admin State          : Up              Oper State           : Up
MTU                  : 1514
Vc Switching         : False
```

PE-1 Service Operation Verification

```
SAP Count          : 1                SDP Bind Count      : 1
Per Svc Hashing    : Disabled
Force QTag Fwd     : Disabled
```

----- Service Access & Destination Points

Identifier	Type	AdmMTU	OprMTU	Adm	Opr
sap:1/1/4:100	q-tag	1578	1578	Up	Up
sdp:17407:4294967295 SB(192.0.2.3)	BgpVpws	0	1552	Up	Up

=====

```
*A:PE-1#
```

The SAP and SDP are all operationally up. Note that the indication “SB” next to the SDP-id signify “Spoke” and “BGP”.

Further verification can be seen below where the ingress label for PE-3, that is, the labels used by PE-1 are shown.

```
*A:PE-1# show service id 1 sdp
=====
Services: Service Destination Points
=====
```

SdpId	Type	Far End addr	Adm	Opr	I.Lbl	E.Lbl
17407:4294967295	BVws	192.0.2.3	Up	Up	262137	262137

```
-----
Number of SDPs : 1
-----
*A:PE-1#
```

The following debug output from PE-1 shows the BGP VPWS NLRI update for Epipe 1 sent by PE-1 to the route reflector (192.0.2.5). This update will then be received by the other PEs.

```
*A:PE-1# debug router bgp update

*A:PE-1# configure log
    log-id 2
        from debug-trace
        to memory
        no shutdown
    exit

*A:PE-1# show log log-id 2
---snip---
```

```
4 2015/07/06 13:30:17.85 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.5
"Peer 1: 192.0.2.5: UPDATE
Peer 1: 192.0.2.5 - Send BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 76
    Flag: 0x90 Type: 14 Len: 32 Multiprotocol Reachable NLRI:
        Address Family L2VPN
```

```

      NextHop len 4 NextHop 192.0.2.1
      [VPLS/VPWS] preflen 21, veid: 1, vbo: 3, vbs: 1, label-base: 262137, RD 65551:1,
csv: 0x00000000, type 1, len 1,
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 16 Extended Community:
    target:65551:10
    l2-vpn/vrf-imp:Encap=4: Flags=none: MTU=1514: PREF=0
"

```

Note the presence of the control flags within the extended community which indicate the status of the BGP VPWS instance.

The control flags are described below:

```

  0 1 2 3 4 5 6 7
  +---+---+---+---+
  |D|A|F|Z|Z|Z|C|S| (Z = MUST Be Zero)
  +---+---+---+---+

```

- D: access circuit down indicator. D is 1 if all access circuits are down, otherwise D is 0.
- A: automatic site id allocation, which is not supported. This is ignored on receipt and set to 0 on sending.
- F: MAC flush indicator, this relates to VPLS. This is set to 0 and ignored on receipt.
- C: presence of a control word. Control word usage is not supported. This is set to 0 on sending (control word not present) and if a non-zero value is received (indicating a control word is required) the pseudowire will not be created.
- S: sequenced delivery. Sequenced delivery is not supported. This is set to 0 on sending (no sequenced delivery) and if a non-zero value is received (indicating sequenced delivery required) the pseudowire will not be created.

The BGP VPWS NLRI is based on that defined for BGP VPLS but is extended with a circuit status vector (CSV). The circuit status vector is used to indicate the status of both the SAP and the spoke-SDP within the local service. As the VE block size used is 1, the most significant bit in the circuit status vector TLV value will be set to 1 if either the SAP or spoke-SDP is down; otherwise, it will be set to 0.

```

6 2015/07/06 13:34:40.86 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.5
"Peer 1: 192.0.2.5: UPDATE
Peer 1: 192.0.2.5 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 76
  Flag: 0x90 Type: 14 Len: 32 Multiprotocol Reachable NLRI:
    Address Family L2VPN
    NextHop len 4 NextHop 192.0.2.1

```

PE-3 Service Operation Verification

```
[VPLS/VPWS] preflen 21, veid: 1, vbo: 3, vbs: 1, label-base: 262137, RD 65551:1,
csv: 0x00000080, type 1, len 1,
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 16 Extended Community:
    target:65551:10
    12-vpn/vrf-imp:Encap=4: Flags=D: MTU=1514: PREF=0
"
```

After shutting down the local SAP, the CSV has the most-significant bit set to 1 (0x80). The BGP VPWS update can be shown using the following command:

```
*A:PE-1# show service l2-route-table bgp-vpws detail
=====
Services: L2 Bgp-Vpws Route Information - Summary
=====

Svc Id       : 1
VeId         : 3
PW Temp Id   : 1
RD           : *65551:3
Next Hop     : 192.0.2.3
State (D-Bit) : up(0)
Path MTU     : 1514
Control Word : 0
Seq Delivery  : 0
Status       : active
Tx Status    : active
CSV          : 0
Preference   : 0
Sdp Bind Id  : 17407:4294967295
=====
*A:PE-1#
```

PE-3 Service Operation Verification

Similar to PE-1, the service operation should be validated on PE-3.

Single Homed BGP VPWS using Pre-Provisioned SDP

It is possible to configure BGP VPWS instances that use RSVP-TE transport tunnels. In this case, the SDPs must be created with the MPLS LSPs mapped and with the signaling set to BGP, as the service labels are signaled using BGP. The pw-template configured within the BGP VPWS instance must use the keyword **use-provisioned-sdp**.

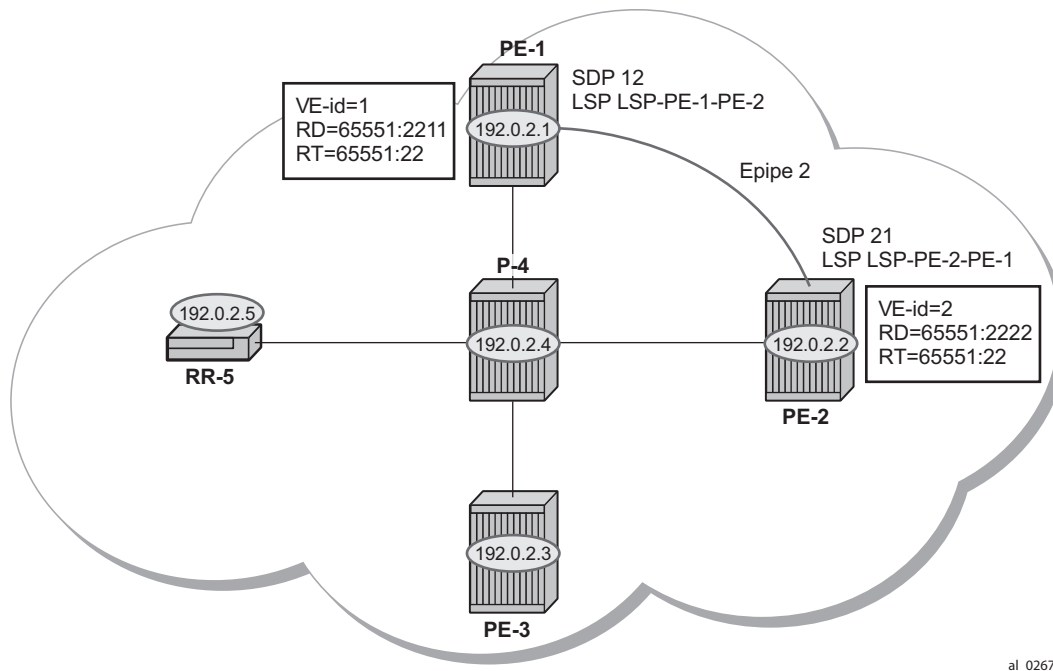


Figure 143: Single Homed BGP VPWS using Pre-Provisioned SDP

Figure 143 shows a schematic of a BGP VPWS where SDPs are pre-provisioned with RSVP-TE signaled transport tunnels.

SDP on PE-1

```
*A:PE-1# configure service
sdp 12 mpls create
description "SDP-PE-1-PE-2_RSVP_BGP"
signaling bgp
far-end 192.0.2.2
lsp "LSP-PE-1-PE-2"
no shutdown
exit
```

Single Homed BGP VPWS using Pre-Provisioned SDP

SDP on PE-2

```
*A:PE-2# configure service
    sdp 21 mpls create
        description "SDP-PE-2-PE-1_RSVP_BGP"
        signaling bgp
        far-end 192.0.2.1
        lsp "LSP-PE-2-PE-1"
        no shutdown
    exit
```

To create a spoke SDP within a service that uses the RSVP-TE transport tunnel, a pseudowire template is required that has the “use-provisioned-sdp” parameter set.

The pw-template is provisioned on both PEs as follows:

```
*A:PE-1# configure service
    pw-template 2 use-provisioned-sdp create
    exit
```

The following output shows the configuration required for a BGP VPWS service using a pseudowire template configured for using pre-provisioned RSVP-TE SDPs.

```
*A:PE-1# configure service
    epipe 2 customer 1 create
        bgp
            route-distinguisher 65551:21
            route-target export target:65551:22 import target:65551:22
            pw-template-binding 2
        exit
    exit
    bgp-vpws
        ve-name "PE-1"
        ve-id 1
    exit
        remote-ve-name "PE-2"
        ve-id 2
    exit
    no shutdown
    exit
    sap 1/1/4:200 create
    exit
    no shutdown
```

The route distinguisher and route target extended community values for Epipe 2 are different from that in Epipe 1. This is to differentiate between the two as their visibility is global within the BGP domain. The VE-ID values can be reused in each Epipe instance, as long as they are unique within the instance.

Similarly, on PE-2 the configuration is as seen below, where the VE-ID is 2:

```
*A:PE-2# configure service
      epipe 2 customer 1 create
        bgp
          route-distinguisher 65551:22
          route-target export target:65551:22 import target:65551:22
          pw-template-binding 2
          exit
        exit
      bgp-vpws
        ve-name "PE-2"
        ve-id 2
        exit
        remote-ve-name "PE-1"
        ve-id 1
        exit
        no shutdown
      exit
    sap 1/1/4:200 create
    exit
    no shutdown
```

Verify that the service is operationally up on PE-1.

```
*A:PE-1# show service id 2 base
=====
Service Basic Information
=====
Service Id      : 2                Vpn Id          : 0
Service Type    : Epipe
Name            : (Not Specified)
Description     : (Not Specified)
Customer Id     : 1                Creation Origin  : manual
Last Status Change: 07/06/2015 13:37:21
Last Mgmt Change  : 07/06/2015 13:37:21
Test Service    : No
Admin State     : Up               Oper State      : Up
MTU             : 1514
Vc Switching    : False
SAP Count       : 1                SDP Bind Count  : 1
Per Svc Hashing : Disabled
Force QTag Fwd  : Disabled

-----
Service Access & Destination Points
-----
Identifier                               Type      AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/4:200                           q-tag     1578    1578    Up   Up
sdp:12:4294967294 S(192.0.2.2)          BgpVpws   0       1552    Up   Up
=====
*A:PE-1#
```

Note that the SDP-id is the pre-provisioned SDP 12.

Single Homed BGP VPWS using Pre-Provisioned SDP

For completeness, verify the service is operationally up on PE-2.

```
*A:PE-2# show service id 2 base
=====
Service Basic Information
=====
Service Id       : 2                Vpn Id           : 0
Service Type     : Epipe
Name             : (Not Specified)
Description      : (Not Specified)
Customer Id      : 1                Creation Origin  : manual
Last Status Change: 07/06/2015 13:38:47
Last Mgmt Change : 07/06/2015 13:38:47
Test Service     : No
Admin State      : Up               Oper State       : Up
MTU              : 1514
Vc Switching     : False
SAP Count        : 1                SDP Bind Count   : 1
Per Svc Hashing  : Disabled
Force QTag Fwd   : Disabled

-----
Service Access & Destination Points
-----
Identifier                               Type      AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/4:200                           q-tag     1578    1578    Up    Up
sdp:21:4294967295 S(192.0.2.1)          BgpVpws   0       1552    Up    Up
=====
*A:PE-2#
```

The SDP-id used is the pre-provisioned SDP 21.

Dual Homed BGP VPWS with Single Pseudowire

For access redundancy, an Epipe using a BGP VPWS service can be configured as dual-homed, as described in *draft-ietf-l2vpn-vpls-multihoming-03*. It can be configured with a single pseudowire setup, where the redundant pseudowire is not created until the initially active pseudowire is removed.

The following diagram shows a setup where an Epipe is configured on each PE. Site B is dual-homed to PE-1 and PE-3 with the remote PE-2 connected to site A; each site connection uses a SAP. A single pseudowire using Ethernet Raw Mode encapsulation connects PE-2 to PE-1 or PE-3 (but not both at the same time). The pseudowire is signaled using BGP VPWS over a tunnel LSP between the PEs.

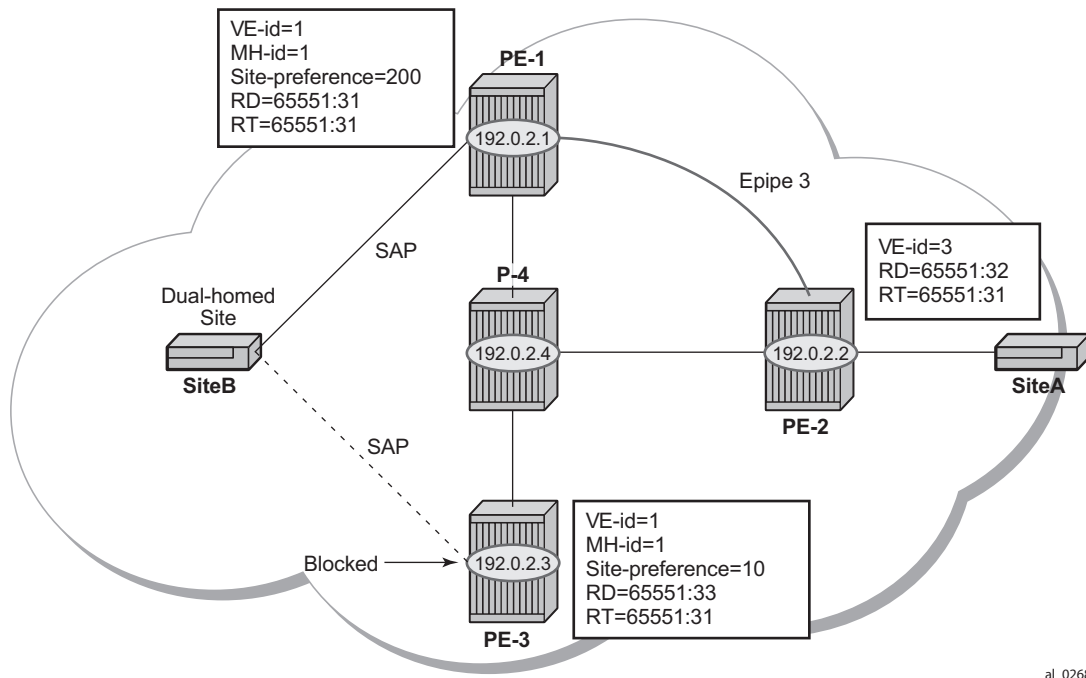


Figure 144: Dual Homed BGP VPWS with Single Pseudowire

BGP multi-homing is configured for the dual-homed site B using a site-id=1. The site-preference on PE-1 is set to 200 and to 10 on PE-3, this ensures that PE-1 will be the site's designated forwarder and the pseudowire from PE-2 will be created to PE-1 when PE-1 is fully operational (no pseudowire is created on PE-2 to PE-3). If PE-1 fails, or the multi-homing site fails over to PE-3, then the pseudowire from PE-2 to PE-1 will be removed and a new pseudowire will be created from PE-2 to PE-3.

Dual Homed BGP VPWS with Single Pseudowire

PE-1 Configuration

```
*A:PE-1# configure service
pw-template 3 create
exit
epipe 3 customer 1 create
  bgp
    route-distinguisher 65551:31
    route-target export target:65551:31 import target:65551:31
    pw-template-binding 3
  exit
exit
bgp-vpws
  ve-name "PE-1"
  ve-id 1
  exit
  remote-ve-name "PE-2"
  ve-id 2
  exit
  no shutdown
exit
site "SITEB" create
  site-id 1
  sap 1/1/4:99
  site-preference 200
  no shutdown
exit
sap 1/1/4:99 create
exit
no shutdown
exit
```

PE-3 Configuration

```
*A:PE-3# configure service
pw-template 3 create
exit
epipe 3 customer 1 create
  bgp
    route-distinguisher 65551:33
    route-target export target:65551:31 import target:65551:31
    pw-template-binding 3
  exit
exit
bgp-vpws
  ve-name "PE-3"
  ve-id 1
  exit
  remote-ve-name "PE-2"
  ve-id 2
  exit
  no shutdown
exit
site "SITEB" create
  site-id 1
  sap 1/1/4:99
  site-preference 10
```

```

        no shutdown
    exit
    sap 1/1/4:99 create
    exit
    no shutdown
exit

```

Note that in the above configurations, the remote-ve-name for PE-2 uses VE-ID 2 on both PE-1 and PE-3.

PE-2 Configuration

```

*A:PE-2# configure service
    pw-template 3 create
    exit
    epipe 3 customer 1 create
        bgp
            route-distinguisher 65551:32
            route-target export target:65551:31 import target:65551:31
            pw-template-binding 3
        exit
    exit
    bgp-vpws
        ve-name "PE-2"
        ve-id 2
    exit
    remote-ve-name "PE-1 or PE-3"
        ve-id 1
    exit
    no shutdown
exit
    sap 1/1/4:99 create
    exit
    no shutdown
exit

```

On PE-2, the remote-ve-name is configured as PE-1 or PE-3; this is because both of these PEs are configured with VE-ID 1.

As a result of this configuration, observe that on PE-2, there are multiple route entries for Route-Target 65551:31. In the BGP routing table, there are two entries per partner PE, one for the BGP-MH update (with site-id=1) and the other for the BGP-VPWS update (with VE-ID=1).

```

*A:PE-2# show router bgp routes l2-vpn rd 65551:31
=====
BGP Router ID:192.0.2.2      AS:65536      Local AS:65536
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP L2VPN Routes

```

Dual Homed BGP VPWS with Single Pseudowire

```

=====
Flag  RouteType      Prefix              MED
      RD              SiteId              Label
      Nexthop         VeId              BlockSize LocalPref
      As-Path         BaseOffset         vplsLabelBase

-----
u*>i  MultiHome      -                  -          0
      65551:31       1                  -          -
      192.0.2.1      -                  -          200
      No As-Path     -                  -
u*>i  VPWS            -                  -          0
      65551:31       -                  -          -
      192.0.2.1      1                  1          200
      No As-Path     2                  262135

-----
Routes : 2
=====
*A:PE-2#
*A:PE-2# show router bgp routes l2-vpn rd 65551:33
=====
BGP Router ID:192.0.2.2      AS:65536      Local AS:65536
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP L2VPN Routes
=====
Flag  RouteType      Prefix              MED
      RD              SiteId              Label
      Nexthop         VeId              BlockSize LocalPref
      As-Path         BaseOffset         vplsLabelBase

-----
u*>i  MultiHome      -                  -          0
      65551:33       1                  -          -
      192.0.2.3      -                  -          10
      No As-Path     -                  -
u*>i  VPWS            -                  -          0
      65551:33       -                  -          -
      192.0.2.3      1                  1          10
      No As-Path     2                  262136

-----
Routes : 2
=====
*A:PE-2#

The route to PE-1 has the higher site preference, so it is selected as the target for the pseudowire.

*A:PE-2# show service l2-route-table bgp-vpws detail
=====
Services: L2 Bgp-Vpws Route Information - Summary
=====

---snip---

Svc Id      : 3

```

```

VeId      : 1
PW Temp Id : 3
RD        : *65551:31
Next Hop   : 192.0.2.1
State (D-Bit) : up(0)
Path MTU    : 1514
Control Word : 0
Seq Delivery : 0
Status      : active
Tx Status    : active
CSV         : 0
Preference  : 200
Sdp Bind Id : 17407:4294967294

```

```

=====
*A:PE-2#

```

After disabling the SAP in the service on PE-1, BGP UPDATE messages are received. The VPLS/VPWS message received on PE-2 from PE-1 shows in the CSV that the access circuit is down (the CSV has the most-significant bit set to 1 (0x80)), so PE-2 selects the update from PE-3 to create the pseudowire. The BGP-MH update received by PE-2 from PE-1 also shows that the local site is down as indicated by the flags=D.

Note in the debug output below,

- BGP MH (multi-homing) entry uses encap-type=19.
- BGP VPWS entry uses encap-type=5 (Ethernet raw mode).

```

37 2015/07/06 13:41:57.10 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.5
"Peer 1: 192.0.2.5: UPDATE
Peer 1: 192.0.2.5 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 90
  Flag: 0x90 Type: 14 Len: 32 Multiprotocol Reachable NLRI:
    Address Family L2VPN
    NextHop len 4 NextHop 192.0.2.1
    [VPLS/VPWS] preflen 21, veid: 1, vbo: 2, vbs: 1, label-base: 262135, RD
65551:31, csv: 0x00000080, type 1, len 1,
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 0
  Flag: 0x80 Type: 9 Len: 4 Originator ID: 192.0.2.1
  Flag: 0x80 Type: 10 Len: 4 Cluster ID:
    1.1.1.1
  Flag: 0xc0 Type: 16 Len: 16 Extended Community:
    target:65551:31
    12-vpn/vrf-imp:Encap=5: Flags=D: MTU=1514: PREF=200
"

36 2015/07/06 13:41:57.10 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.5
"Peer 1: 192.0.2.5: UPDATE
Peer 1: 192.0.2.5 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 86
  Flag: 0x90 Type: 14 Len: 28 Multiprotocol Reachable NLRI:

```

Dual Homed BGP VPWS with Single Pseudowire

```
Address Family L2VPN
NextHop len 4 NextHop 192.0.2.1
[MH] site-id: 1, RD 65551:31
Flag: 0x40 Type: 1 Len: 1 Origin: 0
Flag: 0x40 Type: 2 Len: 0 AS Path:
Flag: 0x80 Type: 4 Len: 4 MED: 0
Flag: 0x40 Type: 5 Len: 4 Local Preference: 0
Flag: 0x80 Type: 9 Len: 4 Originator ID: 192.0.2.1
Flag: 0x80 Type: 10 Len: 4 Cluster ID:
1.1.1.1
Flag: 0xc0 Type: 16 Len: 16 Extended Community:
target:65551:31
l2-vpn/vrf-imp:Encap=19: Flags=D: MTU=0: PREF=200
"
```

The result can be shown on PE-2 as now the spoke SDP is up (active) to PE-3.

```
*A:PE-2# show service l2-route-table bgp-vpws detail
=====
Services: L2 Bgp-Vpws Route Information - Summary
=====

---snip---

Svc Id       : 3
VeId         : 1
PW Temp Id   : 3
RD           : *65551:33
Next Hop     : 192.0.2.3
State (D-Bit) : up(0)
Path MTU     : 1514
Control Word  : 0
Seq Delivery  : 0
Status       : active
Tx Status    : active
CSV          : 0
Preference   : 10
Sdp Bind Id  : 17407:4294967293
=====
*A:PE-2#
```

Dual Homed BGP VPWS with Active/Standby Pseudowire

The second method for BGP VPWS pseudowire redundancy is an active/standby configuration. While in the solution with one pseudowire, the redundant nodes use the same VE-ID for the remote PE and different preferences; in the active/standby solution, the redundant nodes use different VE-IDs for the remote PE and different preferences. The node connecting to both pseudowires (PE-2 in this example) has both remote VE-IDs configured. This allows for faster failover as the standby pseudowire is instantiated in addition to the active pseudowire. If more than two applicable BGP updates are received, at most one standby pseudowire is created (based on the BGP VPWS tie breaking rules).

Figure 145 shows a setup where an Epipe is configured on each PE. Site B is dual-homed to PE-1 and PE-3 with the remote PE-2 connected to site A; each site connection uses a SAP. The active/standby pseudowires using Ethernet Raw Mode encapsulation connect PE-2 to PE-1 and PE-3. The pseudowires are signaled using BGP VPWS over tunnel LSPs between the PEs.

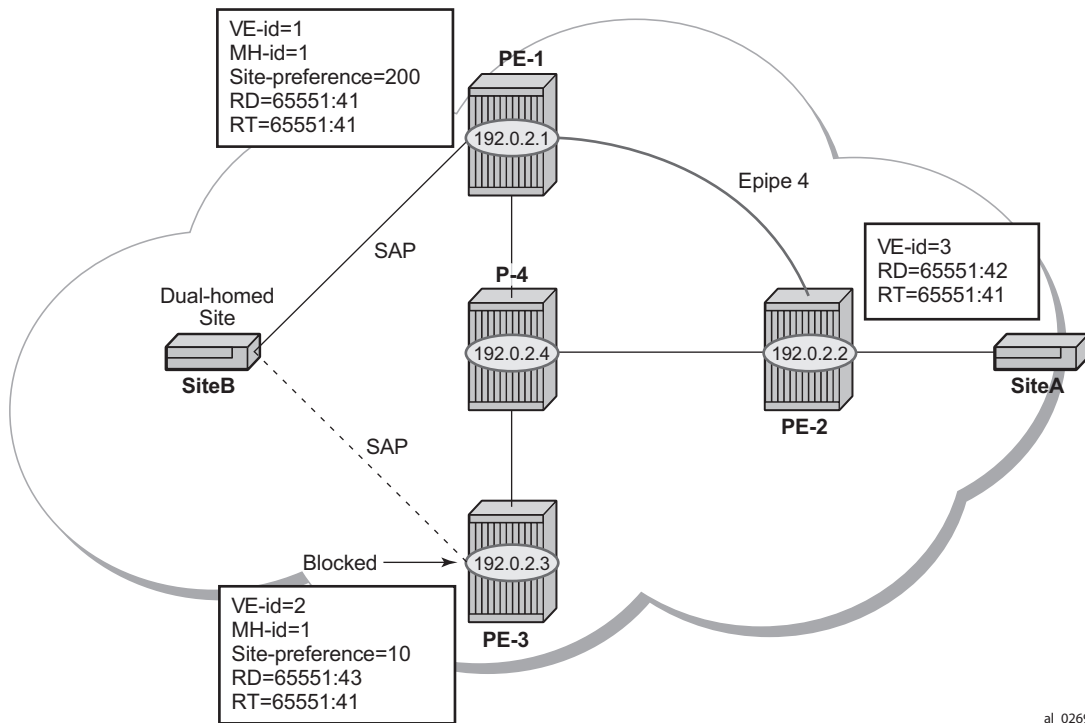


Figure 145: Dual Homed BGP VPWS with Active/Standby Pseudowire

BGP multi-homing (MH) is configured for the dual-homed site B using a site-id=1. The site-preference on PE-1 is set to 200 and to 10 on PE-3; this ensures that PE-1 will be the site's designated forwarder for the MH site. The active pseudowire from PE-2 will be created to PE-1

Dual Homed BGP VPWS with Active/Standby Pseudowire

with the standby pseudowire being created to PE-3. If PE-1 fails, or the multi-homing site fails over to PE-3, then the pseudowire from PE-2 to PE-3 will become active (used as the data path between site A and B).

PE-1 configuration:

```
*A:PE-1# configure service
      pw-template 3 create
      exit
      epipe 4 customer 1 create
        bgp
          route-distinguisher 65551:41
          route-target export target:65551:41 import target:65551:41
          pw-template-binding 3
          exit
        exit
      bgp-vpws
        ve-name "PE-1"
        ve-id 1
        exit
        remote-ve-name "PE-2"
        ve-id 2
        exit
        no shutdown
      exit
      site "SITEB" create
        site-id 1
        sap 1/1/4:44
        site-preference 200
        no shutdown
      exit
      sap 1/1/4:44 create
      exit
      no shutdown
    exit
```


PE-3 configuration:

Note that the local VE-ID is 3 (different from previous example).

```
*A:PE-3# configure service
pw-template 3 create
exit
epipe 4 customer 1 create
  bgp
    route-distinguisher 65551:43
    route-target export target:65551:41 import target:65551:41
    pw-template-binding 3
  exit
exit
bgp-vpws
  ve-name "PE-3"
  ve-id 3
  exit
  remote-ve-name "PE-2"
  ve-id 2
  exit
  no shutdown
exit
site "SITEB" create
  site-id 1
  sap 1/1/4:44
  site-preference 10
  no shutdown
exit
sap 1/1/4:44 create
exit
no shutdown
exit
```

PE-2 configuration:

Note that there are two remote VE names configured, PE-1 and PE-3 (this is the maximum number allowed).

```
*A:PE-2# configure service
pw-template 3 create
exit
epipe 4 customer 1 create
  bgp
    route-distinguisher 65551:42
    route-target export target:65551:41 import target:65551:41
    pw-template-binding 3
  exit
exit
bgp-vpws
  ve-name "PE-2"
  ve-id 2
  exit
  remote-ve-name "PE-1"
```

Dual Homed BGP VPWS with Active/Standby Pseudowire

```
        ve-id 1
    exit
    remote-ve-name "PE-3"
        ve-id 3
    exit
    no shutdown
exit
sap 1/1/4:44 create
exit
no shutdown
```

Compared with the single pseudowire solution, both pseudowires are signaled and up on all PEs. The pseudowire with the higher preference is forwarding traffic (to PE-1), while the Tx Status on the other one is set to inactive.

Verified on PE-2:

```
*A:PE-2# show service l2-route-table bgp-vpws detail
=====
Services: L2 Bgp-Vpws Route Information - Summary
=====

---snip---

Svc Id      : 4
VeId        : 1
PW Temp Id  : 3
RD          : *65551:41
Next Hop    : 192.0.2.1
State (D-Bit) : up(0)
Path MTU    : 1514
Control Word : 0
Seq Delivery : 0
Status      : active
Tx Status   : active
CSV         : 0
Preference  : 200
Sdp Bind Id : 17407:4294967291

Svc Id      : 4
VeId        : 3
PW Temp Id  : 3
RD          : *65551:43
Next Hop    : 192.0.2.3
State (D-Bit) : up(0)
Path MTU    : 1514
Control Word : 0
Seq Delivery : 0
Status      : active
Tx Status   : inactive
CSV         : 0
Preference  : 10
Sdp Bind Id : 17406:4294967290
=====
*A:PE-2#
```

The choice of pseudowire to be used to transmit traffic from PE-2 to PE-1 can also be seen in the endpoint created in the BGP VPWS service. Endpoints are automatically created for the pseudowires within a BGP VPWS service regardless of whether active/standby pseudowires are used; these endpoints are created with a system generated name that ends with the BGP VPWS service id.

```
*A:PE-2# show service id 4 endpoint
=====
Service 4 endpoints
=====
Endpoint name           : _tmnx_BgpVpws-4
Description              : Automatically created BGP-VPWS endpoint
Creation Origin          : bgpVpws
Revert time              : 0
Act Hold Delay           : 0
Standby Signaling Master : false
Standby Signaling Slave  : false
Tx Active (SDP)          : 17407:4294967291
Tx Active Up Time        : 0d 00:00:48
Revert Time Count Down   : N/A
Tx Active Change Count   : 3
Last Tx Active Change    : 07/06/2015 13:46:52
-----
Members
-----
Spoke-sdp: 17406:4294967290 Prec:4          Oper Status: Up
Spoke-sdp: 17407:4294967291 Prec:4          Oper Status: Up
=====
=====
*A:PE-2#
```

Note that the following command has no effect on an automatically created VPWS endpoint.

```
tools perform service id <service-id> endpoint <endpoint-name> force-switchover
```

Conclusion

BGP VPWS allows the delivery of Layer 2 virtual private wire services to customers where BGP is commonly used. This chapter shows the configuration of single and dual-homed BGP VPWS services together with the associated show output, which can be used to verify and troubleshoot them.

EVPN for MPLS Tunnels

In This Chapter

This section provides information about EVPN for MPLS tunnels.

Topics in this section include:

- [Applicability on page 938](#)
- [Overview on page 939](#)
- [Configuration on page 942](#)
- [Conclusion on page 990](#)

Applicability

This chapter is applicable to 7750 SR-7/12, 7750 SR-a4/8, 7750 SR-12E, 7450 ESS-7/12, XRS-20/16c, and 7750-c4/12. Ethernet Virtual Private Networks (EVPN) for Multi-Protocol Label Switching (MPLS) tunnels requires IOM3-XP/IMM or higher-based line cards and chassis-mode D.

The configuration was tested on SR OS release 13.0.R6. A prerequisite is to read the [EVPN for VXLAN Tunnels \(Layer 2\) on page 1033](#) chapter.

Overview

EVPN-MPLS is standardized in RFC 7432, *BGP MPLS-Based Ethernet VPN*, as a Layer 2 VPN technology that can supplement VPLS for ELAN services. Besides the optimizations introduced by EVPN, a significant number of service providers offering ELAN services today are requesting EVPN for their multi-homing capabilities. EVPN supports all-active multi-homing (per-flow load-balancing multi-homing) as well as single-active multi-homing (per-service load-balancing multi-homing). In addition to those superior multi-homing capabilities, EVPN also provides a number of significant benefits, such as:

- IP-VPN-like operation and control for ELAN services.
- Reduction and (in some cases) suppression of the BUM (broadcast, unknown unicast, and multicast) traffic in the network.
- Simple provision and management.
- New set of tools to control the distribution of MAC addresses and ARP entries in the network.

While the VPN for VXLAN tunnels (Layer-2) chapter focuses on the use of EVPN as a control plane for VXLAN tunnels, this chapter provides configuration guidelines for EVPN when used for MPLS tunnels. Similar to EVPN-VXLAN services, VPLS services with EVPN for MPLS tunnels are referred to as EVPN-MPLS services.

As a reference, the EVPN route types and NLRIs (Network Layer Reachability Information messages) used by the EVPN family in RFC 7432 are shown in [Figure 146](#).

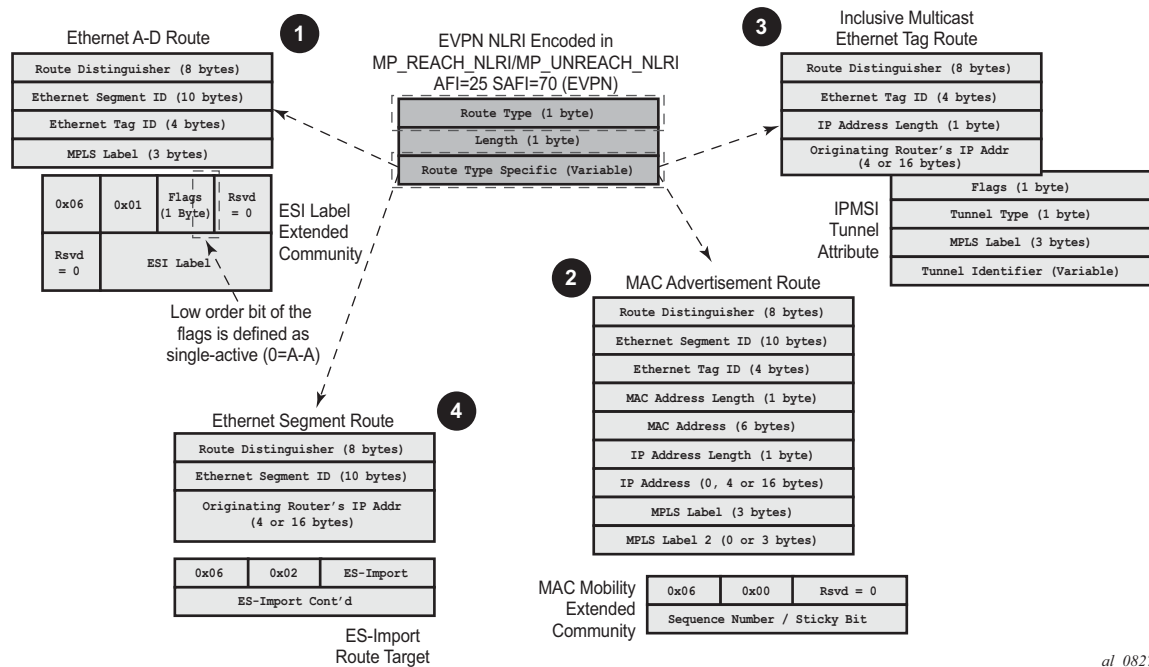


Figure 146: EVPN Route Types and NLRIs

When no EVPN multi-homing is used in the network, only the base routes are used. Routes type 2 and 3 are considered the base and mandatory routes:

- **Route type 2 - MAC/IP route:** This route advertises MAC addresses to be installed in the remote FDBs, or MAC/IP address pairs to be installed in the remote proxy-ARP/ND tables.
- **Route type 3 - Inclusive multicast route:** This route advertises the multicast tree that the advertising PE intends to use for sending BUM (Broadcast, Unknown, Multicast) traffic for an EVI (EVPN Instance). In SR OS 13.0.R6, only ingress replication is supported. The ingress replication information, as well as the downstream MPLS label (for remote PEs to send BUM traffic to the advertising PE) are encoded in the PTA (Provider Multicast Service Interface Tunnel Attribute).

When EVPN multi-homing is used in an EVI, routes type 1 and 4 are used (where type 1 has two different purposes):

- **Route type 1 - Auto-discovery per Ethernet segment (AD per ES) route:** This route is advertised per ES from the PE, carries the Ethernet segment identifier (ESI) label (used for split-horizon) in multi-homing mode, and can affect procedures such as the DF election, as well as the aliasing/backup path/mass withdrawal on remote PEs.

- Route-type 1 - Auto-discovery per EVPN instance (AD per-EVI) (auto-discovery per EVPN instance) route: This route allows the remote PEs to provide aliasing and a backup path to the PEs part of the ES.
- Route type 4 - Ethernet segment (ES) route: This route advertises a local configured ES. The exchange of this route can discover remote PEs that are part of the same ES and the Designated Forwarder (DF) election algorithm among them.

Note: The AD per-EVI, MAC/IP, and inclusive multicast routes are considered service-level BGP-EVPN routes. Their RT/RD (Route-Target/Route-Distinguisher) are taken from the VPLS configuration.

The AD per-ES and the ES routes are considered base-level BGP-EVPN routes. However, their RT/RD are taken differently:

- The ES route RD is taken from the **service>system>bgp-evpn** configuration. The ES route RT is auto-derived from the ethernet-segment.
- The AD per-ES route RD and RT are taken from the **config>service>vpls** configuration in the current release.

Configuration

This section describes the configuration of EVPN-MPLS for Layer 2 services on the 7x50, as well as the available troubleshooting and show commands, and EVPN multi-homing.

Figure 147 shows the topology used throughout this chapter. The network consists of a core with four EVPN PE (PE-2, PE-3, PE-4, and PE-5) and two MTU devices that are dual-homed to the EVPN network. For MTU-1, all-active multi-homing is used, whereas MTU-6 is connected via single-active multi-homing to the EVPN network. Three CEs are connected to VPLS 1 in MTU-1, PE-3, and MTU-6 in order to test the connectivity.

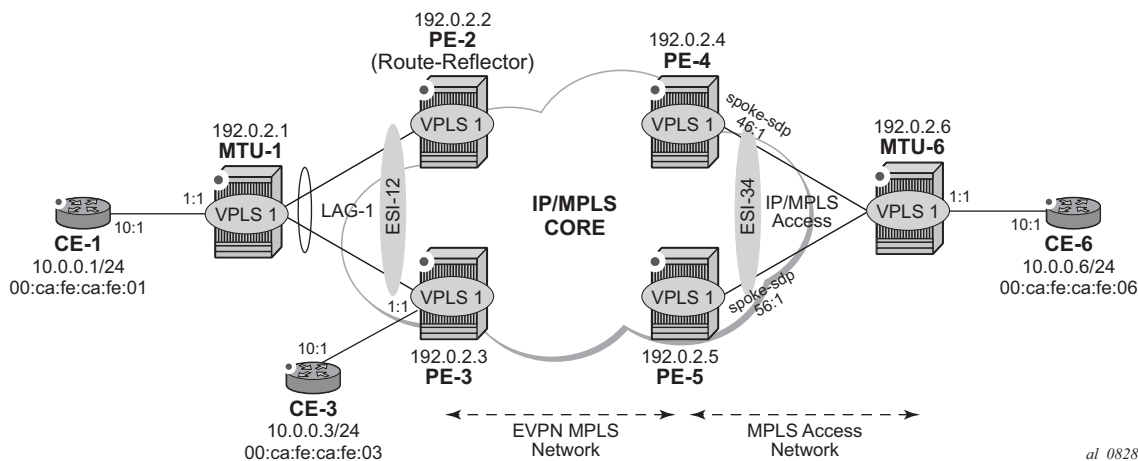


Figure 147: EVPN-MPLS for VPLS Services

As part of the network infrastructure configuration, the following settings and protocols must be added to the configuration before starting with the EVPN-specific configuration for the services:

- The ports interconnecting the four PEs in the core are configured as network ports (or hybrid) and will have router network interfaces defined in them. The ports on PE-2/PE-3 connected to MTU-1 can be access or hybrid ports, whereas the ports on PE-4/PE-5 connected to MTU-6 can be network or hybrid ports. Note: In case of hybrid ports, no LACP can be configured.
- The four PEs in the core (as well as MTU-6 in the access MPLS network) are running ISIS and establishing point-to-point adjacencies for the exchange of the system IP addresses.

- LDP is used as the MPLS protocol to signal transport tunnel labels among PE-2, PE-3, PE-4, PE-5, and MTU-6. There is no LDP running between MTU-1 and the rest of the network, that is, MTU-1 is a pure Ethernet aggregation device.
- EVPN uses MP-BGP for exchanging reachability at service level. Therefore, BGP peering sessions must be established among the core PEs for the EVPN family. Although typically a separate router is used, in this chapter, PE-2 is used as BGP RR (route reflector) for EVPN routes. For example, the following output shows the configuration of BGP in the RR and one of the BGP clients. The relevant commands for EVPN are shown in bold.

```
*A:PE-2>config>router>bgp# info # this is the RR BGP configuration
```

```
-----
vpn-apply-import
vpn-apply-export
min-route-advertisement 1
enable-peer-tracking
rapid-withdrawal
split-horizon
rapid-update evpn
group "internal"
    family evpn
        type internal
        cluster 1.1.1.1
        neighbor 192.0.2.3
        exit
        neighbor 192.0.2.4
        exit
        neighbor 192.0.2.5
        exit
    exit
no shutdown
-----
```

```
A:PE-3>config>router>bgp# info # this is the BGP configuration for a client
```

```
-----
vpn-apply-import
vpn-apply-export
min-route-advertisement 1
enable-peer-tracking
rapid-withdrawal
split-horizon
rapid-update evpn
group "internal"
    family evpn
        type internal
        neighbor 192.0.2.2
        exit
    exit
no shutdown
-----
```

Note: The **def-recv-evpn-encap** command is not used in the above configuration because the default mpls configuration is sufficient to have a correct interpretation of the received EVPN encapsulations.

EVPN-MPLS Configuration without Multi-Homing

```
A:PE-3>config>router>bgp>group>neighbor# info detail | match def-recv
def-recv-evpn-encap mpls
```

```
A:PE-3>config>router>bgp>group>neighbor# def-recv-evpn-encap ?
- def-recv-evpn-encap <encap-type>
```

```
<encap-type>          : mpls|vxlan
```

EVPN routes type 1 (auto-discovery per-EVI route), type 2 (MAC/IP route), type 3 (inclusive multicast route), and type 5 (IP-prefix route) are always sent with the RFC 5512, *The BGP Encapsulation Subsequent Address Family Identifier (SAFI) and the BGP Tunnel Encapsulation Attribute*, BGP encapsulation extended community that indicates the associated encapsulation of the route. Because the use of this extended community is not mandatory in RFC 7432, the def-recv-evpn-encap command indicates to the system what encapsulation is associated with routes received without any encapsulation. By default, this command is set to mpls. When interoperating with third-party EVPN vendors in mixed MPLS and EVPN-VXLAN networks, this command should be revised accordingly.

EVPN-MPLS Configuration without Multi-Homing

After the base infrastructure (interfaces, IGP, LDP, BGP protocols) is configured, the service and EVPN can be enabled. When no multi-homing is used, the EVPN-MPLS configuration in a VPLS service looks similar to the configuration of EVPN-VXLAN for Layer 2, except for the commands related to the MPLS data plane. The following output shows the VPLS-1 configuration in PE-3 as an example:

```
A:PE-3>config>service>vpls# info
-----
      bgp
      exit
      bgp-evpn
      evi 1
      vxlan
      shutdown
      exit
      mpls
      ingress-replication-bum-label
      ecmp 2
      auto-bind-tunnel
      resolution any
      exit
      no shutdown
      exit
    exit
  exit
  stp
  shutdown
  exit
  sap 1/1/1:1 create
  exit
  sap lag-1:1 create
```

```
exit
no shutdown
-----
```

Where the following commands are relevant for a basic EVPN configuration:

- **bgp** enables the context for the BGP configuration relevant to the service. If a manual (non-auto-derived) RD/RT, as well as import/export policies, are needed for the service, the commands under the **bgp** context must be configured. When **bgp-evpn** is enabled in a VPLS instance, other families are supported within the same service (**bgp-ad** and **bgp-mh**, not **bgp-vpls**). This **bgp** context configures the common BGP parameters for all the BGP families in the service. The **pw-template-binding** command is ignored for **bgp-evpn**. Even if the general BGP parameters for the service are auto-derived (as in this example), the **bgp** context must be enabled.

```
A:PE-3>config>service>vpls# bgp ?
- bgp
- no bgp

[no] pw-template-bi* + Configure pw-template bind policy
[no] route-distingu* - Configure route distinguisher
[no] route-target    - Configure route target
[no] vsi-export      - VSI export route policies
[no] vsi-import      - VSI import route policies
```

- **bgp-evpn evi <1..65535>** — The EVPN instance or **evi** is a 2-byte identifier used for the auto-derivation of the service RD, service RT, and for the service-carving algorithm when multi-homing is used. The **evi** can be used for both **bgp-evpn vxlan** and **bgp-evpn mpls** when the user needs to auto-derive the RD and RT for the service. The auto-derivation is always based on:

→ RD system-ip:evi

→ RT autonomous-system:evi

The configured and operating RD/RT values can be checked with the following show command (in this example, the **evi** value is 1):

```
A:PE-3# show service id 1 bgp
=====
BGP Information
=====
Vsi-Import      : None
Vsi-Export      : None
Route Dist      : None
Oper Route Dist : 192.0.2.3:1
Oper RD Type     : derivedEvi
Rte-Target Import : None
Oper RT Imp Origin : derivedEvi
Oper RT Exp Origin : derivedEvi
PW-Template Id   : None
Rte-Target Export : None
Oper RT Import    : 64500:1
Oper RT Export    : 64500:1
=====
```

Note: Although not required for a basic BGP-EVPN MPLS configuration, some other parameters may be used at the `bgp-evpn` context level, when EVPN-MPLS services are deployed. Some examples are listed here:

- **`bgp-evpn>cfm-mac-advertisement`** must be enabled when `eth-cfm` is used across an EVPN-MPLS service among different PEs. If a maintenance endpoint (MEP) or maintenance domain intermediate point (MIP) is configured in any of the SAP/SDP bindings in the VPLS and has to exchange `eth-cfm` packets with a remote MEP/MIP across the EVPN-MPLS core, this command must be enabled. In that way, the MEP/MIP MAC address can be advertised in EVPN (otherwise, the MEP/MIP MAC address would not be learned on remote EVPN-MPLS PEs and `eth-cfm` would not work correctly).
- **`bgp-evpn>mac-advertisement` and `bgp-evpn>mac-duplication`** — See the [EVPN for VXLAN Tunnels \(Layer 2\) on page 1033](#) chapter for a description of these two commands.

Note: **`bgp-evpn>vxlan`** must be **shutdown** in release 13.0.R6 so that **`bgp-evpn mpls`** can be enabled.

After the relevant **VPLS** parameters, **BGP** and **`bgp-evpn`** attributes are added, the specific commands for **`bgp-evpn mpls`** can be configured:

```
A:PE-3>config>service>vpls>bgp-evpn>mpls# info
-----
      ingress-replication-bum-label
      ecmp 2
      auto-bind-tunnel
          resolution any
      exit
      no shutdown
-----
```

- **`ingress-replication-bum-label`** controls whether the system will advertise different service labels for unicast and BUM traffic. If no EVPN multi-homing is configured in the network, this command can be disabled (**no `ingress-replication-bum-label`**) and the same MPLS label will be advertised for the unicast and BUM traffic for the VPLS instance. If EVPN multi-homing is configured in the PE, this command is strongly recommended to avoid potential transient issues. See the EVPN-MPLS multi-homing section.
- **`ecmp`** controls the number of remote PEs to which the local PE can load balance the unicast traffic. See the EVPN multi-homing section.
- **`auto-bind-tunnel`** controls the resolution of EVPN destinations to MPLS transport tunnels. This command is also in VPRN services and works in the same way.
 - If the **`auto-bind-tunnel resolution any`** is configured, as in the example, EVPN destinations in the service are resolved based on the best tunnel in the tunnel table manager (TTM). For instance, the following command shows the existing EVPN destinations for VPLS 1 in PE-3. The EVPN-MPLS destination (Termination

endpoint (TEP) 192.0.2.2, label 262140) is resolved to an LDP transport tunnel because the (best) LDP tunnel to 192.0.2.2 shown in the **show router tunnel-table** is LDP. If there was more than one tunnel type in the TTM to 192.0.2.2, the system would pick the lowest **Pref** (preference) tunnel.

```
A:PE-3# show service id 1 evpn-mpls
=====
BGP EVPN-MPLS Dest
=====
TEP Address      Egr Label      Num. MACs      Mcast          Last Change
      Transport
-----
192.0.2.2        262140         0              Yes            09/11/2015 19:09:05
                  ldp
192.0.2.4        262140         0              Yes            09/11/2015 19:09:05
                  ldp
192.0.2.5        262140         0              Yes            09/11/2015 19:09:05
                  ldp
-----
Number of entries : 3
-----
---snipped--->

*A:PE-3# show router tunnel-table
=====
IPv4 Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId  Pref      Nexthop      Metric
-----
192.0.2.2/32      ldp        MPLS  65540        9          192.168.23.1  10
192.0.2.4/32      ldp        MPLS  65537        9          192.168.34.2  10
192.0.2.5/32      ldp        MPLS  65539        9          192.168.35.2  10
192.0.2.6/32      ldp        MPLS  65538        9          192.168.34.2  20
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
```

- If resolution is set to any, the following tunnel types are selected in order of preference: RSVP, LDP, Segment Routing, and BGP. The user can configure the preference of the segment-routing tunnel type in the TTM for a specific IGP instance.
- If one or more explicit tunnel types are specified using the resolution-filter option, then only these tunnel types will be selected again following the TTM preference.
- The user must set the resolution to filter to activate the list of tunnel-types configured under resolution-filter.

Although not shown in the **bgp-evpn mpls** basic configuration for PE-3, there are other parameters that can be modified:

```
*A:PE-3>config>service>vpls>bgp-evpn>mpls#
  auto-bind-tunn* + Configure BGP EVPN mpls auto-bind-tunnel
[no] control-word - Enable/disable setting the CW
```

EVPN-MPLS Configuration without Multi-Homing

```
ecmp                - Configure maximum ECMP routes information
[no] force-vlan-vc-* - Forces vlan-vc-type forwarding in the data-path
[no] ingress-replic* - Use the same label as the one advertised for unicast traffic
[no] shutdown        - Administratively Enable/Disable BGP-EVPN mpls
[no] split-horizon-* - Configure a split-horizon-group
```

- **control-word** enables/disables the insertion of the control-word in the data path. The control-word is disabled by default and is not signaled in EVPN (based on RFC 7432) and has to be consistently configured in all the PEs in the network. The use of the **control-word** prevents packet-reordering from happening in P routers that misinterpret the first nibble of the payload in the packets they receive. Note: In some third-party EVPN vendors, the control-word is enabled by default, so it is recommended to enable it when interoperating with other vendors.
- **force-vlan-vc-forwarding** allows the system to preserve the vlan-id and pbits of the service-delimiting qtag in a new tag added in the customer frame before sending it to the EVPN core. This command may be used with the **sap ingress vlan-translation** command: the configured translated vlan-id will be sent to the EVPN binds, as opposed to the service-delimiting tag vlan-id. If the ingress SAP/SDP-binding is null encapsulated, the output vlan-id and pbits will be zero.
- **shutdown** enables/disables the use of MPLS for EVPN. When **mpls no shutdown** is issued, a BGP route-refresh message is sent for the EVPN family.
- **split-horizon-group** <group-name> configures an explicit split-horizon-group (SHG) for all the EVPN destinations that can be shared with other SAP/SDP-bindings. See the VPLS to EVPN-MPLS migration and integration section.

After **bgp-evpn mpls** is configured in the service, and **no shutdown**, an inclusive multicast route is sent to the RR. The remote PEs receiving and importing that route will create an EVPN destination to the sending PE. An EVPN destination is identified by a TEP and MPLS label. Use the following show commands to view the service and the EVPN destinations created:

```
show service evpn-mpls
show service id 1 evpn-mpls
show service id 1 bgp-evpn
```

An example of the output is shown below for PE-2 when there is no traffic in the network. Therefore, only inclusive multicast routes have been exchanged among the four PEs.

```
*A:PE-2# show service evpn-mpls
=====
EVPN MPLS Tunnel Endpoints
=====
EvpnMplsTEP Address EVPN-MPLS Dest      ES Dest      ES BMac Dest
-----
192.0.2.3      1          0          0
192.0.2.4      1          0          0
192.0.2.5      1          0          0
-----
Number of EvpnMpls Tunnel Endpoints: 3
=====
```



```

*A:PE-2# show service id 1 evpn-mpls
=====
BGP EVPN-MPLS Dest
=====
TEP Address      Egr Label      Num. MACs      Mcast          Last Change
                  Transport
-----
192.0.2.3        262140         0              Yes            09/11/2015 19:09:05
                  ldp
192.0.2.4        262140         0              Yes            09/11/2015 19:09:05
                  ldp
192.0.2.5        262140         0              Yes            09/11/2015 19:09:05
                  ldp
-----
Number of entries : 3
-----
=====
BGP EVPN-MPLS Ethernet Segment Dest
=====
Eth SegId              Num. Macs          Last Change
-----
No Matching Entries
=====
BGP EVPN-MPLS ES BMAC Dest
=====
ES BMAC Addr          Last Change
-----
No Matching Entries
=====
*A:PE-2#
*A:PE-2# show service id 1 bgp-evpn
=====
BGP EVPN Table
=====
MAC Advertisement      : Enabled          Unknown MAC Route      : Disabled
CFM MAC Advertise      : Disabled
VXLAN Admin Status     : Disabled          Creation Origin        : manual
MAC Dup Detn Moves     : 5                MAC Dup Detn Window    : 3
MAC Dup Detn Retry     : 9                Number of Dup MACs     : 0
IP Route Advertisement : Disabled
EVI                     : 1
-----
Detected Duplicate MAC Addresses      Time Detected
-----
=====
BGP EVPN MPLS Information
=====
Admin Status           : Enabled
Force Vlan Fwding      : Disabled          Control Word           : Disabled
Split Horizon Group: (Not Specified)
Ingress Rep BUM Lbl: Enabled          Max Ecmp Routes        : 2
Ingress Ucast Lbl  : 262141          Ingress Mcast Lbl      : 262140
=====

```

EVPN-MPLS Configuration without Multi-Homing

```
=====
BGP EVPN MPLS Auto Bind Tunnel Information
=====
Resolution          : any
Filter Tunnel Types: (Not Specified)
=====
```

When traffic is generated, the PEs will start learning MAC addresses and advertising them in BGP so that the remote PEs learn those MAC addresses against EVPN destinations. For instance, when CE-3 sends traffic, PE-3 learns its MAC address and advertises it. The remote PEs (for instance, PE-2) will learn the MAC address and associate it with their EVPN destination to PE-3 (192.0.2.3:262141 in this example):

```
*A:PE-2# show service id 1 fdb detail
=====
Forwarding Database, Service 1
=====
ServId      MAC                Source-Identifier      Type      Last Change
-----
1           00:ca:fe:ca:fe:03 eMpls:             Evpn      09/12/15 01:50:54
                192.0.2.3:262141
-----
No. of MAC Entries: 1
-----
Legend:  L=Learned O=Oam P=Protected-MAC C=Conditional S=Static
=====
```

If the ingress-replication-bum-label is enabled in the PEs, the advertisement of MAC addresses will create new EVPN destinations, because the label is different from the one previously sent by the inclusive multicast route that created an EVPN destination. In the example above, when PE-3 advertises the CE-3 MAC address, PE-2 will create a new binding (see below in bold) that shows one MAC address that is not Mcast (multicast) capable:

```
*A:PE-2# show service id 1 evpn-mpls
=====
BGP EVPN-MPLS Dest
=====
TEP Address      Egr Label      Num. MACs  Mcast      Last Change
-----
192.0.2.3        262140         0          Yes        09/12/2015 01:50:54
                  ldp
192.0.2.3        262141         1          No         09/12/2015 02:03:23
                  ldp
192.0.2.4        262140         0          Yes        09/11/2015 19:09:05
                  ldp
192.0.2.5        262140         0          Yes        09/11/2015 19:09:05
                  ldp
-----
Number of entries : 4
=====
```

When an EVPN-MPLS destination or MAC address is not created/installed correctly, the user may check out the BGP-EVPN routes received and the routes kept in the RIB. The routes that the PE receives are shown when debug router bgp update is enabled. These routes are shown even before any BGP processing is carried out.

```
*A:PE-2#
4 2015/09/12 21:41:16.27 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 88
  Flag: 0x90 Type: 14 Len: 44 Multiprotocol Reachable NLRI:
    Address Family EVPN
    NextHop len 4 NextHop 192.0.2.3
    Type: EVPN-MAC Len: 33 RD: 192.0.2.3:1 ESI: ESI-0, tag: 0, mac len: 48 m
ac: 00:ca:fe:ca:fe:03, IP len: 0, IP: NULL, label1: 4194256
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 16 Extended Community:
    target:64500:1
    bgp-tunnel-encap:MPLS
"

*A:PE-2# show router bgp routes evpn mac
=====
BGP Router ID:192.0.2.2      AS:64500      Local AS:64500
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked, x - stale, > - best, b - backup, p - purge
Origin codes  : i - IGP, e - EGP, ? - incomplete

=====
BGP EVPN MAC Routes
=====
Flag  Route Dist.      MacAddr      ESI
      Tag              Mac Mobility  Ip Address
                        NextHop
                        Label1
-----
u*>i  192.0.2.3:1      00:ca:fe:ca:fe:03 ESI-0
      0                Seq:0          N/A
                        192.0.2.3
                        LABEL 262141

u*>i  192.0.2.5:1      00:ca:fe:ca:fe:06 01:00:00:00:00:34:00:00:00:01
      0                Seq:0          N/A
                        192.0.2.5
                        LABEL 262141

-----
Routes : 2
=====
```

If the route is successfully imported, it can be shown in the RIB (show router bgp routes commands). The route shown in the debug and the same route in a show command do not necessarily have the same label value. The reason for this expected mismatch is that the debug command shows the complete 24-bit field value because the route is shown before BGP can decipher whether the label value is an MPLS label (high-order 20-bits of the Label field) or a VNI (all 24 bits of the Label field for VXLAN).

VPLS to EVPN-MPLS Integration

The 7x50 SR OS EVPN implementation supports draft-ietf-bess-evpn-vpls-seamless-integ so that EVPN-MPLS and VPLS can be integrated into the same network and within the same service.

The following behavior enables the integration of EVPN and sdp-bindings in the same VPLS network:

- Systems with EVPN endpoints and SDP-bindings to the same far-end bring down the sdp-bindings.
 - The 7x50 will allow the establishment of an EVPN destination and an sdp-binding to the same far-end but the sdp-binding will be kept operationally down. Only the EVPN endpoint will be operationally up. This is true for spoke-sdps (manual and BGP-AD) and mesh-sdps. It is also true between VXLAN and SDP-bindings.
 - If there is an EVPN endpoint to a specified far-end and a spoke-sdp establishment is attempted, the spoke-sdp will be set up but kept down with an operational flag indicating that there is an EVPN route to the same far-end.
 - If there is a spoke-sdp and a valid/used EVPN route arrives, the EVPN endpoint will be set up and the spoke-sdp will be brought down with an operational flag indicating that there is an EVPN route to the same far-end.
 - In the case of an sdp-binding and EVPN endpoint to different far-end IPs on the same remote PE, both links will be up. This can happen if the sdp-binding is terminated in an IPv6 address or IPv4 address different from the system address where the EVPN endpoint is terminated.

The following example illustrates the description above.

```
*A:PE-2>config>service>vpls# info
-----
      bgp
      exit
      bgp-evpn
        evi 1
        vxlan
          shutdown
        exit
      mpls
        ingress-replication-bum-label
        ecmp 2
```

```

        auto-bind-tunnel
            resolution any
        exit
        no shutdown
    exit
exit
stp
shutdown
exit
sap lag-1:1 create
exit
spoke-sdp 24:1 create
    no shutdown
exit
no shutdown
-----

```

```
*A:PE-4>config>service>vpls# info
```

```

-----
    bgp
    exit
    bgp-evpn
        evi 1
        vxlan
            shutdown
        exit
        mpls
            ingress-replication-bum-label
            ecmp 2
            auto-bind-tunnel
                resolution any
            exit
            no shutdown
        exit
    exit
    stp
        shutdown
    exit
    spoke-sdp 42:1 create
        no shutdown
    exit
    spoke-sdp 46:1 create
        no shutdown
    exit
    no shutdown
-----

```

```
*A:PE-2# show service id 1 base
```

```
---snipped---
```

```
-----
Service Access & Destination Points
-----
```

Identifier	Type	AdmMTU	OprMTU	Adm	Opr
sap:lag-1:1	q-tag	1518	1518	Up	Up
sdp:24:1 S(192.0.2.4)	Spok	0	8974	Up	Down

```
=====
```

```
*A:PE-2# show service id 1 sdp 24 detail | match Flag
Flags                : EvpnRouteConflict
```

- The user can add spoke-sdps and all the EVPN-MPLS endpoints in the same SHG.
 - A CLI command exists under the **bgp-evpn>mpls** context so that the EVPN-MPLS endpoints can be added to an SHG.
 - The **bgp-evpn mpls split-horizon-group** must reference a user-configured split-horizon-group. User-configured split-horizon-groups can be configured within the service context.
 - The same group-name can be associated with saps, spoke-sdps, pw-templates, pw-template-bindings, and EVPN-MPLS endpoints.
 - If the **split-horizon-group** command in **bgp-evpn>mpls** is not used, the default split-horizon-group (in which all the EVPN endpoints are) is still used, but it will not be possible to refer to it on saps/spoke-sdps.
- The system disables the advertisement of MAC addresses learned on spoke SDPs/SAPs that are part of an EVPN split-horizon-group.
 - When the SAPs or spoke SDPs (manual or BGP-AD-discovered) are configured within the same SHG as the EVPN endpoints, MAC addresses will still be learned on them, but will not be advertised in EVPN.
 - The preceding statement is also true if proxy-ARP/ND is enabled and an IP-->MAC address pair is learned on a sap/sdp-binding that belongs to the EVPN SHG.
 - The SAPs and/or spoke-SDPs added to an EVPN SHG should not be part of any EVPN multi-homed ES. If that happened, the PE would still advertise the AD per-EVI route for the SAP and/or spoke-SDP, attracting EVPN traffic that could not be forwarded to that SAP and/or sdp-binding.
 - Similar to the preceding statement, an SHG composed of SAPs/sdp-bindings used in a BGP-MH site should not be configured under **bgp-evpn>mpls>split-horizon-group**. This misconfiguration would prevent traffic being forwarded from the EVPN to the BGP-MH site, regardless of the DF/Non-DF state.

An example of a shared SHG configuration is shown below. Because the SAP and evpn-mpls are in the same SHG, no MAC addresses learned over SAP 1/1/1:2 will be advertised in EVPN (not even static macs).

```
*A:PE-2>config>service>vpls# info
-----
split-horizon-group "CORE" create
exit
bgp
exit
bgp-evpn
evi 1
vxlan
shutdown
exit
```

```

mpls
    split-horizon-group "CORE"
    ingress-replication-bum-label
    ecmp 2
    auto-bind-tunnel
        resolution any
    exit
    no shutdown
exit
exit
stp
    shutdown
exit
sap 1/1/1:2 split-horizon-group "CORE" create
exit
sap lag-1:1 create
exit
no shutdown

```

EVPN-MPLS Multi-Homing

SR OS supports EVPN multi-homing as per RFC 7432.

The EVPN multi-homing implementation is based on the concept of the ES. An ES is a logical structure that can be defined in one or more PEs and identifies the CE (or access network) multi-homed to the EVPN PEs. An ES is associated with a port, LAG, or SDP object, and is shared by all the services defined on those objects.

Each ES has a unique identifier called ESI (Ethernet Segment Identifier) that is 10 bytes and is manually configured in the current release. The ESI is advertised in the control plane to all the PEs in an EVPN network; therefore, it is very important to ensure that the 10-byte ESI value is unique throughout the entire network. Single-homed CEs are assumed to be connected to an ES with ESI = 0 (single-homed ESs are not explicitly configured).

The ES is part of the base BGP-EVPN configuration and is not applied to any EVPN-MPLS service, by default. An ES can be shared by multiple services; the association of a specific SAP or spoke-SDP to an ES is automatically made when the SAP is defined in the same LAG or port configured in the ES, or when the spoke-SDP is defined in the same SDP configured in the ES. The following sections show the configuration of:

- an all-active multi-homing ES with a LAG associated with it
- a single-active multi-homing ES linked to an SDP

All-Active Multi-Homing Concepts

EVPN all-active multi-homing is built around three concepts: DF election, split-horizon (with an ESI-label), and aliasing, as shown in [Figure 148](#), from left to right.

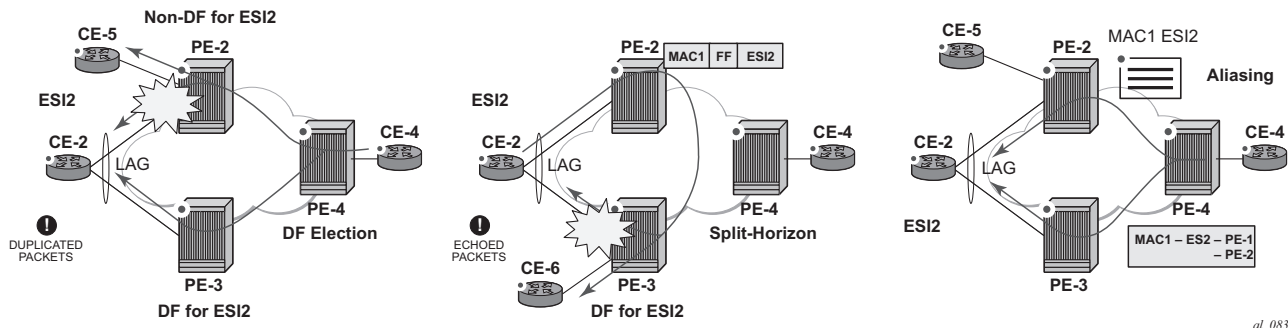


Figure 148: EVPN-MPLS All-Active Multi-Homing Concepts

- With DF election, when PE-4 sends BUM traffic to the remote ES (CE-2), only one PE segment sends the BUM packets to the ES (PE-3 is the DF in the example above, and is elected to send BUM packets to CE-2). Note that the non-DF, PE-2, removes the LAG SAP from the default multicast list (PE-2 does not bring CE-2 down because it still needs to send upstream/downstream unicast traffic). PE-2 and PE-3 elect a DF for each service, based on the ES routes and the service-carving algorithm.
- With split-horizon, the PE part of the ES (PE-3 in the example above) identifies the BUM packets coming from the PE for the remote (PE-2), but within the same ES (ESI-2), and filters the packets so that they are not sent back to the ES, creating duplication. When PE-2 (non-DF) sends BUM traffic to PE-3 (DF), it uses a special MPLS label in the data path that PE-3 previously advertised for ESI-2 in an AD per-ES route. When PE-3 does an ingress lookup, it recognizes the ESI-label and filters the traffic (PE-3 still sends the BUM traffic to other saps/sdp-bindings).
- With aliasing, remote PEs that are not part of the ES can load-balance unicast traffic to all the PEs that are part of the ES, irrespective of from which PE a destination MAC address was learned. PE-4 will create an EVPN destination to ESI-2 that will be resolved to the two next-hops: PE-2 and PE-3. Unicast load-balancing will happen as long as $ecmp > 1$ is enabled in PE-4.

Alcatel-Lucent recommends the use of **ingress-replication-bum-label** on the PEs that are part of an all-active ES. In an all-active multi-homing scenario, if a specified MAC address (for example, the CE-2 MAC address in the left-hand-side diagram), is not learned yet in a remote PE (for example, PE-4), but is known in the two PEs of the ES (for example, PE-2 and PE-3), the latter PEs might send duplicated packets to the CE.

This issue is solved by the use of **ingress-replication-bum-label** in PE-2 and PE-3. If configured, PE-2/PE-3 will know that the received packet is an unknown unicast packet; therefore, the Non-DF (PE-2) will not send the packet to CE-2 and there will not be duplication.

All-Active Multi-Homing Configuration

The all-active multi-homing configuration example is based on [Figure 147](#).

MTU-1 is connected to the EVPN network using all-active multi-homing. According to RFC 7432, MTU-1 will be able to send traffic to both PEs for VPLS-1. Regular LAG load-balancing is used in MTU-1. Remote PEs such as PE-4 or PE-5 will be able to load-balance the unicast traffic to PE-2 and PE-3. PE-2 and PE-3 will discover that both are part of ESI-12 (due to the exchange of ES routes) and will elect a DF for VPLS-1. The non-DF for VPLS-1, in this case PE-2, will remove lag-1:1 from the VPLS-1 default multicast list. Also, when PE-2 and PE-3 send BUM traffic to each other, they will insert an ESI-label so that they can identify that the source of the BUM packet is ESI-12.

The following output shows the configuration of ESI-12 in PE-2 and PE-3, as well as the LAG interfaces for all-active multi-homing (see [Figure 147](#)). The configuration of LAG-1 in MTU-1 is also shown below. Note that, per RFC 7432, only a CE/MTU with a LAG can be connected to an all-active multi-homing ES. No other configuration is permitted on the CE for all-active multi-homing.

```
*A:PE-2# configure lag 1
*A:PE-2>config>lag# info
-----
mode access
encap-type dot1q
port 1/1/2
lacp active administrative-key 1 system-id 00:00:00:00:69:72
no shutdown
-----

A:PE-2>config>service>system>bgp-evpn# info
-----
ethernet-segment "ESI-12" create
esi 01:00:00:00:00:12:00:00:00:01
es-activation-timer 3
service-carving
mode auto
exit
multi-homing all-active
lag 1
no shutdown
exit
-----

A:PE-3# configure lag 1
A:PE-3>config>lag# info
-----
mode access
```

```

encap-type dot1q
port 1/1/3
lacp active administrative-key 1 system-id 00:00:00:00:69:72
no shutdown
-----

A:PE-3>config>service>system>bgp-evpn# info
-----
    ethernet-segment "ESI-12" create
        esi 01:00:00:00:00:12:00:00:00:01
        es-activation-timer 3
        service-carving
            mode auto
        exit
        multi-homing all-active
        lag 1
        no shutdown
    exit
-----

```

When configuring an ES, the following must be considered:

- Any EVPN parameter that is not specific to any particular VPLS service, and is common to all the EVIs, is configured in a base BGP-EVPN instance located at **config>service>system>bgp-evpn**. In this base instance, the following attributes may be configured:
 - **ethernet-segments**
 - the base BGP-EVPN instance **route-distinguisher** that will be used for the ES routes. If this **route-distinguisher** is not configured, by default a type-1 RD will be derived as system-ip:0, as shown in the command help:

```

*A:PE-2>config>service>system>bgp-evpn# route-distinguisher
- no route-distinguisher
- route-distinguisher <rd>

<rd>
: <ip-addr:comm-val>
ip-addr      - a.b.c.d
comm-val     - [0..65535]
default:     system-ip:0

```

- The ES must be configured with a name and can contain the following parameters when configured for all-active multi-homing:
 - **esi** — 10-byte identifier that represents the ES in the BGP control plane. The same esi must be configured in all the PEs connected to the same CE/MTU (using a unique value that cannot be associated with any other CE/MTU/access network). Note that RFC 7432 defines five different types of **esi**. In SR OS, the **type** byte, as well as the other 9 bytes can be arbitrarily configured.
 - **multi-homing all-active** — This command indicates that the ES is in all-active mode.
 - **lag <lag-id>** — The LAG connected to the CE/MTU must be added to the ES. In this example, lag-1 is added to ESI-12, on both PE-2 and PE-3. Although a different lag-id may have been assigned to the same ES on PE-2 and PE-3, PE-2 and PE-3 must have the same configuration on the ES LAG; that is, encap-type. Also, if LACP is added (it is not mandatory), both PEs must have the same admin-key, system-id, and system-priority. MTU-1 will see PE-2 and PE-3 as a single LAG peer. For all-active multi-homing, only the **lag** option is accepted by the system; **port** or **sdp** are not accepted.
 - **[no] shutdown** — This command controls the administrative state of the ES.
- The above parameters are the minimum necessary so that the ES can be activated. In addition to those parameters, there are a few more that the user can configure if requiring values different from the default ones:
 - **es-activation-timer [0..100]** can be configured at **redundancy>bgp-evpn-multi-homing>es-activation-timer** or at **service>system>bgp-evpn>eth-seg>es-activation-timer level** (the most specific value is used).
The **es-activation-timer** operation is explained below:
 - Upon reception of an ES, AD per-ES/EVI route update/withdrawal for a local ESI, the DF-candidate list of IPs is updated and the DF election algorithm is run without waiting for any timer.
 - If the result of the DF election requires the PE to be promoted from non-DF to DF, the **es-activation-timer** will start, and only after its expiration will the PE add the SAP to the default-multicast list. Note: Transitions from non-DF to non-DF, or from DF to non-DF, are immediate and do not wait for any timer.
 - This use of an **es-activation-timer** value minimizes the risks of loops and packet duplication due to **transient** multiple DFs.
 - The same **es-activation-timer** must be configured in all the PEs that are part of the same ESI. The user must configure either a long timer to minimize the risks of loops/duplication, or **es-activation-timer=0** to speed up the convergence for NDF → DF transitions. The default value is 3 seconds.
 - **service-carving** — As defined in RFC 7432, service-carving controls the distribution of DF/non-DF roles across the different services defined in an ES.

```
*A:PE-2>config>service>system>bgp-evpn>eth-seg>service-carving# mode
- mode {manual|auto}

<manual|auto>          : auto|manual|off
```

```
*A:PE-2>config>service>system>bgp-evpn>eth-seg>service-carving# manual
- manual

[no] evi          - Configure EVI range
[no] isid         - Configure ISID range
```

→ As shown above, **service-carving** has three different modes:

- **service-carving mode auto** (default) — The DF election algorithm will run the function $[V(\text{evi}) \bmod N(\text{peers}) = i(\text{ordinal})]$ to know who the DF for a specified service and ESI is. In this example, ESI-12 is configured with mode **auto**; therefore, for VPLS-1 (with EVI-1), PE-3 will be elected as DF because $\text{evi}(1) \bmod (2)\text{peers} = 1$, and the ordinal 1 corresponds to the second lowest IP, PE-3. The algorithm takes the configured **evi** in the service; therefore, the **evi** is mandatory, and for the same service must match in all the PEs that are part of the ES. This guarantees that the election algorithm is consistent across all the PEs of the ESI.
- **service-carving mode manual** — The user can manually decide for which **evi** identifies the PE is DF or **primary: service-carving mode manual / manual evi <start> [to <to>] primary**. The PE will be non-DF for the non-specified EVIs. If **service-carving mode manual** is configured, but no range is defined, all the services are considered to be non-DF. If a range is configured, but the **service-carving** is not **mode manual**, the range has no effect. Only two PEs are supported when **service-carving mode manual** is configured.
- **service-carving mode off** the lowest originator IP will win the election for a specified service and ES.

→ Because the **evi** is used for the service-carving algorithm, it must always be configured in a service with SAPs/SDP bindings created in an ES, irrespective of the service-carving mode (service-carving off, auto, or manual).

Although not configured as part of the ES, the **config>redundancy>bgp-evpn-multi-homing>boot-timer** allows the necessary time for the control plane protocols to come up after the PE has rebooted, and before bringing up the ESs and running the DF algorithm. Some considerations about the boot-timer:

- The **boot-timer** should use a value long enough to allow the IOMs and BGP sessions to come up before exchanging ES routes and run the DF election for each EVI (it is 10 s, by default).
- The **boot-timer** runs per EVI on the ESs in the system. While **system-up-time < boot-timer**, the system will not run the DF election for any EVI. When the boot-timer expires, the DF election for the EVI is run and, if the system is elected DF for the EVI, the **es-activation-timer** will start.
- The system will not advertise ES routes until the boot timer expires. This guarantees that the peer ES PEs do not run the DF election either, until the PE is ready to become the DF, if needed.

- The following show command displays the configured **boot-timer**, as well as the remaining timer if the system is still in boot-stage.

```
A:PE-2# show redundancy bgp-evpn-multi-homing
=====
Redundancy BGP EVPN Multi-homing Information
=====
Boot-Timer           : 10 secs
Boot-Timer Remaining : 0 secs
ES Activation Timer   : 3 secs
=====
*A:PE-2#
```

After ESI-12 is configured in PE-2 and PE-3, the lag-1 saps in both PEs can be added to the VPLS-1 service. Until the ESI-12 is successfully no shutdown, the lag saps will be kept down with a StandByForMHPProtocol flag. This is illustrated in the following example for PE-2.

```
*A:PE-2>config>service>vpls(1)# info
-----
      bgp
      exit
      bgp-evpn
      evi 1
      vxlan
      shutdown
      exit
      mpls
      ingress-replication-bum-label
      ecmp 2
      auto-bind-tunnel
      resolution any
      exit
      no shutdown
      exit
    exit
  exit
  stp
  shutdown
  exit
  sap lag-1:1 create
  exit
  no shutdown
  -----

*A:PE-2>config>service>vpls# show service id 1 sap lag-1:1 detail | match Oper
Admin State      : Up                Oper State      : Down
Admin MTU          : 1518              Oper MTU         : 1518
Oper Group         : (none)            Monitor Oper Grp : (none)
Oper Class         : 1                 Oper Weight      : 1
Stp Admin State    : Up                Stp Oper State   : Down
Admin Edge         : Disabled           Oper Edge        : N/A

*A:PE-2>config>service>vpls# show service id 1 sap lag-1:1 detail | match Flag
Flags            : StandByForMHPProtocol

*A:PE-2>config>service>vpls# /configure service system bgp-evpn ethernet-segment
"ESI-12" no shutdown
```

```
2 2015/09/13 14:47:17.91 UTC MINOR: SVCMGR #2203 Base
"Status of SAP lag-1:1 in service 1 (customer 1) changed to admin=up oper=up flags="
```

All-Active Multi-Homing Operation

To confirm that all-active multi-homing is working correctly for ESI-12, the user can use the following commands:

- **show service system bgp-evpn** — Shows the RD is used for the ES route.
- **show service system bgp-evpn ethernet-segment** — Shows all the ESs configured in the PE and their admin/operational status.
- **show service system bgp-evpn ethernet-segment name ESI-12 evi 1** — Shows the DF candidate PEs for evi 1 and whether the system is DF for evi.
- **show service system bgp-evpn ethernet-segment name ESI-12 all** — Shows all the information related to a specific ESI.

As an example, the last (all) command is shown for PE-2 and PE-3:

```
*A:PE-2# show service system bgp-evpn ethernet-segment name "ESI-12" all
=====
Service Ethernet Segment
=====
Name                : ESI-12
Admin State          : Enabled          Oper State          : Up
ESI                  : 01:00:00:00:00:12:00:00:00:01
Multi-homing         : allActive        Oper Multi-homing    : allActive
Source BMAC LSB      : <none>
Lag Id               : 1
ES Activation Timer   : 3 secs
Exp/Imp Route-Target : target:00:00:00:00:12:00

Svc Carving          : auto
ES SHG Label          : 262142
=====
EVI Information
=====
EVI          SvcId          Actv Timer Rem      DF
-----
1             1              0                  no
-----
Number of entries: 1
=====
DF Candidate list
-----
EVI          DF Address
-----
1             192.0.2.2
```

```

1                                     192.0.2.3
-----
Number of entries: 2
-----
---snipped---

A:PE-3# show service system bgp-evpn ethernet-segment name "ESI-12" all
=====
Service Ethernet Segment
=====
Name                : ESI-12
Admin State         : Enabled           Oper State          : Up
ESI                 : 01:00:00:00:00:12:00:00:00:01
Multi-homing        : allActive         Oper Multi-homing    : allActive
Source BMAC LSB     : <none>
Lag Id              : 1
ES Activation Timer  : 3 secs
Exp/Imp Route-Target : target:00:00:00:00:12:00

Svc Carving         : auto
ES SHG Label        : 262142
=====
EVI Information
=====
EVI                SvcId          Actv Timer Rem      DF
-----
1                   1              0                  yes
-----
Number of entries: 1
=====
DF Candidate list
-----
EVI                DF Address
-----
1                   192.0.2.2
1                   192.0.2.3
-----
Number of entries: 2
-----
---snipped---

```

The above command shows the ESI-12 configuration on both PEs and the result of the DF election for evi 1. This output demonstrates two points:

- The ES **Exp/Imp Route-Target** is shown as **target:00:00:00:00:12:00**. This RT is auto-derived from the ESI bytes 2 to 7 (with the type byte being byte 1). Only PE-2 and PE-3 generate this RT and therefore import each other's ES route. This auto-derived ES RT is exported/imported due to auto-created router policies that are applied to the base system BGP-EVPN instance: **_ES_EvpnEthSegRtExp** and **_ES_EvpnEthSegRtImp**. These policies are created when the first ES no shutdown is successfully executed in the system. The following output shows the ES route received on PE-2 as well as the auto-created policies.

```
*A:PE-2#
1 2015/09/13 15:24:38.28 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 70
    Flag: 0x90 Type: 14 Len: 34 Multiprotocol Reachable NLRI:
        Address Family EVPN
        NextHop len 4 NextHop 192.0.2.3
        Type: EVPN-Eth-Seg Len: 23 RD: 192.0.2.3:0 ESI: 01:00:00:00:00:12:00:00:
00:01, IP-Len: 4 Orig-IP-Addr: 192.0.2.3

    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:00:00:00:00:00:12:00
"

4 2015/09/13 15:24:38.28 UTC MINOR: SVCMMGR #2094 Base
"Ethernet Segment:ESI-12, EVI:1, Designated Forwarding state changed to:false"

*A:PE-2# show router policy "_ES_EvpnEthSegRtExp"
    entry 1
        from
            family evpn
        exit
        action accept
    exit
exit
*A:PE-2# show router policy "_ES_EvpnEthSegRtImp"
    entry 3
        from
            community "_ES_ESI-12"
            family evpn
        exit
        action accept
    exit
exit
*A:PE-2# show router policy community "_ES_ESI-12"
community "_ES_ESI-12" members "target:00:00:00:00:00:12:00"
```

- The **show service system bgp-evpn ethernet-segment name ESI-12 all** command shows the ESI-label allocated to the PE: **ES SHG Label 262142** in the CLI output for PE-3. In this example, this label is allocated by PE-3 for ESI-12 (a different one is allocated

per ESI) and advertised in the AD per-ES route for ESI-12. The following output shows the AD per-ES and AD per-EVI (for evi 1) routes sent by PE-3 and received by PE-2.

- The AD per-ES route can be identified by the **MAX-ET** in the ethernet-tag field (as per RFC 7432) and carries the ESI-label as well as the multi-homing mode (all-active in this case) in the ESI-label extended community (see [Figure 146](#)). A different AD per-ES per evi is sent in the current release.
- The AD per-EVI route will have an eth-tag 0 and will carry the service label in the NLRI.

```
*A:PE-2#
2 2015/09/13 15:24:38.27 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 80
  Flag: 0x90 Type: 14 Len: 36 Multiprotocol Reachable NLRI:
    Address Family EVPN
    NextHop len 4 NextHop 192.0.2.3
    Type: EVPN-AD Len: 25 RD: 192.0.2.3:1 ESI: 01:00:00:00:00:12:00:00:00:01
, tag: MAX-ET Label: 0

  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 16 Extended Community:
    target:64500:1
    esi-label:262142/All-Active
"

3 2015/09/13 15:24:38.27 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 80
  Flag: 0x90 Type: 14 Len: 36 Multiprotocol Reachable NLRI:
    Address Family EVPN
    NextHop len 4 NextHop 192.0.2.3
    Type: EVPN-AD Len: 25 RD: 192.0.2.3:1 ESI: 01:00:00:00:00:12:00:00:00:01
, tag: 0 Label: 4194256

  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 16 Extended Community:
    target:64500:1
    bgp-tunnel-encap:MPLS
"

*A:PE-2# show router bgp routes evpn auto-disc esi 01:00:00:00:00:12:00:00:00:01
=====
BGP Router ID:192.0.2.2          AS:64500          Local AS:64500
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
```

EVPN-MPLS Multi-Homing

```

                                l - leaked, x - stale, > - best, b - backup, p - purge
Origin codes : i - IGP, e - EGP, ? - incomplete
=====
BGP EVPN Auto-Disc Routes
=====
Flag   Route Dist.      ESI                               NextHop
      Tag                               Label
-----
u*>i  192.0.2.3:1      01:00:00:00:00:12:00:00:00:01  192.0.2.3
      0                                     LABEL 262141

u*>i  192.0.2.3:1      01:00:00:00:00:12:00:00:00:01  192.0.2.3
      MAX-ET                                     LABEL 0
-----
Routes : 2
=====

*A:PE-2# show router bgp routes evpn auto-disc esi 01:00:00:00:00:12:00:00:00:01
hunt
---snipped---
=====
BGP EVPN Auto-Disc Routes
=====
-----
RIB In Entries
-----
Network      : N/A
Nexthop      : 192.0.2.3
From         : 192.0.2.3
Res. Nexthop : 192.168.23.2
---snipped---
Community    : target:64500:1 bgp-tunnel-encap:MPLS
---snipped---
EVPN type    : AUTO-DISC
ESI          : 01:00:00:00:00:12:00:00:00:01
Tag          : 0
Route Dist.  : 192.0.2.3:1
MPLS Label   : LABEL 262141

---snipped---

Network      : N/A
Nexthop      : 192.0.2.3
From         : 192.0.2.3
Res. Nexthop : 192.168.23.2
---snipped---
Community    : target:64500:1 esi-label:262142/All-Active
---snipped---
EVPN type    : AUTO-DISC
ESI          : 01:00:00:00:00:12:00:00:00:01
Tag          : MAX-ET
Route Dist.  : 192.0.2.3:1
MPLS Label   : LABEL 0
---snipped---
```

From a service perspective, as soon as CE-1 sends some traffic, the PE learning the CE-1 MAC address will advertise it to the network. The remote PEs (PE-4 and PE-5) will create a new EVPN-MPLS ES destination to ESI-12, with two next-hops: PE-2 and PE-3. The following outputs show the following information:

- PE-4 has learned AD per-EVI/ES routes for ESI-12 from PE-2 and PE-3, as well as the CE-1 MAC address from PE-3 (because MTU-1 picked up its link to PE-3 to send CE-1 frames).

```
A:PE-4# show router bgp routes evpn auto-disc esi 01:00:00:00:00:12:00:00:00:01
=====
BGP Router ID:192.0.2.4          AS:64500          Local AS:64500
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked, x - stale, > - best, b - backup, p - purge
Origin codes  : i - IGP, e - EGP, ? - incomplete

=====
BGP EVPN Auto-Disc Routes
=====
```

Flag	Route Dist. Tag	ESI	NextHop Label
u*>i	192.0.2.2:1 0	01:00:00:00:00:12:00:00:00:01	192.0.2.2 LABEL 262141
u*>i	192.0.2.2:1 MAX-ET	01:00:00:00:00:12:00:00:00:01	192.0.2.2 LABEL 0
u*>i	192.0.2.3:1 0	01:00:00:00:00:12:00:00:00:01	192.0.2.3 LABEL 262141
u*>i	192.0.2.3:1 MAX-ET	01:00:00:00:00:12:00:00:00:01	192.0.2.3 LABEL 0

```
-----
Routes : 4
=====

A:PE-4# show router bgp routes evpn mac
=====
BGP Router ID:192.0.2.4          AS:64500          Local AS:64500
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked, x - stale, > - best, b - backup, p - purge
Origin codes  : i - IGP, e - EGP, ? - incomplete

=====
BGP EVPN MAC Routes
=====
```

Flag	Route Dist. Tag	MacAddr Mac Mobility	ESI Ip Address NextHop Label1
u*>i	192.0.2.3:1 0	00:ca:fe:ca:fe:01 Seq:0	01:00:00:00:00:12:00:00:00:01 N/A

192.0.2.3
LABEL 262141

- In the FDB for VPLS-1, PE-4 has learned the CE-1 MAC address associated with a newly created EVPN-MPLS ES destination:

```
A:PE-4# show service id 1 fdb detail
=====
Forwarding Database, Service 1
=====
ServId      MAC                      Source-Identifier      Type      Last Change
-----
1           00:ca:fe:ca:fe:01 eES:                   Evpn      09/13/15 16:47:00
                                01:00:00:00:00:12:00:00:00:01
---snipped---
```

- Due to the aliasing function, the newly created EVPN-MPLS ES destination to ESI-12 has two next-hops (PE-2 and PE-3), to which PE-4 can load-balance the unicast traffic because **ecmp 2** is configured in the VPLS-1 of PE-4. The **show service id 1 evpn-mpls esi 01:00:00:00:00:12:00:00:00:01** command shows the next-hops that the EVPN-MPLS ES destination is resolved to.

```
A:PE-4# show service id 1 evpn-mpls
=====
BGP EVPN-MPLS Dest
=====
TEP Address      Egr Label      Num. MACs      Mcast      Last Change
-----
192.0.2.2        262140         0              Yes        09/11/2015 19:16:40
                  ldp
192.0.2.3        262140         0              Yes        09/12/2015 01:58:29
                  ldp
192.0.2.5        262140         0              Yes        09/11/2015 19:16:40
                  ldp
-----
Number of entries : 3
-----
=====
BGP EVPN-MPLS Ethernet Segment Dest
=====
Eth SegId              Num. Macs      Last Change
-----
01:00:00:00:00:12:00:00:00:01  1              09/13/2015 16:47:00
--snipped--

A:PE-4# show service id 1 evpn-mpls esi 01:00:00:00:00:12:00:00:00:01
=====
BGP EVPN-MPLS Ethernet Segment Dest
=====
Eth SegId              Num. Macs      Last Change
-----
01:00:00:00:00:12:00:00:00:01  1              09/13/2015 16:47:00
=====
```

BGP EVPN-MPLS Dest TEP Info

TEP Address	Egr Label Transport	Last Change
192.0.2.2	262141 ldp	09/13/2015 16:47:00
192.0.2.3	262141 ldp	09/13/2015 16:47:00
Number of entries : 2		

- PE-3 will show the CE-1 MAC address as learned locally in sap lag-1:1 (because the data plane learning of the CE-1 MAC address happened in PE-3). For PE-2, even though it learned the MAC address from EVPN, it will install it as associated with sap lag-1:1 because the EVPN route came with ESI-12, which is a local ESI. Due to this, whenever PE-2 receives a frame with MAC DA = the CE-1 MAC address, it will be able to forward the frame locally to the sap lag-1:1. The following output shows the CE-1 MAC address as it is installed in PE-2 and PE-3:

*A:PE-2# show service id 1 fdb detail

Forwarding Database, Service 1				
ServId	MAC	Source-Identifier	Type Age	Last Change
1	00:ca:fe:ca:fe:01	sap:lag-1:1	Evpn	09/13/15 16:49:59

---snipped---

*A:PE-3# show service id 1 fdb detail

Forwarding Database, Service 1				
ServId	MAC	Source-Identifier	Type Age	Last Change
1	00:ca:fe:ca:fe:01	sap:lag-1:1	L/30	09/13/15 16:49:59

---snipped---

Single-Active Multi-Homing Concepts

There are two new concepts in EVPN single-active multi-homing: mass-withdraw and backup path. They are illustrated in [Figure 149](#).

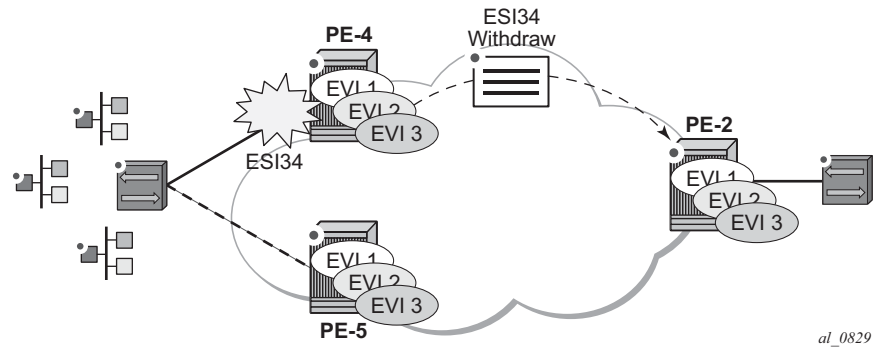


Figure 149: EVPN-MPLS Single-Active Multi-Homing: Mass-Withdraw, Backup Path

- With mass-withdraw, when ESI-34 goes down, PE-2 does not have to wait for all the MAC routes to be withdrawn to converge all the services. Instead, PE-4 will withdraw the AD per-ES routes (also the AD per-EVI and MAC routes) and that will be used at PE-2 as a notification to stop sending traffic to PE-4 for any MAC address associated with ESI-34. Note: In release 13.0.R6, a separate AD per-ES route is sent per EVI. Therefore, PE-4 will withdraw a separate route per EVI.
- With backup path, when PE-2 is notified of the ESI-34 failure due to the withdrawn AD routes, it will not flush any MAC address associated with ESI-34. Instead, it will change the next-hop of the EVPN-MPLS ES destination to the remaining PE in the ESI-34. Note: Backup path only works when there are two PEs in the same ES. If there were more than two PEs in ESI-34, PE-2 would flush all the MAC addresses upon receiving a mass-withdraw notification, because it would not know who the new active PE is.

Single-Active Multi-Homing Configuration

The single-active multi-homing configuration example is based on [Figure 147](#):

MTU-6 is connected to the EVPN network using single-active multi-homing. With the MTU-6 configuration, a VPLS service with active-standby spoke-sdp to PE-4 and PE-5 is configured. In PE-4 and PE-5, the SDP connected to MTU-6 is linked to ESI-34. Both will run the DF election algorithm for EVI 1, and the non-DF PE (PE-4 in this example) will bring down the spoke-sdp and notify MTU-6.

The following output shows the configuration of ESI-34 in PE-4 and PE-5, as well as the SDPs. The configuration of MTU-6 is also shown for completeness. It is important to keep the default **no ignore-standby-signaling** command on MTU-6 spoke-sdps because the PW switchover in MTU-6 will be triggered based on the PW status bits sent by PE-4 and PE-5.

```
A:PE-4# configure service sdp 46
A:PE-4>config>service>sdp# info
-----
        far-end 192.0.2.6
        ldp
        keep-alive
            shutdown
        exit
        no shutdown
-----

A:PE-4>config>service>sdp# /configure service system bgp-evpn
A:PE-4>config>service>system>bgp-evpn# info
-----
        ethernet-segment "ESI-34" create
            esi 01:00:00:00:00:34:00:00:00:01
            es-activation-timer 3
            service-carving
                mode auto
            exit
            multi-homing single-active
            sdp 46
            no shutdown
        exit
-----

A:PE-5# configure service sdp 56
A:PE-5>config>service>sdp# info
-----
        far-end 192.0.2.6
        ldp
        keep-alive
            shutdown
        exit
        no shutdown
-----

A:PE-5>config>service>sdp# /configure service system bgp-evpn
A:PE-5>config>service>system>bgp-evpn# info
-----
        ethernet-segment "ESI-34" create
            esi 01:00:00:00:00:34:00:00:00:01
```

```

        es-activation-timer 3
        service-carving
            mode auto
        exit
        multi-homing single-active
        sdp 56
        no shutdown
    exit
-----
A:MTU-6# configure service vpls 1
A:MTU-6>config>service>vpls# info
-----
        stp
            shutdown
        exit
        endpoint "CORE" create
        exit
        sap 1/1/1:1 create
        exit
        spoke-sdp 64:1 endpoint "CORE" create
            stp
                shutdown
            exit
            no shutdown
        exit
        spoke-sdp 65:1 endpoint "CORE" create
            stp
                shutdown
            exit
            no shutdown
        exit
        no shutdown
    -----

```

For a detailed description of the base BGP-EVPN instance and **ethernet-segment** configuration, see the [All-Active Multi-Homing Configuration on page 957](#) section. The **es-activation-timer**, **esi**, **service-carving**, **boot-timer**, and **shutdown** commands are used in the same way as for all-active multi-homing. Only the differences compared to all-active multi-homing are described below:

- **multi-homing single-active** must be configured so that the ES acts as single-active. Optionally, the **no-esi-label** attribute can be added to the **multi-homing single-active** command. This attribute controls the use of the ESI-label for single-active multi-homing. Although the ESI-label is always used in all-active multi-homing when sending BUM traffic between the PEs in the ES, it is configurable for single-active. However, the default option (using ESI-label) is recommended to avoid potential transient issues when there is a DF switchover.
- **sdp <sdp-id>** is configured so that the ES can be associated with the SDP connected to MTU-6. Although all-active multi-homing only allows lag associations to the ES, single-active allows lag, port, and sdp. In this example, sdp is the option, because the access network is MPLS-based.

Similar to the all-active multi-homing case, when configuring the service in PE-4 and PE-5, the service objects are automatically associated with the ESI-34, because they are defined in the SDPs linked to the ESI.

```
A:PE-5>config>service>vpls# info
```

```
-----
      bgp
      exit
      bgp-evpn
        evi 1
        vxlan
          shutdown
        exit
        mpls
          ingress-replication-bum-label
          ecmp 2
          auto-bind-tunnel
            resolution any
          exit
          no shutdown
        exit
      exit
      stp
        shutdown
      exit
      spoke-sdp 56:1 create
        no shutdown
      exit
      no shutdown
-----
```

In all-active multi-homing, the non-DF does not bring down the service sap associated with the ES (it only removes it from the default-multicast-list). However, in single-active multi-homing, the service spoke-sdp (or sap, if that was the object associated) is brought operationally down. The following output shows the spoke-sdp state in PE-4 (non-DF), as operationally down with the **StandbyForMHPProtocol** flag and the **Local Pw Bits** that are signaled to MTU-6:

```
A:PE-4# show service system bgp-evpn ethernet-segment name "ESI-34" evi 1
=====
EVI DF and Candidate List
=====
EVI          SvcId      Actv Timer Rem      DF  DF Last Change
-----
1             1           0                  no  09/11/2015 19:16:14
=====
DF Candidates                                Time Added
-----
192.0.2.4                                09/11/2015 19:16:40
192.0.2.5                                09/11/2015 19:16:40
-----
Number of entries: 2
=====

A:PE-4# show service id 1 base
```

```
---snipped---
-----
Service Access & Destination Points
-----
Identifier                                Type          AdmMTU  OprMTU  Adm  Opr
-----
sdp:46:1 S(192.0.2.6)                    Spok          0       8974    Up   Down
=====

A:PE-4# show service id 1 sdp 46:1 detail | match Flag
Flags                                     : StandbyForMHPProtocol

A:PE-4# show service id 1 sdp 46:1 detail | match Pw
Local Pw Bits                           : pwFwdingStandby
Peer Pw Bits                             : None
```

Single-Active Multi-Homing Operation

The same commands used in the [All-Active Multi-Homing Operation on page 962](#) section can be used for single-active; see that section.

Note: The **show service system bgp-evpn ethernet-segment name ESI-34** command shows an ethernet-segment **Oper Multi-homing** in addition to the configured **Multi-homing** mode. This occurs because, in spite of configuring the ES as all-active, it may operate as single-active if there is a mismatch between the modes advertised by PE-4 and PE-5 in the AD per-ES routes (per RFC 7432). In this example, the configured and the operational value are the same:

```
A:PE-4# show service system bgp-evpn ethernet-segment name "ESI-34"
=====
Service Ethernet Segment
=====
Name                               : ESI-34
Admin State                         : Enabled          Oper State           : Up
ESI                                : 01:00:00:00:00:34:00:00:00:01
Multi-homing                       : singleActive      Oper Multi-homing    : singleActive
Source BMAC LSB                    : <none>
Sdp Id                             : 46
ES Activation Timer                 : 3 secs
Exp/Imp Route-Target               : target:00:00:00:00:34:00

Svc Carving                         : auto
ES SHG Label                       : 262142
=====
```

As soon as CE-6 sends some traffic, the DF PE (PE-5) will learn the CE-6 MAC address and will advertise it to the network. The remote PEs (PE-2 and PE-3) will create a new EVPN-MPLS ES destination to ESI-34, but this time with only one next-hop, PE-5, because this is single-active multi-homing. The following outputs show the following information:

- PE-2 has learned AD per-EVI/ES routes for ESI-34 from PE-4 and PE-5, as well as the CE-6 MAC address from an ES EVPN-MPLS destination, which is resolved to PE-5 (the DF for ESI-34).

```
*A:PE-2# show router bgp routes evpn auto-disc esi 01:00:00:00:00:34:00:00:00:01
=====
BGP Router ID:192.0.2.2          AS:64500          Local AS:64500
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked, x - stale, > - best, b - backup, p - purge
Origin codes  : i - IGP, e - EGP, ? - incomplete
=====
BGP EVPN Auto-Disc Routes
=====
Flag  Route Dist.      ESI                      NextHop
      Tag              Label
-----
u*>i  192.0.2.4:1        01:00:00:00:00:34:00:00:00:01 192.0.2.4
      0                                LABEL 262141

u*>i  192.0.2.4:1        01:00:00:00:00:34:00:00:00:01 192.0.2.4
      MAX-ET                          LABEL 0

u*>i  192.0.2.5:1        01:00:00:00:00:34:00:00:00:01 192.0.2.5
      0                                LABEL 262141

u*>i  192.0.2.5:1        01:00:00:00:00:34:00:00:00:01 192.0.2.5
      MAX-ET                          LABEL 0

-----
Routes : 4
=====

*A:PE-2# show service id 1 fdb detail
=====
Forwarding Database, Service 1
=====
ServId  MAC              Source-Identifier      Type      Last Change
                               Age
-----
1       00:ca:fe:ca:fe:01  sap:lag-1:1           Evpn      09/13/15 18:22:41
1       00:ca:fe:ca:fe:06 eES:                   Evpn      09/13/15 18:22:41
                               01:00:00:00:00:34:00:00:00:01
-----
No. of MAC Entries: 2
-----
Legend: L=Learned O=Oam P=Protected-MAC C=Conditional S=Static
=====

*A:PE-2# show service id 1 evpn-mpls esi 01:00:00:00:00:34:00:00:00:01
=====
BGP EVPN-MPLS Ethernet Segment Dest
=====
Eth SegId              Num. Macs              Last Change
-----
01:00:00:00:00:34:00:00:00:01  1                      09/13/2015 18:22:41
=====
```

```

=====
BGP EVPN-MPLS Dest TEP Info
=====
TEP Address          Egr Label          Last Change
                    Transport
-----
192.0.2.5            262141             09/13/2015 18:22:41
                    ldp
-----
Number of entries : 1
-----
=====

```

- In this case, the local PEs, PE-4 and PE-5, will learn the CE MAC address from an EVPN-MPLS destination and a local spoke-sdp, respectively.

```

A:PE-4# show service id 1 fdb detail
=====
Forwarding Database, Service 1
=====
ServId   MAC                Source-Identifier    Type      Last Change
                    Age
-----
1        00:ca:fe:ca:fe:06 eES:                Evpn      09/13/15 18:30:14
                    01:00:00:00:00:34:00:00:00:01
-----

```

No. of MAC Entries: 1

Legend: L=Learned O=Oam P=Protected-MAC C=Conditional S=Static

```

A:PE-4# show service id 1 evpn-mpls esi 01:00:00:00:00:34:00:00:00:01
=====

```

```

BGP EVPN-MPLS Ethernet Segment Dest
=====
Eth SegId          Num. Macs          Last Change
-----
01:00:00:00:00:34:00:00:00:01  1                09/13/2015 18:30:14
=====

```

```

BGP EVPN-MPLS Dest TEP Info
=====
TEP Address          Egr Label          Last Change
                    Transport
-----
192.0.2.5            262141             09/13/2015 18:30:14
                    ldp
-----
Number of entries : 1
-----
=====

```

```

A:PE-5# show service id 1 fdb detail
=====
Forwarding Database, Service 1
=====
ServId   MAC                Source-Identifier    Type      Last Change
                    Age
-----

```

```

1          00:ca:fe:ca:fe:06 sdp:56:1          L/60      09/13/15 22:57:26
-----
No. of MAC Entries: 1
-----
Legend:  L=Learned O=Oam P=Protected-MAC C=Conditional S=Static
=====

```

Ethernet-Segment Failures

If either ES fails, a DF re-election will happen and the corresponding AD per-ES/EVI routes will be withdrawn, causing the remote PEs to modify the list of next-hops for the EVPN-MPLS ES destination. The following example illustrates a failure on the SDP between MTU-6 and PE-5 (the DF).

Step 1. A failure occurs in the LSP between MTU-6 and PE-5. This can be any event that brings the SDP down.

```

*A:PE-5#
7 2015/09/13 23:24:54.60 UTC MINOR: SVCMMGR #2303 Base
"Status of SDP 56 changed to admin=up oper=down"

```

Step 2. Immediately, PE-5 gives up the DF role and withdraws the ES route, as well as the AD routes and MAC routes. As soon as PE-4 receives any ES or AD withdraw, it will re-run the DF algorithm and, after the es-activation-timer, it will become the DF and activate its spoke-sdp.

```

*A:PE-5#
8 2015/09/13 23:24:54.60 UTC MINOR: SVCMMGR #2094 Base
"Ethernet Segment:ESI-34, EVI:1, Designated Forwarding state changed to:false"

*A:PE-5# show service system bgp-evpn ethernet-segment name "ESI-34"
=====
Service Ethernet Segment
=====
Name                : ESI-34
Admin State         : Enabled          Oper State          : Down
ESI                 : 01:00:00:00:00:34:00:00:00:01
Multi-homing        : singleActive      Oper Multi-homing    : singleActive
Source BMAC LSB     : <none>
Sdp Id              : 56
ES Activation Timer  : 3 secs
Exp/Imp Route-Target : target:00:00:00:00:34:00

Svc Carving         : auto
ES SHG Label        : 262142
=====
*A:PE-5# show service system bgp-evpn ethernet-segment name "ESI-34" evi 1
=====
EVI DF and Candidate List
=====
EVI      SvcId      Actv Timer Rem      DF  DF Last Change
-----

```

```

1          1          0          no 09/13/2015 23:24:55
=====
DF Candidates                               Time Added
-----
192.0.2.4                                09/11/2015 19:16:40
-----
Number of entries: 1
=====

*A:PE-4#
7 2015/09/13 23:24:57.57 UTC MINOR: SVCMMGR #2094 Base
"Ethernet Segment:ESI-34, EVI:1, Designated Forwarding state changed to:true"

8 2015/09/13 23:24:57.57 UTC MINOR: SVCMMGR #2326 Base
"Status of SDP Bind 46:1 in service 1 (customer 1) local PW status bits changed
to none"

9 2015/09/13 23:24:57.57 UTC MINOR: SVCMMGR #2306 Base
"Status of SDP Bind 46:1 in service 1 (customer 1) changed to admin=up oper=up f
lags="

*A:PE-4# show service system bgp-evpn ethernet-segment name "ESI-34"
=====
Service Ethernet Segment
=====
Name                : ESI-34
Admin State         : Enabled           Oper State        : Up
ESI                 : 01:00:00:00:00:34:00:00:00:01
Multi-homing        : singleActive      Oper Multi-homing   : singleActive
Source BMAC LSB     : <none>
Sdp Id              : 46
ES Activation Timer  : 3 secs
Exp/Imp Route-Target : target:00:00:00:00:34:00

Svc Carving         : auto
ES SHG Label        : 262142
=====
*A:PE-4# show service system bgp-evpn ethernet-segment name "ESI-34" evi 1
=====
EVI DF and Candidate List
=====
EVI      SvcId      Actv Timer Rem      DF  DF Last Change
-----
1          1          0          yes 09/13/2015 23:24:58
=====
DF Candidates                               Time Added
-----
192.0.2.4                                09/11/2015 19:16:40
-----
Number of entries: 1
=====

```

Step 3. The remote PEs, PE-2 and PE-3, receive the AD routes withdrawal and modify the next-hop for the EVPN-MPLS ES destination.

```

52 2015/09/13 23:17:21.60 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.5

```

```

"Peer 1: 192.0.2.5: UPDATE
Peer 1: 192.0.2.5 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 86
  Flag: 0x90 Type: 15 Len: 82 Multiprotocol Unreachable NLRI:
    Address Family EVPN
      Type: EVPN-AD Len: 25 RD: 192.0.2.5:1 ESI: 01:00:00:00:00:34:00:00:00:01
, tag: 0 Label: 0

      Type: EVPN-AD Len: 25 RD: 192.0.2.5:1 ESI: 01:00:00:00:00:34:00:00:00:01
, tag: MAX-ET Label: 0

*A:PE-2# show service id 1 evpn-mpls esi 01:00:00:00:00:34:00:00:00:01
=====
BGP EVPN-MPLS Ethernet Segment Dest
=====
Eth SegId                               Num. Macs                               Last Change
-----
01:00:00:00:00:34:00:00:00:01          1                               09/13/2015 23:17:22
=====
BGP EVPN-MPLS Dest TEP Info
=====
TEP Address                             Egr Label                             Last Change
                                Transport
-----
192.0.2.4                               262141                               09/13/2015 23:17:22
                                ldp
-----
Number of entries : 1
=====

```

The following must be considered:

- The DF election procedure is revertive, that is, when the failed SDP comes back up, PE-5 will take over again as DF and the network will re-converge.
- The DF election is triggered by the following events:
 - **config>service>system>bgp-evpn>eth-seg#** no shutdown triggers the DF election for all the services in the ES.
 - A new update/withdrawal of an ES route (containing an ESI configured locally) triggers the DF election for all the services in the ESI.
 - A new update/withdrawal of an AD per-ES route (containing an ESI configured locally) triggers the DF election for all the services associated with the list of RTs received along with the route.
 - A new update of an AD per-ES route with a change in the ESI-label extended community (single-active bit or MPLS label) triggers the DF election for all the services associated with the list of RTs received along with the route.
 - A new update/withdrawal of an AD route per-EVI (containing an ESI configured locally) triggers the DF election for that service.

BGP-EVPN Route Selection in EVPN Networks

The selection of the best route for a MAC address is as follows:

- If a PE receives more than one route for the same MAC address, the best MAC route is chosen:
 - If the route key is equal in two or more routes (that is, the mac, mac-length, ip, ip-length, RD, eth-tag), then regular BGP selection applies:
 - If Local-Pref, AS-path, ORIGIN, and MED are equal, the lowest IGP distance to the BGP next-hop is chosen (unless **ignore-nh-metric** is configured). If the BGP next-hop is resolved by an LSP, the cost from the tunnel-table is used.
 - As a last resort tie-breaker, the route with the lowest originator ID, or received from the peer with the lowest BGP Identifier, is chosen (unless **ignore-router-id** is configured and the routes being compared are EBGp routes).
 - If the mac-length, mac, ip-length, ip, eth-tag are equal, and the RD is different, the EVPN selection process is applied in the following order:
 - Conditional static macs (local protected macs)
 - EVPN static macs (remote protected macs)
 - Data plane learned MACs (regular learning on saps/sdp-bindings)
 - EVPN macs with higher SEQ number
 - Lowest IP (next-hop IP of the EVPN NLRI)
 - Lowest eth-tag (will be normally zero)
 - Lowest RD
- After a MAC route is selected, the system checks for an associated ES.
 - If it has an ES, the system uses the MAC address as the EVPN-MPLS ES destination. The ES destination is constructed based on the AD per-EVI routes received for that ES (regardless of MAC address priorities with the ES).
 - The system selects the first ECMP number of AD per-EVI routes arranged by the IP address of PEs (lower IPs are selected first).
 - If the same PE has advertised multiple RDs, the system selects the route with the lowest RD for that PE.

In the example of [Figure 147](#), PE-4 resolves the next-hops for ESI-12 as described in the second choice above, that is, because **ecmp=2**, the two available next-hops are chosen. If **ecmp** is changed to 1, PE-4 will pick up the lower IP (in the BGP next-hop). This is illustrated in the following output:

```
*A:PE-4# show service id 1 evpn-mpls esi 01:00:00:00:00:12:00:00:00:01
=====
BGP EVPN-MPLS Ethernet Segment Dest
=====
Eth SegId                               Num. Macs                               Last Change
```



```

-----
01:00:00:00:00:12:00:00:00:01  1                                09/13/2015 23:15:32
=====
BGP EVPN-MPLS Dest TEP Info
=====
TEP Address          Egr Label          Last Change
                    Transport
-----
192.0.2.2            262141             09/13/2015 23:13:16
                    ldp
192.0.2.3            262141             09/13/2015 23:15:32
                    ldp
-----
Number of entries : 2
-----
=====
*A:PE-4# configure service vpls 1 bgp-evpn mpls ecmp 1
*A:PE-4# show service id 1 evpn-mpls esi 01:00:00:00:00:12:00:00:00:01
=====
BGP EVPN-MPLS Ethernet Segment Dest
=====
Eth SegId          Num. Macs          Last Change
-----
01:00:00:00:00:12:00:00:00:01  1                09/14/2015 00:00:24
=====
BGP EVPN-MPLS Dest TEP Info
=====
TEP Address          Egr Label          Last Change
                    Transport
-----
192.0.2.2            262141             09/13/2015 23:13:16
                    ldp
-----
Number of entries : 1
-----
=====

```

Comparing EVPN Multi-homing and BGP Multi-homing

EVPN-MPLS services support EVPN-MH (EVPN multi-homing) and also BGP-MH as in chapter [BGP Multi-Homing for VPLS Networks on page 825](#). While EVPN-MH is the standard way of providing access resiliency in RFC 7432, BGP-MH is also a standard mechanism supported in VPLS or EVPN networks. The following table provides some comparison between both technologies.

Table 7: Comparing EVPN Multi-homing and BGP Multi-homing

VPN Requirements	EVPN-MH	BGP-MH	Comments
All-active MH (flow-based load-balancing)	Yes	No	EVPN-MH provides better bandwidth utilization
Single-active MH (service-based load-balancing)	Yes	Yes	
DF PE election - automatic service balancing	Yes Service-carving	No Requires vsi policies and LP manipulation	EVPN-MH provides better automation
DF PE election – manual configuration per service	Yes	No	EVPN-MH allows for manual DF config for EVIs and ISIDs (2 PEs)
Split-horizon indication in the data plane	Yes ESI-label	No	Prevents transient loops when dual-active DFs show up
DF indication in the control plane	No	Yes	BGP MH guarantees one DF at a time. EVPN relies on Timers to ensure one DF at a time
Allows multiple SAPs or SDP-bindings per service on the same site	No	Yes Through the use of SHGs	
Boot timer and site(es)-activation-timers	Yes	Yes	BGP-MH supports more granular configuration (service level)
Support for oper-groups	No	Yes	
Non-DF notification to the CE (MPLS and CFM)	Yes	Yes	Avoids blackholing

In addition to the above comparison, the following configuration excerpt compares EVPN-MH with BGP-MH on a bgp-evpn VPLS service and shows that, while EVPN-MH does not have any configuration at service level, BGP-MH is configured within the VPLS context, which gives a more granular control over the redundancy provided. Refer to the [BGP Multi-Homing for VPLS Networks on page 825](#) chapter for more information about BGP-MH.

```

config>service>system>bgp-evpn# info
-----
ethernet-segment "ESI-34" create
esi 01:00:00:00:00:34:00:00:00:01
es-activation-timer 3
service-carving
mode auto
exit
multi-homing single-active
sdp 46
no shutdown
exit

config>service>vpls# info
-----
bgp
exit
bgp-evpn
evi 1
vxlan
shutdown
exit
mpls
ingress-replication-bum-label
ecmp 2
auto-bind-tunnel
resolution any
exit
no shutdown
exit
exit
spoke-sdp 46:1 create
no shutdown
exit
no shutdown

config>service>vpls# info
-----
bgp
exit
bgp-evpn
evi 1
vxlan
shutdown
exit
mpls
ingress-replication-bum-label
ecmp 2
auto-bind-tunnel
resolution any
exit
no shutdown
exit
exit
site "site-1" create
no shutdown
site-id 1
spoke-sdp 46:1
site-activation-timer 3
exit
spoke-sdp 46:1 create
no shutdown
exit
no shutdown

```

Proxy-ARP/ND Configuration for EVPN-MPLS Networks

Although not strictly a BGP-EVPN configuration, **vpls>proxy-arp** and **vpls>proxy-nd** functions are typically enabled along with EVPN-MPLS in order to reduce the amount of flooding in the network. The proxy-ARP/ND agent in the VPLS service will snoop ARP-Requests and/or Neighbor Solicitation messages and will reply to those messages locally (if the information is known) without having to flood the requests to the network.

```

A:PE-2# configure service vpls 1 proxy-arp ?
- no proxy-arp
- proxy-arp

[no] age-time          - Configure aging timer for proxy ARP entries
dup-detect             - Configure anti-spoofing MAC address information
[no] dynamic-arp-po*   - Configure population of dynamic proxy ARP entries
[no] garp-flood-evpn   - Configure to flood GARP request/replys into EVPN
[no] send-refresh      - Configure send refresh time

```

```

[no] shutdown          - Administratively enable/disable proxy ARP configuration
[no] static             - Configure static IP address to MAC address associations
    table-size         - Configure the maximum number of entries in the proxy ARP table
[no] unknown-arp-re*    - Configure to flood unknown ARP request

A:PE-2# configure service vpls 1 proxy-nd ?
  - no proxy-nd
  - proxy-nd

[no] age-time           - Configure aging timer for proxy ND entries
    dup-detect          - Configure anti-spoofing MAC address information
[no] dynamic-nd-pop*    - Configure population of dynamic proxy ND entries
    evpn-nd-advert*     - Configure EVPN Neighbor Discovery advertisements
[no] host-unsolicit*    - Configure whether to flood evpn with host neighbor advertisement
[no] router-unsolic*    - Configure whether to flood evpn with router neighbor advertisement
[no] send-refresh       - Configure send refresh time
[no] shutdown          - Administratively enable/disable proxy ND configuration
[no] static             - Configure static IP address to MAC address associations
    table-size         - Configure the maximum number of entries in the proxy ND table
[no] unknown-ns-flo*    - Configure to flood unknown ND solicitation

```

When proxy-ARP/ND is enabled, the following configuration guidelines must be followed:

- **dynamic-arp-populate** or **dynamic-nd-populate** should be used only in networks with a consistent configuration of this command in all PEs.
- When using **dynamic-arp-populate/dynamic-nd-populate**, the **age-time** value should be configured to a value equal to three times the **send-refresh** value. This will help reduce the EVPN withdrawals and re-advertisements in the network.
- With large **age-time** values, it would be sufficient to configure the **send-refresh** value to half of the **proxy-ARP/ND age-time** or **FDB age-time**.
- In scaled environments (with thousands of services), it is not recommended to set the send-refresh value to less than 300 s. In such scenarios, Alcatel-Lucent recommends using a minimum proxy-ARP/ND **age-time** and FDB **age** of 900 s.
- The use of the following commands reduces or suppresses the ARP/ND flooding in an EVPN network, because EVPN MAC routes replace the function of the regular data plane ARP/ND messages:
 - **no garp-flood-evpn**
 - **no unknown-arp-request-flood-evpn**
 - **no unknown-ns-flood-evpn**
 - **no host-unsolicited-na-flood-evpn**
 - **no router-unsolicited-na-flood-evpn**
- Alcatel-Lucent recommends using the above commands only in EVPN networks where the CEs are routers directly connected to an SR OS node acting as the PE. Networks using aggregation switches between the host/routers and the PEs should flood GARP/ND messages in EVPN to make sure the remote caches are updated and BGP does not miss the advertisement of these entries.

- When the **anti-spoof-mac** is used with proxy-ARP/ND, ingress filters (in the access SAPs/SDP-bindings) should be configured to drop all traffic with destination anti-spoof-mac. The same MAC address should be configured in all PEs where dup-detect is active.
- When proxy-ND is used, the configuration of the following commands should be consistent in all the PEs in the network:
 - **router-unsolicited-na-flood-evpn**
 - **host-unsolicited-na-flood-evpn**
 - **evpn-nd-advertise**
- Because EVPN does not propagate the **router** flag in IPv6--> MAC address advertisements, in a mixed network with hosts and routers where **evpn-nd-advertise router** is configured, unsolicited host NA messages should be flooded so that the entire network gets to learn all of the host and router ND entries. In the same way, **evpn-nd-advertise host** should be configured so that unsolicited router NA messages are flooded.

Finally, along with proxy-ARP/ND, **vpls>discard-unknown** may be used in some EVPN-MPLS deployments where all the CEs are routers and they announce themselves to the network by sending GARPs or NAs (Neighbor Solicitation messages). According to RFC 7432, whether or not to flood packets to unknown destination MAC addresses should be an administrative choice, depending on how learning happens between CEs and PEs. **Discard-unknown** provides that administrative choice in case all the MAC addresses in an EVI can be learned even before any traffic is exchanged.

Proxy-ARP/ND along with **discard-unknown** helps reduce the BUM traffic in an EVPN network significantly; however, their use must be analyzed and considered, depending on the type of CEs in the EVI.

An example of proxy-ARP configuration in PE-2 is shown below. The same configuration should be added to PE-3/4/5. When a new ARP message is received on any of the PEs, they will learn the IP-MAC address pair and will advertise it to the network.

```
A:PE-2>config>service>vpls# info
-----
---snipped---
    bgp
    exit
    bgp-evpn
        evi 1
        vxlan
            shutdown
        exit
    mpls
        split-horizon-group "CORE"
        ingress-replication-bum-label
        ecmp 2
        auto-bind-tunnel
            resolution any
        exit
        no shutdown
    exit
```

```

exit
stp
    shutdown
exit
---snipped---
sap lag-1:1 create
exit
proxy-arp
    age-time 900
    send-refresh 300
    dynamic-arp-populate
    no shutdown
exit
no shutdown
-----

```

Note: Enabling proxy-ARP increases the number of MAC/IP routes being sent by the PEs. This is due to the following reasons:

- An additional MAC/IP route will be advertised per new learned IP1-MAC address pair, regardless of having advertised the same MAC address already.
- A MAC per VPLS service will be advertised with a system MAC address. That MAC address will be used as MAC SA for proxy-ARP confirm messages when an IP moves to a different PE.

The following output shows the MAC/IP routes on PE-2 when proxy-ARP is enabled in the network.

```

A:PE-2# show router bgp routes evpn mac
=====
BGP Router ID:192.0.2.2      AS:64500      Local AS:64500
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked, x - stale, > - best, b - backup, p - purge
Origin codes  : i - IGP, e - EGP, ? - incomplete
=====
BGP EVPN MAC Routes
=====
Flag  Route Dist.      MacAddr      ESI
      Tag             Mac Mobility  Ip Address
                               NextHop
                               Label1
-----
u*>i  192.0.2.3:1      00:ca:fe:ca:fe:01 01:00:00:00:00:12:00:00:00:01
      0              Seq:0            10.0.0.1
                               192.0.2.3
                               LABEL 262141

u*>i  192.0.2.3:1      00:ca:fe:ca:fe:01 01:00:00:00:00:12:00:00:00:01
      0              Seq:0            N/A
                               192.0.2.3
                               LABEL 262141

```

```

u*>i 192.0.2.3:1      d8:48:ff:00:03:3a ESI-0
      0              Static      N/A
                               192.0.2.3
                               LABEL 262141

u*>i 192.0.2.4:1      d8:46:ff:00:03:3a ESI-0
      0              Static      N/A
                               192.0.2.4
                               LABEL 262141

u*>i 192.0.2.5:1      00:ca:fe:ca:fe:06 01:00:00:00:00:34:00:00:00:01
      0              Seq:0       10.0.0.6
                               192.0.2.5
                               LABEL 262141

u*>i 192.0.2.5:1      00:ca:fe:ca:fe:06 01:00:00:00:00:34:00:00:00:01
      0              Seq:0       N/A
                               192.0.2.5
                               LABEL 262141

u*>i 192.0.2.5:1      d8:49:ff:00:03:3a ESI-0
      0              Static      N/A
                               192.0.2.5
                               LABEL 262141

```

```

-----
Routes : 7
=====

```

Troubleshooting and Debug Commands

When troubleshooting an EVPN-MPLS network, the following show commands and debug commands are recommended, as already discussed throughout this chapter:

- **show redundancy bgp-evpn-multi-homing**
- **show router bgp routes evpn (and filters)**
- **show service evpn-mpls [<TEP ip-address>]**
- **show service id bgp-evpn**
- **show service id evpn-mpls (and modifiers)**
- **show service id fdb (and modifiers)**
- **show service system bgp-evpn**
- **show service system bgp-evpn ethernet-segment (and modifiers)**
- **debug router bgp update**
- **log-id 99**

In addition to the above commands, the following tools dump commands may also help:

- **tools dump service evpn usage** — This command shows the amount of EVPN-MPLS (and EVPN-VXLAN) destinations consumed in the system.
- **tools dump service system bgp-evpn ethernet-segment <name> evi <[1..65535]> df** — This command computes the DF election for a specific ESI and EVI. Note: The **show service system bgp-evpn ethernet-segment** commands show whether the local PE is DF or non-DF for a specific EVI, but it does not show who the DF is if it is not the local PE. In case of more than 2 PEs in the ES, this command may be especially useful.

Some examples are provided below for PE-2. PE-2 is showing seven EVPN-MPLS destinations due to the following:

- Each remote PE consumes one EVPN-MPLS destination for unicast (if they advertise MAC/IP routes to PE-2 and the ingress-replication-bum-label is configured in all the PEs). PE-2 has three remote unicast EVPN-MPLS destinations.
- Each remote PE consumes one EVPN-MPLS destination for multicast (if they advertise inclusive multicast routes to PE-2). PE-2 has three remote multicast EVPN-MPLS destinations.
- Each remote ES consumes one EVPN-MPLS destination (it is only one per ES, regardless of the multi-homing mode and the number of PEs in the ES). PE-2 has one remote ES (ESI-34).


```
A:PE-2# tools dump service evpn usage
EVPN usage statistics at 002 06:29:38.940:
```

```

MPLS-TEP : 3
VXLAN-TEP : 0
Total-TEP : 3/ 16383

Mpls Dests (TEP, Egress Label + ES + ES-BMAC) : 7
Vxlan Dests (TEP, Egress VNI) : 0
Total-Dest : 7/131071

Sdp Bind + Evpn Dests : 7/196607
ES L2/L3 PBR : 0/ 32767

```

```
*A:PE-2# tools dump service system bgp-evpn ethernet-segment "ESI-12" evi 1 df
```

```
[09/14/2015 01:40:28] Computed DF: 192.0.2.3 (Remote) (Boot Timer Expired: Yes)
```

Conclusion

SR OS has a full RFC 7432 EVPN-MPLS implementation including single-active and all-active multi-homing. This example has shown how to configure and operate EVPN-MPLS for a simple non multi-homing configuration as well as a multi-homing configuration. Other topics, such as the integration of VPLS objects with EVPN-MPLS and proxy-arp/nd, have also been discussed.

EVPN for PBB over MPLS (PBB-EVPN)

In This Chapter

This section provides information about EVPN for PBB over MPLS (PBB-EVPN).

Topics in this section include:

- [Applicability on page 992](#)
- [Overview on page 993](#)
- [Configuration on page 996](#)
- [Conclusion on page 1031](#)

Applicability

This chapter is applicable to 7750 SR-7/12, 7750 SR-a4/8, 7750 SR-12E, 7450 ESS-7/12, XRS-20/16c, and 7750-c4/12. Ethernet Virtual Private Networks (EVPN) for Multi-Protocol Label Switching (MPLS) tunnels requires IOM3-XP/IMM or higher-based line cards and chassis-mode D.

The configuration was tested on SR OS release 13.0.R6.

Important note: A prerequisite is to read the [EVPN for MPLS Tunnels on page 937](#) chapter.

Overview

EVPN for Provider Backbone Bridging (PBB) over MPLS (hereafter called PBB-EVPN) is specified in RFC 7623, *Provider Backbone Bridging Combined with Ethernet VPN (PBB-EVPN)*. It provides a simplified version of EVPN-MPLS for cases where the network requires very high scalability and does not need all the advanced features supported by EVPN-MPLS (but still requires single-active and all-active multi-homing capabilities). [Table 8](#) provides a comparison between the capabilities of EVPN and PBB-EVPN in SR OS, and may help to choose between them when designing a VPN service.

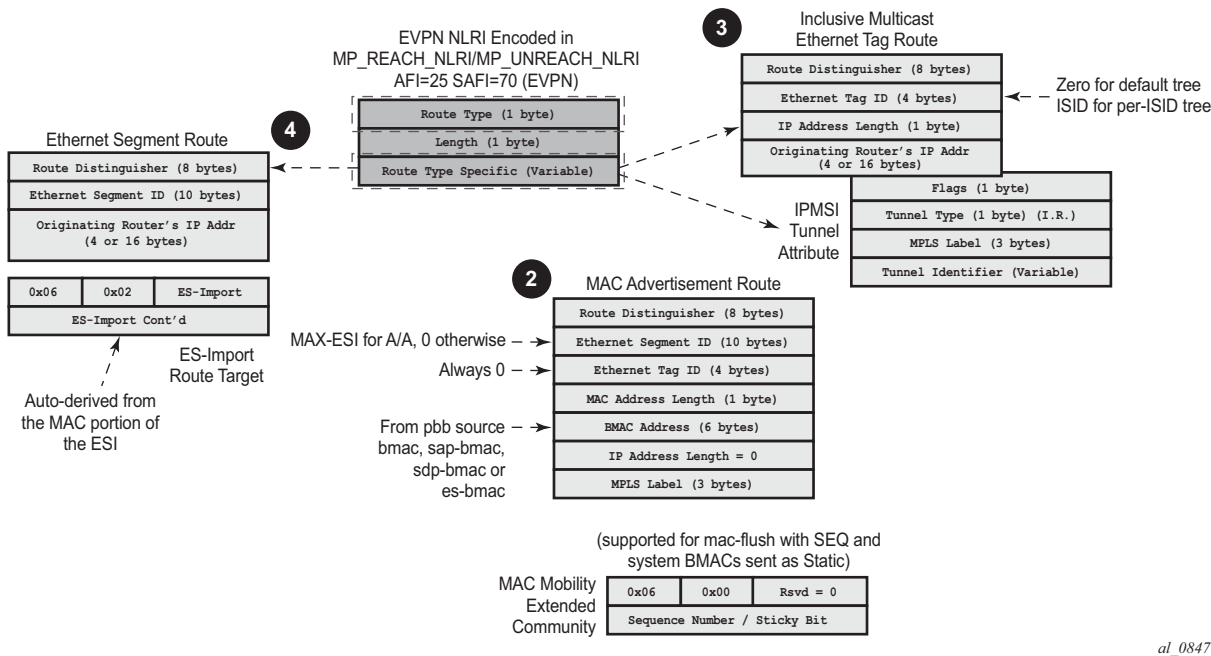
Table 8: EVPN and PBB-EVPN SR OS Feature Comparison

VPN requirements	EVPN	PBB-EVPN	Comments
All-active Multi-Homing (MH) (flow-based load-balancing)	Yes	Yes	Allows better bandwidth utilization
Single-active MH (service-based load-balancing)	Yes	Yes	
Ethernet Local Area Network (ELAN) and point-to-point E-Line services	No	Yes	Note: EVPN only supports ELAN in release 13.0.R6 and later
Inter-subnet-forwarding	Yes	No	Allows combined layer 2/layer 3 services. EVPN-Virtual eXtensible LAN (EVPN-VXLAN) is needed for layer 3 in the current release.
Proxy-address resolution protocol / neighbor discovery (Proxy-ARP/ND) and IP-duplication protection	Yes	No	Allows broadcast, unknown and multicast (BUM) traffic reduction and better security
Customer MAC (CMAC) protection	Yes	No	Allows protecting key static CMACs
Data Center integration	Yes	No	Integration with VXLAN and Nuage virtualized services directory (VSD)
Control plane overhead	Medium	Low	PBB-EVPN only advertises Backbone MACs (BMACs) and no route type 1s
Confinement of CMAC learning	No	Yes	CMACs are only learned on PEs with flows using those CMACs
CMAC summarization	No	Yes	Aggregation of CMACs into BMACs

PBB-EVPN is a combination of 802.1ah PBB and RFC7432, *BGP MPLS-Based Ethernet VPN*, EVPN-MPLS and reuses the PBB-Virtual private LAN service (VPLS) service model, where border gateway protocol BGP-EVPN is enabled in the backbone VPLS (B-VPLS) domain. EVPN is used as the control plane in the B-VPLS domain to control the distribution of BMACs and set up per-backbone service instance identifier (ISID) flooding trees for service instance VPLS (I-VPLS) services. The learning of the CMACs, either on local SAPs/SDP-bindings or associated with remote BMACs, is still performed in the data plane. Only the learning of BMACs in the B-VPLS is performed through BGP.

The SR OS PBB-EVPN implementation supports I-VPLS and PBB-Epipe services, including single-active and all-active multi-homing.

Because PBB-EVPN is based on the same control plane model as EVPN for MPLS, Alcatel-Lucent recommends reading the [EVPN for MPLS Tunnels on page 937](#) chapter before configuring PBB-EVPN. PBB-EVPN uses a subset of the BGP-EVPN routes described in [EVPN for MPLS Tunnels on page 937](#) as shown in [Figure 150](#).



al_0847

Figure 150: EVPN Route Types

When no EVPN multi-homing is used in the network, only the base routes are used. Routes type 2 and 3 are considered the base and mandatory routes:

- Route type 2 — (B) MAC route — In PBB-EVPN, this route is used for the advertisement of BMACs that will be installed in the remote forwarding data bases (FDBs). There are no

IP addresses advertised in PBB-EVPN. The MAC mobility extended community is used for advertising system BMACs as **protected** (with the sticky bit set) and it is also used for CMAC flush in some single-homing scenarios that will be described later.

- Route type 3 — Inclusive Multicast route — This route is used for the advertisement of the I-VPLS ISIDs (no Epipes) and the desired multicast tree for each of them. The ISIDs are encoded in the ethernet-tag field of the network layer reachability information (NLRI). When the B-VPLS is created with no shutdown, an Inclusive Multicast route with ISID = 0 is advertised. This is for the creation of the default multicast tree.

When EVPN multi-homing is used in an ISID, route type 4s are used. In PBB-EVPN, there is no route type 1 advertised when multi-homing is used on the ISID services (I-VPLS and Epipes). Only route type 4 (Ethernet segment (ES) route) is used, and in the same way as it is for EVPN-MPLS. See the [EVPN for MPLS Tunnels on page 937](#) example for more information about ES routes, how they are formed, and how their RT/RD values are populated.

Configuration

This example describes the basic PBB-EVPN configuration first (without multi-homing) and how the flood containment is handled in PBB-EVPN. Flood containment refers to the efficient distribution of the BUM traffic generated for an ISID.

Networks are not always greenfield, so a smooth migration of PBB-EVPN from PBB-VPLS is required to minimize the effect on existing services. This example also describes this migration, starting from a common PBB-VPLS configuration.

Finally, this example describes the configuration of PBB-EVPN multi-homing.

The same setup described in the VPN for MPLS tunnels example is used:

- Four PEs in the core (PE-2, PE-3, PE-4, and PE-5).
- The PEs are interconnected in the same way as explained in [EVPN for MPLS Tunnels on page 937](#) same IP addressing, IS-IS, transport LDP, and BGP peering configuration. There is not any difference with the basic infrastructure. See the EVPN for MPLS Tunnels example if more information is required.
- When configuring multi-homing, MTU-1 and MTU-6 are connected to the core.

PBB-EVPN Configuration without Multi-Homing

[Figure 151](#) shows the network setup used in this chapter.

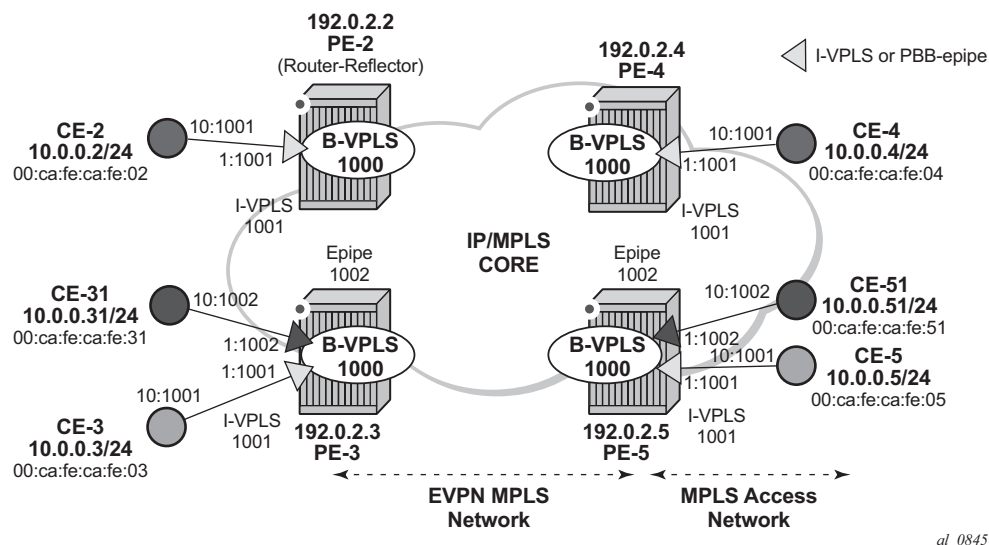


Figure 151: PBB-EVPN Network without Multi-Homing

When configuring PBB-EVPN:

- There is no difference at the access side (I-VPLS and Epipe configuration) compared to other PBB technologies supported in SR OS, such as shortest path bridging for MAC (SPBM) or PBB-VPLS.
- The B-VPLS becomes an EVPN-MPLS service, where `bgp-evpn mpls` is added.

The following output shows an example of a basic configuration in PE-3. B-VPLS 1000 is `bgp-evpn` enabled and I-VPLS 1001 / Epipe 1002 is linked to B-VPLS 1000.

```
A:PE-3>config>service# info
-----
---snipped---
vpls 1000 customer 1 b-vpls create
  service-mtu 2000
  pbb
    source-bmac 00:00:00:00:00:03
  exit
  bgp
  exit
  bgp-evpn
    evi 1000
    vxlan
      shutdown
    exit
    mpls
      auto-bind-tunnel
      resolution any
```

PBB-EVPN Configuration without Multi-Homing

```
        exit
        no shutdown
    exit
    stp
        shutdown
    exit
    no shutdown
exit
vpls 1001 customer 1 i-vpls create
    pbb
        backbone-vpls 1000
    exit
    stp
        shutdown
    exit
    sap 1/1/1:1001 create
    exit
    no shutdown
exit
epipe 1002 customer 1 create
    pbb
        tunnel 1000 backbone-dest-mac 00:00:00:00:00:05 isid 1002
    exit
    sap 1/1/1:1002 create
    exit
    no shutdown
exit
```

In the preceding output, there is no new configuration needed for I-VPLS/Epipe services. As for the B-VPLS, the output shows the minimum configuration required. If needed, the following parameters can be modified under **bgp-evpn**:

- bgp-evpn>evi
- bgp-evpn>cfm-mac-advertisement
- bgp-evpn>mac-advertisement
- bgp-evpn>mac-duplication
- bgp-evpn>mpls>ingress-replication-bum-label
- bgp-evpn>mpls>ecmp
- bgp-evpn>mpls>auto-bind-tunnel
- bgp-evpn>mpls>control-word
- bgp-evpn>mpls>split-horizon-group
- bgp-evpn>mpls>shutdown

A detailed description of these commands is included in [EVPN for MPLS Tunnels on page 937](#). In addition to the above, the following **service>(b-)vpls>pbb** commands are relevant for PBB-EVPN in the B-VPLS service:

- **force-qtag-forwarding** allows the transparent transport of the customer 802.1p bits across the B-VPLS services.
- **source-bmac** can modify the source BMAC for all the PBB packets containing traffic from non-multi-homed I-VPLS and Epipe services.
- **use-es-bmac** instructs the system to use an ES-specific BMAC for traffic coming from an ES on an I-VPLS or Epipe.
- **use-sap-bmac** instructs the system to use a SAP-specific BMAC for traffic coming from an MC-LAG I-VPLS/Epipe SAP.

Flood Containment for I-VPLS Services

In general, PBB technologies in SR OS support a way to contain flooding for a specified I-VPLS ISID, so that BUM traffic for that ISID only reaches the PEs where the ISID is locally defined. Each PE creates a multicast forwarding information base (MFIB) per I-VPLS ISID on the B-VPLS instance. That MFIB supports SAP/SDP-binding endpoints that can be populated by:

- MMRP in regular PBB-VPLS
- IS-IS in SPBM

In PBB-EVPN, B-VPLS EVPN destinations can be added to the MFIBs using EVPN Inclusive Multicast Ethernet tag routes when they include the ISID in the Ethernet-tag. By default, when a B-VPLS is successfully enabled (no shutdown), the PE advertises:

- An Inclusive Multicast route for ISID = 0 — This allows the remote PEs to add the advertising PE to the default-multicast-list for the B-VPLS.
- An Inclusive Multicast route for each local ISID defined in the system (a local ISID includes configured I-VPLS and static-ISIDs) — This allows the remote PEs to create MFIB entries in the B-VPLS for the received ISIDs.

Because EVPN destinations, B-SAPs, and B-spoke-SDPs can coexist in the same B-VPLS, be aware of the different flooding lists created and how they are used in a B-VPLS. [Figure 152](#) illustrates this concept with an example for B-VPLS 1000 in PE-1. The assumptions are:

- I-VPLS 1001 is created in PE-1, PE-2, and PE-4 only.
- PE-1, PE-2, PE-3, PE-4, and PE-5 support **bgp-evpn** in B-VPLS 1000.
- PE-6 and PE-7 only support spoke-SDPs.
- PE-1 is connected to all six PEs.

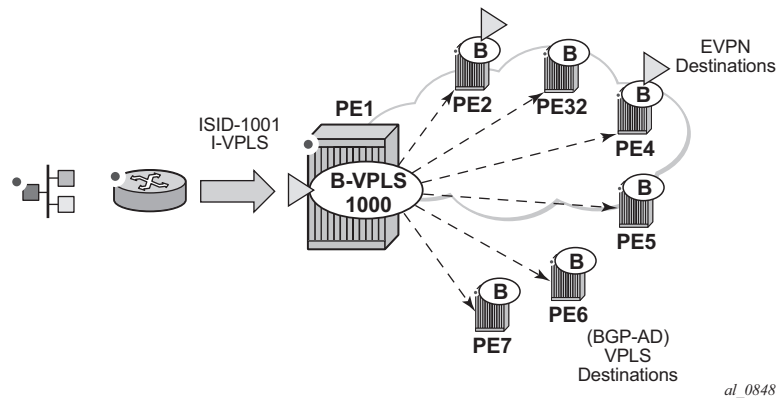


Figure 152: PBB-EVPN — Flooding Lists

In this situation, PE-1 creates two flooding lists in B-VPLS 1000:

- Default-multicast-list — composed of:
 - All the EVPN PEs that advertised ISID = 0 (PE-2, PE-3, PE-4, PE-5).
 - All the b-spoke-sdps (or b-saps) (PE-6, PE-7).
 - All the EVPN PEs that advertised ISID 1001 and no ISID 0 (if an isid-policy is created in PE-1 stating **use-def-mcast** for ISID 1001). Note: third-party PEs may not advertise ISID = 0, but only non-zero ISIDs.
- MFIB for ISID 1001 is composed of:
 - All the EVPN PEs that advertised ISID 1001 (PE-2 and PE-4) unless there is an isid-policy in PE-1 stating **use-def-mcast** for ISID 1001.
 - Static-ISIDs defined in manual b-spoke-sdps and b-saps (static-isids cannot be created on bgp-ad auto-discovered b-spoke-sdps).

Based on the above, when BUM traffic is sent to I-VPLS 1001 on PE-1:

- The traffic is encapsulated in PBB with the group BMAC for ISID 1001 and sent (by default) to the MFIB created for ISID 1001 (PE-2 and PE-4).
- If an isid-policy is added with **use-def-mcast** for ISID 1001, the BUM traffic is encapsulated in PBB with the group BMAC for ISID 1001 and sent to the default-multicast-list, that is, all six remote PEs.

Referring to [Figure 151](#), the following output illustrates the use of the isid-policy in PBB-EVPN. PE-2 does not have any isid-policy configured; when it receives BUM traffic from the local I-VPLS 1001, it uses the MFIB for ISID 1001:

```
*A:PE-2>config>service>vpls# info
```

```

-----
service-mtu 2000
pbb
    source-bmac 00:00:00:00:00:02
exit
bgp
exit
bgp-evpn
    evi 1000
    vxlan
        shutdown
    exit
    mpls
        auto-bind-tunnel
        resolution any
    exit
    no shutdown
exit
exit
stp
    shutdown
exit
no shutdown
-----

*A:PE-2>config>service>vpls# /show service id 1000 mfib
=====
Multicast FIB, Service 1000
=====
Source Address  Group Address          Sap/Sdp Id                Svc Id  Fwd
                                           Blk
-----
*              01:1E:83:00:03:E9    b-eMpls:192.0.2.3:262134   Local   Fwd
                                           b-eMpls:192.0.2.4:262130   Local   Fwd
                                           b-eMpls:192.0.2.5:262132   Local   Fwd
-----
Number of entries: 1
=====

```

However, an isid-policy can be added to modify this behavior and allow PE-2 to use the default-multicast-list. If I-VPLS 1001 exists in all the remote PEs (as in this example), using the default-multicast list is as efficient as using the MFIB and saves expensive MFIB resources. In the following output, as soon as the isid-policy is added, the MFIB entries for ISID 1001 are removed and PE-2 starts using the default-multicast list. The **tools dump service id 1000 evpn-mpls default-multicast-list** command shows the EVPN destinations that are part of the default-multicast list:

```

*A:PE-2>config>service>vpls# info
-----
service-mtu 2000
pbb
    source-bmac 00:00:00:00:00:02
exit
bgp
exit
bgp-evpn

```

Flood Containment for I-VPLS Services

```

    evi 1000
    vxlan
        shutdown
    exit
    mpls
        auto-bind-tunnel
        resolution any
    exit
    no shutdown
    exit
exit
stp
    shutdown
exit
isis-policy
    entry 10 create
        use-def-mcast
        range 1001 to 2000
    exit
exit
no shutdown
-----

*A:PE-2# tools dump service id 1000 evpn-mpls default-multicast-list
-----
TEP Address                                Egr Label
                                           Transport
-----
192.0.2.3                                262134
                                           ldp
192.0.2.4                                262130
                                           ldp
192.0.2.5                                262132
                                           ldp
-----

*A:PE-2>config>service>vpls# /show service id 1000 mfib
=====
Multicast FIB, Service 1000
=====
Source Address  Group Address      Sap/Sdp Id              Svc Id  Fwd
                                           Blk
-----
Number of entries: 0
=====
```

PBB-VPLS to PBB-EVPN Migration

The principles required for migrating a PBB-VPLS network to PBB-EVPN are explained in section the **VPLS to EVPN-MPLS Integration** of the [EVPN for MPLS Tunnels on page 937](#) example. Those principles are also applicable to EVPN destinations and spoke-SDPs in the B-VPLS and can be summarized in three points:

1. Systems with an EVPN destination and SDP-binding to the same far-end IP bring down the SDP-binding. This avoids loops when both constructs exist in the same network.
2. SDP-bindings and EVPN destinations can be placed in the same **split-horizon-group (SHG)**. When traffic from an SDP-binding/EVPN destination belonging to that SHG is received on a PE, it is never forwarded to another SDP-binding/EVPN destination on the same SHG.
3. MAC addresses learned on an SDP-binding or SAP, that belong to an SHG where EVPN destinations are also created, are not advertised in BGP-EVPN.

Based on those principles, this section describes how to migrate a PBB-VPLS network to PBB-EVPN. The network in [Figure 151](#) represents a regular PBB-VPLS network that needs to be migrated to PBB-EVPN.

In that network, the four PEs are running BGP-AD and TLDP for the discovery and setup of the pseudowires in the B-VPLS instance. The advantage of this configuration is that the migration can be done node-by-node and with minimum impact on customer service.

Initial Configuration

Initially, the network is configured for PBB-VPLS with BGP-AD in B-VPLS 1000. An EVPN family is to be added. At the access, I-VPLS 1001 is connected to the CEs. As an example, the configuration in PE-3 is shown below. An equivalent configuration exists in the other three PEs.

Note: The EVPN family is added to the BGP configuration since PBB-EVPN uses this address family. Assuming there are redundant route reflectors (RRs), the addition of EVPN can be done without service impact. In this example the assumption is that the PEs are already configured with the EVPN family.

```
*A:PE-3>config>router>bgp# info
-----
      vpn-apply-import
      vpn-apply-export
      min-route-advertisement 1
      enable-peer-tracking
      rapid-withdrawal
      split-horizon
      rapid-update evpn
      group "internal"
```

PBB-VPLS to PBB-EVPN Migration

```

        family l2-vpn evpn
        type internal
        neighbor 192.0.2.2
        exit
    exit
    no shutdown
-----
*A:PE-3>config>router>bgp# /configure service
*A:PE-3>config>service# info
-----
---snipped---
    pw-template 1 create
        split-horizon-group "CORE"
    exit
    exit
---snipped---

    vpls 1000 customer 1 b-vpls create
        service-mtu 2000
        pbb
            source-bmac 00:00:00:00:00:03
        exit
        bgp
            pw-template-binding 1
        exit
        exit
        bgp-ad
            vpls-id 64500:1000
            no shutdown
        exit
        stp
            shutdown
        exit
        no shutdown
    exit
    vpls 1001 customer 1 i-vpls create
        pbb
            backbone-vpls 1000
        exit
        exit
        stp
            shutdown
        exit
        sap 1/1/1:1001 create
        exit
        no shutdown
    exit
-----

*A:PE-3# show service id 1000 base
=====
Service Basic Information
=====
Service Id      : 1000                Vpn Id          : 0
Service Type    : b-VPLS
---snipped---

Oper Backbone Src : 00:00:00:00:00:03
```



```

Use SAP B-MAC      : Disabled
i-Vpls Count       : 1
Epipe Count        : 0
Use ESI B-MAC      : Disabled

```

Service Access & Destination Points

Identifier	Type	AdmMTU	OprMTU	Adm	Opr
sdp:17405:4294967293 SB(192.0.2.5)	BgpAd	0	8974	Up	Up
sdp:17406:4294967294 SB(192.0.2.4)	BgpAd	0	8974	Up	Up
sdp:17407:4294967295 SB(192.0.2.2)	BgpAd	0	8974	Up	Up

* indicates that the corresponding row element may have been truncated.

Multiple MAC registration protocol (MMRP) is not used in the B-VPLS instance. If it were enabled, MMRP would have to be disabled in the network before this migration. If there are ISIDs using B-VPLS SDP-bindings to reach some remote locations and B-VPLS EVPN destinations to reach others, the default-multicast list must be used in the current release (the MFIB cannot be used if there is a mix of both types). Therefore, during the migration process, the ISIDs must be added to the default-multicast-list.

Step 1. Add service-level SHG (if not already there).

From the first node being migrated to PBB-EVPN to all nodes migrated, PBB-VPLS and PBB-EVPN have to coexist within the same meshed network. That is, EVPN-MPLS destinations and SDP-bindings need to be defined in the same **split-horizon-group**. Therefore, if there is no **split-horizon-group** defined in the B-VPLS, the first step is to add it. In our example, the **split-horizon-group** is defined at the **config>service>pw-template>level**; therefore, it has to be added at the B-VPLS level.

Notes:

- When the **service>split-horizon-group** is removed, an eval-pw-template must be performed.
- After adding the **split-horizon-group** at the service level, an eval-pw-template must be performed again so that the SDP-bindings take the new SHG configuration.
- During the time between the **split-horizon-group** being removed and added back again, the SDP-bindings can forward BUM traffic to each other, so this operation must be done carefully to avoid loops.

Assuming that the first node to be migrated is PE-3, the following output shows the procedure for adding the **split-horizon-group** at the service level.

```
*A:PE-3>config>service>pw-template# no split-horizon-group
```

```

*A:PE-3>config>service>vpls# /tools perform service id 1000 eval-pw-template 1 allow-ser-
vice-impact
eval-pw-template succeeded for Svc 1000 17405:4294967293 Policy 1
eval-pw-template succeeded for Svc 1000 17406:4294967294 Policy 1

```

PBB-VPLS to PBB-EVPN Migration

```
eval-pw-template succeeded for Svc 1000 17407:4294967295 Policy 1

*A:PE-3>config>service>vpls# split-horizon-group CORE create
*A:PE-3>config>service>vpls>split-horizon-group# exit
*A:PE-3>config>service>vpls# bgp pw-template-binding 1 split-horizon-group "CORE"

*A:PE-3# /tools perform service id 1000 eval-pw-template 1 allow-service-impact
eval-pw-template succeeded for Svc 1000 17405:4294967293 Policy 1
eval-pw-template succeeded for Svc 1000 17406:4294967294 Policy 1
eval-pw-template succeeded for Svc 1000 17407:4294967295 Policy 1

*A:PE-3>config>service>vpls# info
-----
    service-mtu 2000
    pbb
        source-bmac 00:00:00:00:00:03
    exit
    split-horizon-group "CORE" create
    exit
    bgp
        pw-template-binding 1 split-horizon-group "CORE"
        exit
    exit
    bgp-ad
        vpls-id 64500:1000
        no shutdown
    exit
    stp
        shutdown
    exit
    no shutdown
```

Step 2. Add bgp-evpn and ISID-policy configuration to the B-VPLS.

After the B-VPLS is configured with the split-horizon-group, the bgp-evpn configuration and ISID-policy can be added (still in shutdown). The new configuration is shown in bold.

```
*A:PE-3>config>service>vpls# info
-----
    service-mtu 2000
    pbb
        source-bmac 00:00:00:00:00:03
    exit
    split-horizon-group "CORE" create
    exit
    bgp
        pw-template-binding 1 split-horizon-group "CORE"
        exit
    exit
    bgp-ad
        vpls-id 64500:1000
        no shutdown
    exit
    bgp-evpn
        evi 1000
        vxlan
            shutdown
```

```

exit
mpls
    split-horizon-group "CORE"
    auto-bind-tunnel
        resolution any
    exit
    shutdown
exit
exit
stp
    shutdown
exit
isis-policy
    entry 10 create
        use-def-mcast
        range 1001 to 3000
    exit
exit
no shutdown

```

Step 3. Configure `bgp-evpn mpls no shutdown` on the PE.

When the configuration is ready, the **bgp-evpn mpls** context can be **no shutdown**.

```
*A:PE-3>config>service>vpls# bgp-evpn mpls no shutdown
```

The preceding **no shutdown** triggers a route-refresh message for the EVPN family from PE-3, but no changes happen because PE-3 does not create any EVPN destinations until it imports EVPN routes from the other PEs. The three spoke-sdps to the remote PEs are still up.

Step 4. Repeat Steps 1 to 3 for the second PE.

The same Steps 1 to 3 are repeated for PE-5. When **bgp-evpn mpls no shutdown** is executed, PE-5 sends a route-refresh and gets the BGP-EVPN routes from PE-3. As a result of that, PE-3 brings down the spoke-sdp to PE-5 and creates an EVPN destination to PE-5. The same process happens in PE-5. The following CLI output shows the received routes in PE-3 and spoke-SDP going down.

```

97 2015/09/14 23:56:36.52 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.2
"Peer 1: 192.0.2.2: UPDATE
Peer 1: 192.0.2.2 - Received BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 117
    Flag: 0x90 Type: 14 Len: 47 Multiprotocol Reachable NLRI:
        Address Family EVPN
        NextHop len 4 NextHop 192.0.2.5
        Type: EVPN-Incl-mcast Len: 17 RD: 64500:1000, tag: 1001, orig_addr len:
32, orig_addr: 192.0.2.5
        Type: EVPN-Incl-mcast Len: 17 RD: 64500:1000, tag: 0, orig_addr len: 32,
orig_addr: 192.0.2.5
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100

```

PBB-VPLS to PBB-EVPN Migration

```
Flag: 0x80 Type: 9 Len: 4 Originator ID: 192.0.2.5
Flag: 0x80 Type: 10 Len: 4 Cluster ID:
1.1.1.1
Flag: 0xc0 Type: 16 Len: 16 Extended Community:
target:64500:1000
bgp-tunnel-encap:MPLS
Flag: 0xc0 Type: 22 Len: 9 PMSI:
Tunnel-type Ingress Replication (6)
Flags [Leaf not required]
MPLS Label 4194112
Tunnel-Endpoint 192.0.2.5
"

1 2015/09/14 23:56:36.52 UTC MINOR: SVCMMGR #2306 Base
"Status of SDP Bind 17405:4294967293 in service 1000 (customer 1) changed to adm
in=up oper=down flags="

*A:PE-3# show service id 1000 base
---snipped---

-----
Service Access & Destination Points
-----
Identifier                                     Type      AdmMTU  OprMTU  Adm  Opr
-----
sdp:17405:4294967293 SB(192.0.2.5)          BgpAd     0       8974    Up   Down
sdp:17406:4294967294 SB(192.0.2.4)          BgpAd     0       8974    Up   Up
sdp:17407:4294967295 SB(192.0.2.2)          BgpAd     0       8974    Up   Up
=====
* indicates that the corresponding row element may have been truncated.

*A:PE-3# show service id 1000 sdp 17405:4294967293 detail | match Flag
Flags                                     : EvpnRouteConflict

*A:PE-3# show service id 1000 evpn-mpls
=====
BGP EVPN-MPLS Dest
=====
TEP Address      Egr Label      Num. MACs      Mcast          Last Change
                  Transport
-----
192.0.2.5        262132         1              Yes            09/14/2015 23:56:37
                  ldp
-----
Number of entries : 1
-----
=====

---snipped---
```

Step 5. Repeat Steps 1 to 3 for the rest of the PEs.

The same process is repeated in all the PEs, node-by-node. The service impact for the I-VPLS 1001 is minimal.

Step 6. (Optional) Remove the ISID policy.

When all the PEs in the B-VPLS 1000 are migrated, the isid-policy can optionally be removed, node-by-node. This forces the B-VPLS instance to start using the MFIB to send I-VPLS BUM traffic to the remote nodes. Note: This has no effect on Epipes (traffic is always unicast for Epipes).

Before removing the ISID-policy and starting to use the MFIB, it is recommended to check that the Inclusive Multicast routes for an ISID to the remote PEs are all active. Otherwise, connectivity for BUM traffic could be interrupted if any of the expected routes are not active. This is illustrated below for PE-3. The routes for ISID 1001 are valid and used by BGP (flags **u*>i**). After removing the ISID-policy, the MFIB is populated with entries for the ISID 1001 group BMAC to the three remote PEs where ISID 1001 is defined.

```
*A:PE-3# show service id 1000 evpn-mpls
=====
BGP EVPN-MPLS Dest
=====
TEP Address      Egr Label      Num. MACs      Mcast          Last Change
      Transport
-----
192.0.2.2        262134         1              Yes            09/15/2015 00:13:40
                  ldp
192.0.2.4        262130         1              Yes            09/15/2015 00:14:05
                  ldp
192.0.2.5        262132         1              Yes            09/14/2015 23:56:37
                  ldp
-----
Number of entries : 3
=====

*A:PE-3# show router bgp routes evpn inclusive-mcast tag 1001
=====
BGP Router ID:192.0.2.3      AS:64500      Local AS:64500
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked, x - stale, > - best, b - backup, p - purge
Origin codes  : i - IGP, e - EGP, ? - incomplete
=====
BGP EVPN Inclusive-Mcast Routes
=====
Flag  Route Dist.      OrigAddr      NextHop
      Tag
-----
u*>i  64500:1000         192.0.2.5     192.0.2.5
      1001
u*>i  64500:1000         192.0.2.4     192.0.2.4
      1001
u*>i  64500:1000         192.0.2.2     192.0.2.2
      1001
-----
Routes : 3
```

```

=====
*A:PE-3# show service id 1000 mfib
=====
Multicast FIB, Service 1000
=====
Source Address  Group Address          Sap/Sdp Id          Svc Id  Fwd
                                           Blk
-----
Number of entries: 0
=====

*A:PE-3>config>service>vpls# isid-policy no entry 10

*A:PE-3# show service id 1000 mfib
=====
Multicast FIB, Service 1000
=====
Source Address  Group Address          Sap/Sdp Id          Svc Id  Fwd
                                           Blk
-----
*                01:1E:83:00:03:E9      b-eMpls:192.0.2.2:262134    Local    Fwd
                                           b-eMpls:192.0.2.4:262130    Local    Fwd
                                           b-eMpls:192.0.2.5:262132    Local    Fwd
-----
Number of entries: 1
=====

```

Step 7. (Optional) Remove the BGP-AD configuration.

The BGP-AD configuration can stay in the B-VPLS services. However, when the entire network is migrated to PBB-EVPN, all the spoke-sdps will be operationally down and, even if they are not forwarding traffic, they consume resources in the system. Consider removing the BGP-AD configuration and, therefore, the spoke-SDPs.

The below example shows the removal of bgp-ad in PE-4. Be aware that when bgp-ad is removed from the configuration, if the RD/RT was derived from the vpls-id (as in this example), a new RD/RT must be auto-derived for the service. Therefore, new updates will be sent for all the EVPN NLRIs, as shown in the following output.

```

*A:PE-4# show service id 1000 bgp
=====
BGP Information
=====
Vsi-Import      : None
Vsi-Export      : None
Route Dist      : None
Oper Route Dist : 64500:1000
Oper RD Type    : derivedVpls
Rte-Target Import : None
Oper RT Imp Origin : derivedVpls
Oper RT Exp Origin : derivedVpls
Rte-Target Export: None
Oper RT Import    : 64500:1000
Oper RT Export    : 64500:1000

```

```

PW-Template Id      : 1                      PW-Template SHG   : CORE
Oper Group          : None
Mon Oper Group      : None
BFD Template        : None
BFD-Enabled         : no                     BFD-Encap           : ipv4
Import Rte-Tgt      : None
-----
=====

*A:PE-4# configure service vpls 1000 bgp-ad shutdown

13 2015/09/15 00:44:06.40 UTC MINOR: SVCNMR #2306 Base
"Status of SDP Bind 17406:4294967294 in service 1000 (customer 1) changed to adm
in=down oper=down flags=sdpBindAdminDown noIngressVcLabel noEgressVcLabel "

14 2015/09/15 00:44:06.41 UTC MAJOR: SVCNMR #2320 Base
"Service Id 1000, Dynamic bgp-l2vpn SDP Bind Id 17406:4294967294 was deleted."

16 2015/09/15 00:44:06.41 UTC MAJOR: SVCNMR #2319 Base
"Dynamic bgp-l2vpn SDP 17406 (192.0.2.5) was deleted."

*A:PE-4# configure service vpls 1000 bgp no pw-template-binding 1
*A:PE-4#
*A:PE-4# configure service vpls 1000 no bgp-ad
*A:PE-4#
*A:PE-4# show service id 1000 bgp
=====
BGP Information
=====
Vsi-Import          : None
Vsi-Export          : None
Route Dist          : None
Oper Route Dist     : 192.0.2.4:1000
Oper RD Type        : derivedEvi
Rte-Target Import   : None                      Rte-Target Export: None
Oper RT Imp Origin  : derivedEvi                  Oper RT Import   : 64500:1000
Oper RT Exp Origin  : derivedEvi                  Oper RT Export   : 64500:1000
PW-Template Id      : None
-----
=====

```

In this case, the system picks up the RD in the following order:

1. Manual RD or auto-RD always take precedence when configured.
2. If no manual/auto-RD, the RD is derived from the **bgp-ad vpls-id**.
3. If no manual/auto-rd/vpls-id configuration, the RD is derived from the **bgp evpn evi**.
4. If no manual/auto-rd/vpls-id/evi configuration, there will not be RD and the service will fail.

If in the migration from BGP-AD to BGP-EVPN, the advertisement of new updates is not needed, the initial configuration must include manual/auto-RDs. If manual/auto-RDs were not included, a **bgp-ad shutdown** would not cause the change of RD and the consequent BGP updates.

PBB-EVPN Multi-Homing

This section provides configuration guidelines for PBB-EVPN multi-homing. In the same way that EVPN-MPLS supports single-active and all-active multi-homing, PBB-EVPN can also be configured to support both modes. The same **ethernet-segment** that is used for regular EVPN-MPLS service saps and spoke-sdps can be shared with I-VPLS/Epipe SAPs and spoke-SDPs.

Figure 153 shows the network setup used in this section.

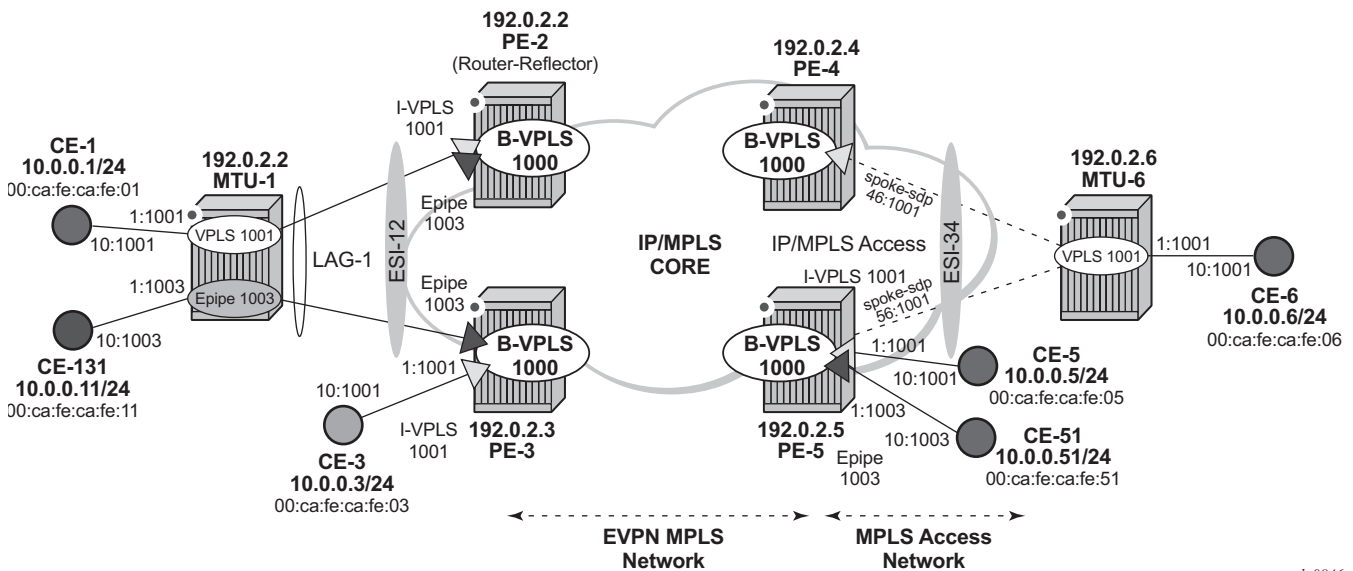


Figure 153: PBB-EVPN Multi-homing

MTU-1 and MTU-6 have been added to the network (compared to Figure 151). I-VPLS 1001 has two new sites that are multi-homed to the PBB-EVPN network. MTU-1 uses all-active multi-homing, whereas MTU-6 is connected to a single-active ES. As with EVPN-MPLS, all-active multi-homing is only supported when a LAG is used at the access. Single-active multi-homing can be supported with regular Ethernet ports (that can form an independent LAG per PE) or SDPs.

Draft-ietf-l2vpn-pbb-evpn describes two types of system BMAC assignments that a PE can implement in a B-VPLS when ESs are present:

- Shared BMAC addresses that can be used for all the single-homed CEs and a number of multi-homed CEs connected to **ethernet-segments**.
- Dedicated BMAC addresses per **ethernet-segment**.

In this chapter and in SR OS terminology:

- A shared-bmac (in IETF) is a **source-bmac** as configured in **service>(b)vpls>pbb>source-bmac**. All the I-VPLS/Epipe traffic coming from single-homed CEs is sent encapsulated in a PBB packet with that **source-bmac**.
- A dedicated-bmac per ES (in IETF) is an **es-bmac** as activated in **service>(b)vpls>pbb>use-es-bmac** and generated from the combination of **vpls>pbb>source-bmac** plus **ethernet-segment>source-bmac-lsb**. If configured, any I-VPLS/Epipe traffic coming from an ES is encapsulated in a PBB packet with the es-bmac as the source BMAC.

The system allows the following user choices per B-VPLS and ES:

- A dedicated **es-bmac** per ES can be used. In that case, the **pbb>use-es-bmac** command is configured in the B-VPLS. In all-active multi-homing, all the PEs that are part of the ES source the PBB packets with the same **es-bmac**; single-active multi-homing requires the use of a different **es-bmac** per PE.
- A non-dedicated **source-bmac** can be used (this is only possible in single-active multi-homing). In this case, the user does not configure **pbb>use-es-bmac** and the regular **source-bmac** is used for the traffic. A different **source-bmac** has to be advertised per PE.

As discussed, single-active multi-homing can use **source-bmacs** or **es-bmacs**. Using one type or another has a different impact on CMAC flushing, as illustrated in [Figure 154](#).

- If **es-bmacs** are used, as shown on the right-hand side of [Figure 154](#), a less-impacting CMAC flush is achieved, therefore minimizing the flooding after ES failures. In the case of ES failure, PE-1 withdraws the **es-bmac** 00:12 and the remote PE-3 only flushes the CMACs associated with that **es-bmac** (only the CMACs behind the CE are flushed).
- If **source-bmacs** are used, as shown on the left-hand side of [Figure 154](#), in the case of ES failure, a BGP update with higher sequence number is issued by PE-1 and the remote PE-3 flushes all the CMACs associated with the **source-bmac**. Therefore, all the CMACs behind the B-VPLS of the PEs will be flushed, as opposed to only the CMACs behind the CE of the Ethernet Service Instances (ESIs).

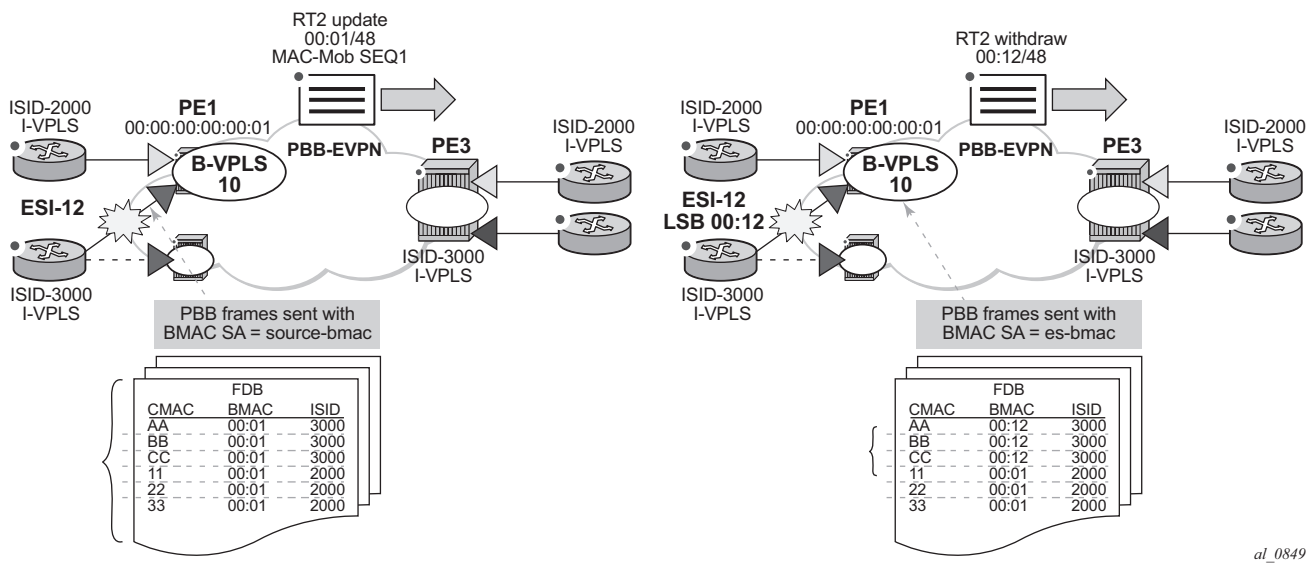


Figure 154: The Use of es-bmac to Minimize CMAC Flush

Table 9 shows the PBB-EVPN multi-homing combinations supported in the current release in the topology of Figure 4.

Table 9: PBB-EVPN Multi-Homing Supported Combinations in SR OS

CE Connectiv-ity	PE Connectiv-ity	PE Redun-dancy	BMAC Assign-ment	I-VPLS Support	Epip- Support
LAG (LACP optional)	LAG SAP	EVPN MH all-active	use-es-bmac (shared BMAC)	Yes	Yes
Ethernet ports (no LAG)	LAG SAP or port SAP	EVPN MH single-active	use-es-bmac (dedicated per PE)	Yes	No
Ethernet ports (no LAG)	LAG SAP or port SAP	EVPN MH single-active	source-bmac (dedicated per PE)	Yes	No
MPLS	spoke-SDP	EVPN MH single-active	source-bmac (dedicated per PE)	Yes	No
MPLS	spoke-SDP	EVPN MH single-active	use-es-bmac (dedicated per PE)	Yes	No

As an example, the configurations of the first, and last two, rows (lag sap all-active, mpls source-bmac, and mpls use-es-bmac, respectively) will be discussed in the following three sections.

PBB-EVPN all-active multi-homing for I-VPLS and Epipe

Figure 153 shows a PBB-EVPN network where ESI-12 is configured as an all-active multi-homing ES on PE-2 and PE-3. Two services are using ESI-12: I-VPLS 1001 and Epipe 1003. The following output shows the relevant configuration in PE-2 and PE-3.

```
*A:PE-2>config>service# info
-----
      pbb
        mac-name "PE-5" 00:00:00:00:00:05
      exit
---snipped---
      system
        bgp-evpn
          ethernet-segment "ESI-12" create
            esi 01:00:00:00:00:12:00:00:00:01
            source-bmac-lsb 12-12 es-bmac-table-size 8
            es-activation-timer 3
            service-carving
              mode auto
            exit
            multi-homing all-active
            lag 1
            no shutdown
          exit
        exit
      exit

vpls 1000 customer 1 b-vpls create
  service-mtu 2000
  pbb
    source-bmac 00:00:00:00:00:02
    use-es-bmac
  exit
  split-horizon-group "CORE" create
  exit
  bgp
  exit
  bgp-evpn
    evi 1000
    vxlan
      shutdown
    exit
    mpls
      split-horizon-group "CORE"
      ecmp 2
      auto-bind-tunnel
      resolution any
    exit
    no shutdown
  exit
  exit
  stp
    shutdown
  exit
  no shutdown
exit
```

PBB-EVPN all-active multi-homing for I-VPLS and Epipes

```
vpls 1001 customer 1 i-vpls create
  pbb
    backbone-vpls 1000
    exit
  exit
  stp
    shutdown
  exit
  sap lag-1:1001 create
  exit
  no shutdown
exit
epipe 1003 customer 1 create
  pbb
    tunnel 1000 backbone-dest-mac "PE-5" isid 1003
  exit
  sap lag-1:1003 create
  exit
  no shutdown
exit
-----
A:PE-3>config>service# info
-----
  pbb
    mac-name "PE-5" 00:00:00:00:00:05
  exit
---snipped---
system
  bgp-evpn
    ethernet-segment "ESI-12" create
      esi 01:00:00:00:00:12:00:00:00:01
      source-bmac-lsb 12-12 es-bmac-table-size 8
      es-activation-timer 3
      service-carving
        mode auto
      exit
      multi-homing all-active
      lag 1
      no shutdown
    exit
  exit
exit

vpls 1000 customer 1 b-vpls create
  service-mtu 2000
  pbb
    source-bmac 00:00:00:00:00:03
    use-es-bmac
  exit
  split-horizon-group "CORE" create
  exit
  bgp
  exit
  bgp-evpn
    evi 1000
    vxlan
      shutdown
    exit
```

```

mpls
    split-horizon-group "CORE"
    ecmp 2
    auto-bind-tunnel
        resolution any
    exit
    no shutdown
exit
exit
stp
    shutdown
exit
no shutdown
exit
vpls 1001 customer 1 i-vpls create
    pbb
        backbone-vpls 1000
        exit
    exit
    stp
        shutdown
    exit
    sap 1/1/1:1001 create
    exit
    sap lag-1:1001 create
    exit
    no shutdown
exit

epipe 1003 customer 1 create
    pbb
        tunnel 1000 backbone-dest-mac "PE-5" isid 1003
    exit
    sap lag-1:1003 create
    exit
    no shutdown
exit

```

The preceding configuration shows that Epipe 1003 has a PBB tunnel pointing at the PE-5 source-bmac. Epipe 1003 has the following configuration in PE-5 (the PBB tunnel points at the ESI-12 es-bmac):

```

A:PE-5>config>service>pbb# info
-----
    mac-name "ES-MAC-12" 00:00:00:00:12:12
-----
A:PE-5>config>service>pbb# /configure service epipe 1003
A:PE-5>config>service>epipe# info
-----
    pbb
        tunnel 1000 backbone-dest-mac "ES-MAC-12" isid 1003
    exit
    sap 1/1/1:1003 create
    exit
    no shutdown
-----

```

Source-bmacs and es-bmacs are distributed in BGP-EVPN. PE-2 and PE-3 will each advertise their own source-bmac in a MAC route with ESI-0 and the shared es-bmac with ESI-MAX (as per the RFC 7623). The es-bmac that each PE uses in a B-VPLS is derived from the configured **service>(b)vpls>pbb>source-bmac** (four high-order bytes) and the ESI-12 configured source-bmac-lsb. In this example, PE-2 and PE-3 will both derive es-bmac 00:00:00:00:12:12. Note: For both PEs to derive the required same es-bmac, the four high-order bytes of the source-bmac must match on both PEs.

The **es-bmac-table-size** parameter modifies the default value (8) for the maximum number of es-bmacs that can be associated with the ethernet-segment across different B-VPLS services. When **source-bmac-lsb** is configured, the associated **es-bmac-table-size** is reserved out of the total FDB space.

The following outputs show the source-bmacs and es-bmac and how they are advertised and installed in the B-VPLS FDB. Note: A PE does not show its own system BMACs in the FDB.

```
*A:PE-2# show service system bgp-evpn ethernet-segment name "ESI-12" | match BMAC
Source BMAC LSB          : 12-12
```

```
*A:PE-2# show service id 1000 base
=====
Service Basic Information
=====
Service Id       : 1000                Vpn Id       : 0
Service Type     : b-VPLS
---snipped---
Oper Backbone Src : 00:00:00:00:00:02
Use SAP B-MAC    : Disabled
i-Vpls Count     : 1
Epipe Count      : 1
Use ESI B-MAC    : Enabled

--snipped--
```

```
A:PE-3# show service system bgp-evpn ethernet-segment name "ESI-12" | match BMAC
Source BMAC LSB          : 12-12
```

```
A:PE-3# show service id 1000 base
=====
Service Basic Information
=====
Service Id       : 1000                Vpn Id       : 0
Service Type     : b-VPLS
---snipped---
Oper Backbone Src : 00:00:00:00:00:03
Use SAP B-MAC    : Disabled
i-Vpls Count     : 1
Epipe Count      : 2
Use ESI B-MAC    : Enabled
```

```
*A:PE-2# show service id 1000 fdb detail
```

```

=====
Forwarding Database, Service 1000
=====
ServId      MAC                Source-Identifier      Type      Last Change
              Age
-----
1000        00:00:00:00:00:03  eMpls:                EvpnS     09/15/15 23:03:03
              192.0.2.3:262136
1000        00:00:00:00:00:04  eMpls:                EvpnS     09/15/15 22:05:20
              192.0.2.4:262133
1000        00:00:00:00:00:05  eMpls:                EvpnS     09/15/15 22:04:35
              192.0.2.5:262142
-----
No. of MAC Entries: 3
-----
Legend:  L=Learned O=Oam P=Protected-MAC C=Conditional S=Static
=====

```

A:PE-4# show service id 1000 fdb detail

```

=====
Forwarding Database, Service 1000
=====
ServId      MAC                Source-Identifier      Type      Last Change
              Age
-----
1000        00:00:00:00:00:02  eMpls:                EvpnS     09/15/15 22:13:01
              192.0.2.2:262134
1000        00:00:00:00:00:03  eMpls:                EvpnS     09/15/15 23:10:42
              192.0.2.3:262136
1000        00:00:00:00:00:05  eMpls:                EvpnS     09/15/15 22:13:01
              192.0.2.5:262142
1000        00:00:00:00:12:12  eES:                  EvpnS     09/15/15 22:13:01
              MAX-ESI
-----
No. of MAC Entries: 4
-----
Legend:  L=Learned O=Oam P=Protected-MAC C=Conditional S=Static
=====

```

A:PE-4# show router bgp routes evpn mac mac-address 00:00:00:00:12:12

```

=====
BGP Router ID:192.0.2.4      AS:64500      Local AS:64500
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
              l - leaked, x - stale, > - best, b - backup, p - purge
Origin codes : i - IGP, e - EGP, ? - incomplete
=====
BGP EVPN MAC Routes
=====
Flag  Route Dist.      MacAddr      ESI
      Tag           Mac Mobility  Ip Address
                        NextHop
                        Label1
-----
u*>i  192.0.2.2:1000    00:00:00:00:12:12  ESI-MAX
      0              Static          N/A
                        192.0.2.2
                        LABEL 262134

```

Designated Forwarder (DF) Election

```
u*>i 192.0.2.3:1000      00:00:00:00:12:12 ESI-MAX
      0                  Static      N/A
                                   192.0.2.3
                                   LABEL 262136
```

```
-----
Routes : 2
=====
```

PBB-EVPN all-active multi-homing is based on the same concepts as EVPN-MPLS all-active multi-homing: DF election, split-horizon, and aliasing.

Designated Forwarder (DF) Election

- Only the DF PE for an ISID will send multicast traffic to the ES. The DF PE for an ISID can be shown with the following command:

```
A:PE-3# show service system bgp-evpn ethernet-segment name "ESI-12" isid 1003
=====
ISID DF and Candidate List
=====
Isid          SvcId          Actv Timer Rem      DF  DF Last Change
-----
1003          1003           0                   yes 09/15/2015 23:02:48
=====
DF Candidates                                Time Added
-----
192.0.2.2                                09/15/2015 23:03:02
192.0.2.3                                09/15/2015 23:02:45
-----
Number of entries: 2
=====
```

- The DF PE identifies multicast traffic by looking at either the destination BMAC or the EVPN label (which can be unicast or multicast).
- In the case of Epipes, there are also DF and non-DF PEs. However, traffic is usually unicast (sent to the PBB tunnel backbone-destination-bmac). The non-DF PE will not usually discard Epipe traffic to the ES, unless the packet comes with an EVPN multicast label. To avoid packet duplication at the CE for Epipes, it is recommended to either:
 - configure **discard-unknown** on all the B-VPLS instances where there are PBB-Epipes. This will prevent the ingress PE from flooding Epipe traffic if the PBB tunnel BMAC is unknown in the FDB.
 - configure **ingress-replication-bum-label** so that, when the PBB tunnel BMAC is unknown in the FDB, the ingress PE sends traffic with a multicast label. The non-DF will discard traffic identified as multicast at Epipes.

Ethernet-Segment Split-horizon

In PBB-EVPN all-active multi-homing, the split-horizon function is not based in the ESI label but in a source BMAC check. When BUM traffic is received on an I-VPLS, the PE will encapsulate it in PBB using the **es-bmac** as source BMAC and the group BMAC for the ISID. When the DF PE for the ISID receives that packet, it will not send it back to the ES if the packet is identified as being originated from the ES itself (based on the **es-bmac** shared between the PEs).

Aliasing

Aliasing is based on the advertisement of the same **es-bmac** with MAX-ESI from the PEs part of the same ES. PE-2 and PE-3 advertise the **es-bmac 00:00:00:00:12:12** with MAX-ESI (ESI = all FFs, as per the RFC 7623) and as Static (protected). When the remote PEs, PE-4, and PE-5, receive the two routes for the same BMAC and MAX-ESI, they will create a single EVPN-MPLS destination that will give more than one next-hop (in this case 2), as long as ecmp > 1:

```
A:PE-4# show service id 1000 evpn-mpls
=====
BGP EVPN-MPLS Dest
=====
TEP Address      Egr Label      Num. MACs      Mcast          Last Change
      Transport
-----
192.0.2.2        262134         1              Yes            09/15/2015 22:13:01
                  ldp
192.0.2.3        262136         1              Yes            09/15/2015 23:10:42
                  ldp
192.0.2.5        262142         1              Yes            09/15/2015 22:13:01
                  ldp
-----
Number of entries : 3
-----
=====
BGP EVPN-MPLS Ethernet Segment Dest
=====
Eth SegId              Num. Macs          Last Change
-----
No Matching Entries
=====
BGP EVPN-MPLS ES BMAC Dest
=====
ES BMAC Addr              Last Change
-----
00:00:00:00:12:12        09/15/2015 23:10:42
-----
Number of entries: 1
-----
=====

A:PE-4# show service id 1000 evpn-mpls es-bmac 00:00:00:00:12:12
```

Ethernet-Segment Split-horizon

```

=====
BGP EVPN-MPLS ES BMAC Dest
=====
ES BMAC Addr                               Last Change
-----
00:00:00:00:12:12                          09/15/2015 23:10:42
=====

BGP EVPN-MPLS ES BMAC Dest TEP Info
=====
TEP Address          Egr Label          Last Change
                    Transport
-----
192.0.2.2            262134          09/15/2015 22:13:01
                    ldp
192.0.2.3            262136          09/15/2015 23:10:42
                    ldp
-----
Number of entries : 2
=====

```

A similar output will be obtained in PE-5. Unicast traffic entering I-VPLS 1001 in either PE-4 or PE-5 will be hashed and load balanced to PE-2 and PE-3 if the destination CMAC lookup yields an **es-bmac-dest**:

```

A:PE-5# show service id 1001 fdb detail pbb
=====
Forwarding Database, i-Vpls Service 1001
=====
MAC                Source-Identifier    B-Svc    b-Vpls MAC        Type/Age
-----
00:ca:fe:ca:fe:01 eES-BMAC:          1000      00:00:00:00:12:12 L/21
                  00:00:00:00:12:12
00:ca:fe:ca:fe:03 b-eMpls:           1000      00:00:00:00:00:03 L/0
                  192.0.2.3:262136
00:ca:fe:ca:fe:05 sap:1/1/1:1001      1000      N/A              L/0
00:ca:fe:ca:fe:06 sdp:56:1001        1000      N/A              L/0
=====

A:PE-5# show service id 1001 fdb evpn-mpls es-bmac-dest 00:00:00:00:12:12
=====
Forwarding Database, Service 1001
=====
ServId    MAC                Source-Identifier    Type        Last Change
                    Age
-----
1001      00:ca:fe:ca:fe:01 eES-BMAC:          L/51        09/16/15 00:30:30
                  00:00:00:00:12:12
-----
No. of Entries: 1
-----
Legend:  L=Learned O=Oam P=Protected-MAC C=Conditional S=Static
=====

```

If a failure occurs in the ES, the PE will withdraw the **es-bmac** and the remote PEs will remove one next-hop of the ES-BMAC EVPN-MPLS destination.

For PBB-Epipes, aliasing will also work, as long as shared-queuing or policing are enabled on the ingress PE Epipe. In [Figure 153](#), Epipe 1003 on PE-5 requires shared-queuing or policing at the ingress SAP. Otherwise, the traffic will be sent to only one PE of the ES (usually to the lower-IP PE).

For more information about the configuration of the **ethernet-segment** and its parameters, see the [EVPN for MPLS Tunnels on page 937](#) chapter.

PBB-EVPN Single-Active Multi-Homing for I-VPLS with source-bmacs

ESI-34 is a single-active **ethernet-segment** (see [Figure 153](#)) with SDPs linked to it. As indicated in [Table 9](#), only I-VPLS services can be used in this configuration. As discussed in section [PBB-EVPN Multi-Homing on page 1012](#) single-active ES and B-VPLS services can be configured to either use **source-bmacs** or **es-bmacs**. The following configuration shows the former option:

```
A:PE-4>config>service# info
-----
    sdp 46 mpls create
        far-end 192.0.2.6
        ldp
        keep-alive
        shutdown
        exit
        no shutdown
    exit
---snipped---
    system
        bgp-evpn
            ethernet-segment "ESI-34" create
                esi 01:00:00:00:00:34:00:00:00:01
                source-bmac-lsb 34-04 es-bmac-table-size 8
                es-activation-timer 3
                service-carving
                    mode auto
                exit
                multi-homing single-active
                sdp 46
                no shutdown
---snipped---
    vpls 1000 customer 1 b-vpls create
        service-mtu 2000
        pbb
            source-bmac 00:00:00:00:00:04
        exit
        split-horizon-group "CORE" create
        exit
        bgp
```

PBB-EVPN Single-Active Multi-Homing for I-VPLS with source-bmacs

```
exit
bgp-evpn
  evi 1000
  vxlan
    shutdown
  exit
  mpls
    split-horizon-group "CORE"
    ecmp 2
    auto-bind-tunnel
      resolution any
    exit
    no shutdown
  exit
exit
stp
  shutdown
exit
no shutdown
exit
vpls 1001 customer 1 i-vpls create
  pbb
    backbone-vpls 1000
    exit
  exit
  stp
    shutdown
  exit
  spoke-sdp 46:1001 create
    no shutdown
  exit
  no shutdown
exit
-----
A:PE-5>config>service# info
-----
  sdp 56 mpls create
    far-end 192.0.2.6
    ldp
    keep-alive
    shutdown
  exit
  no shutdown
exit
---snipped---
system
  bgp-evpn
    ethernet-segment "ESI-34" create
      esi 01:00:00:00:00:34:00:00:00:01
      source-bmac-lsb 34-05 es-bmac-table-size 8
      es-activation-timer 3
      service-carving
        mode auto
      exit
      multi-homing single-active
      sdp 56
      no shutdown
---snipped---
```

```

vpls 1000 customer 1 b-vpls create
  service-mtu 2000
  pbb
    source-bmac 00:00:00:00:00:05
  exit
  split-horizon-group "CORE" create
  exit
  bgp
  exit
  bgp-evpn
    evi 1000
    vxlan
      shutdown
    exit
    mpls
      split-horizon-group "CORE"
      ecmp 2
      auto-bind-tunnel
        resolution any
      exit
      no shutdown
    exit
  exit
  stp
    shutdown
  exit
  no shutdown
exit
vpls 1001 customer 1 i-vpls create
  pbb
    backbone-vpls 1000
    exit
  exit
  stp
    shutdown
  exit
  sap 1/1/1:1001 create
  exit
  spoke-sdp 56:1001 create
    no shutdown
  exit
  no shutdown
exit

```

With the preceding configuration, PE-4 and PE-5 will not advertise es-bmacs with MAX-ESI. Therefore, all the remote BMACs on PE-2/PE-3 are associated with regular backbone EVPN-MPLS destinations. The CMACs will be learned in the data plane associated with local sap/sdp-bindings or remote BMACs. An example for the I-VPLS and B-VPLS FDB in PE-2 follows:

```

A:PE-2# show service id 1001 fdb detail pbb
=====
Forwarding Database, i-Vpls Service 1001
=====

```

MAC	Source-Identifier	B-Svc	b-Vpls MAC	Type/Age
00:ca:fe:ca:fe:01	sap:lag-1:1001	1000	N/A	L/206

PBB-EVPN Single-Active Multi-Homing for I-VPLS with source-bmacs

```

00:ca:fe:ca:fe:05 b-eMpls:          1000      00:00:00:00:00:05 L/309
          192.0.2.5:262142
00:ca:fe:ca:fe:06 b-eMpls:          1000      00:00:00:00:00:05 L/180
          192.0.2.5:262142
=====

A:PE-2# show service id 1000 fdb detail
=====
Forwarding Database, Service 1000
=====

```

ServId	MAC	Source-Identifier	Type Age	Last Change
1000	00:00:00:00:00:03	eMpls: 192.0.2.3:262136	EvpnS	09/15/15 23:03:03
1000	00:00:00:00:00:04	eMpls: 192.0.2.4:262133	EvpnS	09/15/15 22:05:20
1000	00:00:00:00:00:05	eMpls: 192.0.2.5:262142	EvpnS	09/15/15 22:04:35

```

-----
No. of MAC Entries: 3
-----
Legend:  L=Learned O=Oam P=Protected-MAC C=Conditional S=Static
=====

```

In the preceding example, the DF for ISID 1001 is PE-5. With a failure event on the SDP to MTU-6, PE-5 will not withdraw the advertised **source-bmac** (because it is still being used as source-bmac for other services and even CEs within the same service). PE-5 will send an update of the same source-bmac instead, increasing the sequence number in the MAC mobility extended community. That will be a **flush-all-from-me** indication for the remote PEs (they will flush all the CMACs associated with the updated source-bmac, irrespective of the service).

When the former DF (PE-5) comes back up, PE-4 will become non-DF and will send a CMAC flush indication using the same mechanism as discussed above.

The following example shows a failure of SDP 56 in PE-5 and the corresponding DF switchover and CMAC flush.

```

*A:PE-5#
18 2015/09/16 01:47:07.86 UTC MINOR: SVCMGR #2303 Base
"Status of SDP 56 changed to admin=up oper=down"

20 2015/09/16 01:47:07.86 UTC MINOR: SVCMGR #2095 Base
"Ethernet Segment:ESI-34, ISID:1001, Designated Forwarding state changed to:false"

4 2015/09/16 01:47:07.86 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.2
"Peer 1: 192.0.2.2: UPDATE
Peer 1: 192.0.2.2 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 96
  Flag: 0x90 Type: 14 Len: 44 Multiprotocol Reachable NLRI:
    Address Family EVPN
    NextHop len 4 NextHop 192.0.2.5
    Type: EVPN-MAC Len: 33 RD: 192.0.2.5:1000 ESI: ESI-0, tag: 0, mac len: 4
  8 mac: 00:00:00:00:00:05, IP len: 0, IP: NULL, label1: 4194272

```

```

Flag: 0x40 Type: 1 Len: 1 Origin: 0
Flag: 0x40 Type: 2 Len: 0 AS Path:
Flag: 0x80 Type: 4 Len: 4 MED: 0
Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
Flag: 0xc0 Type: 16 Len: 24 Extended Community:
    target:64500:1000
    bgp-tunnel-encap:MPLS
    mac-mobility:Seq:1/Static

```

Note: Individual SAP or spoke SDP failures do not trigger any MAC flush or DF re-election. This is as per RFC 7623. In EVPN-MPLS, individual Sap/spoke SDP failures are captured by the AD per-EVI withdrawal, which triggers a DF switchover.

PBB-EVPN Single-Active Multi-Homing for I-VPLS with es-bmacs

As discussed throughout this document, the use of **es-bmacs** for single-active multi-homing can minimize the number of CMACs flushed in a network. A simple change is necessary: activate the **use-es-bmac** command and ensure that the generated es-bmacs in PE-4 and PE-5 are different (the **source-bmac-lsb** in the previous configuration had different values for PE-4 and PE-5 already):

```

A:PE-4# configure service vpls 1000 pbb use-es-bmac
A:PE-5# configure service vpls 1000 pbb use-es-bmac

```

```

A:PE-4# show service system bgp-evpn ethernet-segment name "ESI-34" | match BMAC
Source BMAC LSB          : 34-04

```

```

A:PE-5# show service system bgp-evpn ethernet-segment name "ESI-34" | match BMAC
Source BMAC LSB          : 34-05

```

```

A:PE-2# show service id 1000 fdb detail
=====
Forwarding Database, Service 1000
=====

```

ServId	MAC	Source-Identifier	Type Age	Last Change
1000	00:00:00:00:00:03	eMpls: 192.0.2.3:262136	EvpnS	09/15/15 23:03:03
1000	00:00:00:00:00:04	eMpls: 192.0.2.4:262133	EvpnS	09/15/15 22:05:20
1000	00:00:00:00:00:05	eMpls: 192.0.2.5:262142	EvpnS	09/15/15 22:04:35
1000	00:00:00:00:34:04	eMpls: 192.0.2.4:262133	EvpnS	09/16/15 01:52:26
1000	00:00:00:00:34:05	eMpls: 192.0.2.5:262142	EvpnS	09/16/15 01:56:29

```

-----
No. of MAC Entries: 5
-----
Legend:  L=Learned O=Oam P=Protected-MAC C=Conditional S=Static
=====

```

The remote PEs (such as PE-2 in the preceding output) will receive two more BMACs in BGP. However, the benefit is that in case of a failure in ESI-34 (as before) the es-bmac is withdrawn and the remote PEs will only flush the CMACs associated with the remote ES-34, as opposed to all the CMACs associated with PE-5.

PBB-EVPN Single-Active Multi-Homing for Epipes

In the network in [Figure 153](#), Epipes can only support single-homing or all-active multi-homing but not single-active. Single-active multi-homing for Epipes is only supported in the following scenarios.

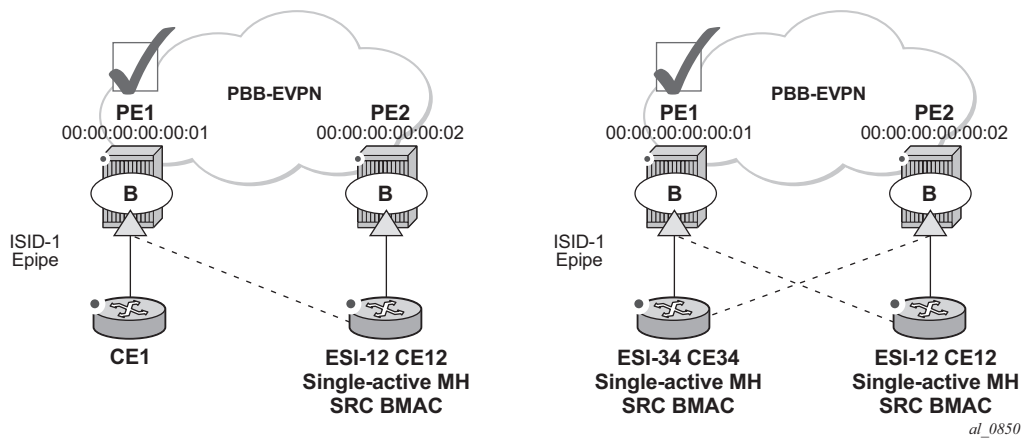


Figure 155: PBB-EVPN Single-Active Support for Epipes

Single-active multi-homing is supported for redundancy in a two-node, three or four SAP, scenario, as displayed in Figure 6. In these two cases, the Epipe PBB tunnel will be configured with the source BMAC of the remote PE node. When two SAPs are active in the same Epipe, local-switching is used to exchange frames between the CEs.

All-active multi-homing is not supported for redundancy in this scenario because the PE-1 PBB tunnel cannot point at a locally defined ES-BMAC.

PBB-EVPN Multi-Homing Operation

See the [EVPN for MPLS Tunnels on page 937](#) chapter for the commands to operate **ethernet-segments**. Consider that there are no AD routes in PBB-EVPN. Also, the DF election algorithm will be based on the ISID values as opposed to EVIs.

Troubleshooting and Debug Commands

When troubleshooting PBB-EVPN networks, most of the troubleshooting commands discussed in [EVPN for MPLS Tunnels on page 937](#) can be used in the B-VPLS service and the base **service>system>bgp-evpn** instance. Some examples of useful commands are:

- show redundancy bgp-evpn-multi-homing
- show router bgp routes evpn (and filters)
- show service evpn-mpls [<TEP ip-address>]
- show service id bgp-evpn
- show service id evpn-mpls (and modifiers)
- show service id fdb pbb (and modifiers)
- show service system bgp-evpn
- show service system bgp-evpn ethernet-segment (and modifiers)
- debug router bgp update
- log-id 99

In addition, the following **tools dump** commands also discussed in [EVPN for MPLS Tunnels on page 937](#) can help too:

- tools dump service evpn usage
- tools dump service system bgp-evpn ethernet-segment <name> isid df (Note: **isid** is used instead of **evi**.)

There are two aspects that are specific to PBB-EVPN and not EVPN:

1. Consumption of virtual BMACs in the system, source-bmacs, sap-bmacs, sdp-bmacs, and es-bmacs are system BMACs that use FDB space but are not shown in the FDB together with the rest of the learned MACs. The following command provides information about the virtual system MACs consumed in the system.

```
A:PE-3# tools dump redundancy src-bmac-lsb
Src-bmac-lsb:      3 (00-03) User: B-Vpls - 1 service(s)
Src-bmac-lsb:  4626 (12-12) User: ES

Total Src-bmac-lsbs = 2
```

2. Consumption of MFIBs — when ISIDs are not using the default-multicast list in the B-VPLS context for sending BUM traffic, an MFIB is consumed per ISID. The following command provides information about the consumption of MFIBs per system and per B-VPLS.

```
A:PE-2# tools dump service vpls-pbb-mfib-stats  
detail
```

```
A:PE-2# tools dump service vpls-pbb-mfib-stats detail
```

```
Service Manager VPLS PBB MFIB statistics at 000 05:06:39.170:
```

Usage per Service

ServiceId	MFIB User	Count
-----+-----+-----		
1000	Evpn	1
-----+-----+-----		
	Total	1

MMRP

```
Current Usage      :      0  
System Limit       : 8191 Full, 40959 EOnly  
Per Service Limit  : 2048 Full, 8192 EOnly
```

SPB

```
Current Usage      :      0  
System Limit       : 8191  
Per Service Limit  : 8191
```

Evpn

```
Current Usage      :      1  
System Limit       : 40959  
Per Service Limit  : 8191
```

Conclusion

In addition to a full RFC 7432 EVPN-MPLS implementation, SR OS supports PBB-EVPN as per RFC 7623 for large Layer 2 deployments, including single-active and all-active multi-homing. This example has shown how to configure and operate a PBB-EVPN network focusing on the specific aspects of PBB-EVPN compared to EVPN-MPLS.

EVPN for VXLAN Tunnels (Layer 2)

In This Chapter

This section provides information about Layer 2 and EVPN.

Topics in this section include:

- [Applicability on page 1034](#)
- [Overview on page 1035](#)
- [Configuration on page 1037](#)
- [Conclusion on page 1062](#)

Applicability

This example is applicable to the 7950 XRS, 7750 SR-c4/c12, 7750 SR-7/12 and 7450 ESS-6/6v/7/12, but it is not supported on the 7750 SR-1, 7450 ESS-1 or 7710 SR. Virtual eXtensible Local Area Network (VXLAN) requires IOM3-XP/IMM or higher-based line cards and chassis-mode D. Ethernet Virtual Private Networks (EVPN) is a control plane technology and does not have line card hardware dependencies.

The configuration was tested in release 12.0.R4.

Overview

SR OS supports the EVPN control plane with Virtual eXtensible Local Area Network (VXLAN) data plane in VPLS services.

EVPN is an IETF technology (draft-ietf-l2vpn-evpn) that uses a new BGP address family which allows VPLS services to be operated in a similar way to IP-VPNs, where the MAC addresses, IP addresses and the information to set up the flooding tree are distributed by BGP. While EVPN can be used as the control plane for different data plane encapsulations, only VXLAN is supported in SR OS in the release tested.

VXLAN (draft-mahalingam-dutt-dcops-vxlan) is an overlay IP tunneling technology used to carry Ethernet traffic over any IP network, and it is becoming the de-facto standard for overlay data centers and networks. Compared to other IP overlay tunneling technologies, such as GRE, VXLAN supports multi-tenancy and multi-pathing:

- A tenant identifier, the VXLAN Network Identifier (VNI), is encoded in the VXLAN header and allows each tenant to have an isolated Layer 2 domain.
- VXLAN supports multi-pathing scalability through ECMP. VXLAN uses the outer source UDP port as an entropy field that can be used by the core IP routers to balance the load across different paths.

In SR OS, EVPN and VXLAN can be enabled in VPLS or R-VPLS services. In this example, EVPN-VXLAN services will refer to VPLS or R-VPLS services with EVPN and VXLAN enabled. These services can terminate/originate VXLAN tunnels and may have SAPs and/or SDP bindings at the same time. Some other SR OS implementation-specific considerations are listed below:

- VXLAN is only supported on network or hybrid ports with null or dot1q encapsulation on Ethernet/LAG/POS/APS interfaces.
- VXLAN packets are originated/terminated with the system IPv4 address, in other words, a system originating VXLAN packets will use the system IP address as source outer IPv4 address and systems will only process VXLAN packets if their destination outer IPv4 address matches its own system IP address.
- Data plane MAC learning is not supported over VXLAN bindings. Only the control plane (EVPN) will be used for populating the FDB with MAC addresses associated to VXLAN bindings.

- EVPN provides support for the following features that are tested in this document:
 - The BGP advertisement of the MAC addresses learned on SAPs, SDP-bindings and conditional static MACs to the remote BGP peers. The advertisement of MAC addresses in BGP can optionally be disabled.
 - The optional advertisement of an unknown MAC route, that allows the remote EVPN PE or Network Virtualization Edge devices (NVEs) to suppress the unknown unicast flooding and send any unknown unicast frame to the owner of the unknown MAC route.
 - Ingress replication of Broadcast, Unknown unicast and Multicast (BUM) packets over VXLAN.
 - A Proxy-ARP table per service populated by the MAC-IP pairs received in BGP MAC advertisements. When an ARP request is received on a SAP or SDP-binding, the system will perform a lookup on this table and will reply to the ARP request if the lookup yields a valid result.
 - MAC mobility and static-mac protection as described in draft-ietf-l2vpn-evpn, as well as MAC duplication detection.
- Multi-homing redundancy for SAPs and SDP-bindings in EVPN-VXLAN services is supported through BGP Multi-homing (L2VPN BGP address family). Only one BGP-MH site is supported in an EVPN-VXLAN service.

One of the main applications for EVPN-VXLAN services in SR OS is the Data Center Gateway (DC GW) function. In such an application, EVPN and VXLAN are expected to be used within the Data Center and VPLS SDP-bindings or SAPs are expected to be used for the connectivity to the WAN. When the system is used as a DC GW a VPLS service is configured per Layer 2 domain that has to be extended to the WAN. In those VPLS services, BGP EVPN automatically sets up the VXLAN auto-bindings that connect the DC GW to the Data Center NVEs. The WAN connectivity is based on regular VPLS constructs where SAPs (null, dot1q and QinQ), spoke-SDPs (FEC type 128 and 129, not BGP-VPLS) and mesh-SDPs are supported. B-VPLS or I-VPLS services are not supported.

Although the DC GW application is one of the most common uses for this feature, this example focuses on the configuration and operation of EVPN-VXLAN for Layer 2 services in general, and its integration with regular VPLS services in MPLS networks.

Configuration

This section describes the configuration of EVPN-VXLAN on the 7x50 as well as the available troubleshooting and show commands. This example focuses on the following configuration aspects:

- Enabling EVPN and VXLAN in a VPLS service, including the use of BGP-EVPN, BGP-AD (BGP Auto-discovery) and BGP-MH (BGP Multi-homing) in the same VPLS instance.
- Scaling BGP-MH resiliency with the use of operational groups (oper-groups).
- Use of proxy-arp in EVPN-VXLAN services
- MAC mobility, MAC duplication and MAC protection in EVPN-VXLAN services.

The configuration will be shown for PE-71, PE-69 and PE-72 only; the PEs in Overlay-Network-2 (Figure 156) have an equivalent configuration.

Enabling EVPN-VXLAN in a VPLS Service

Figure 156 shows the topology used in this example.

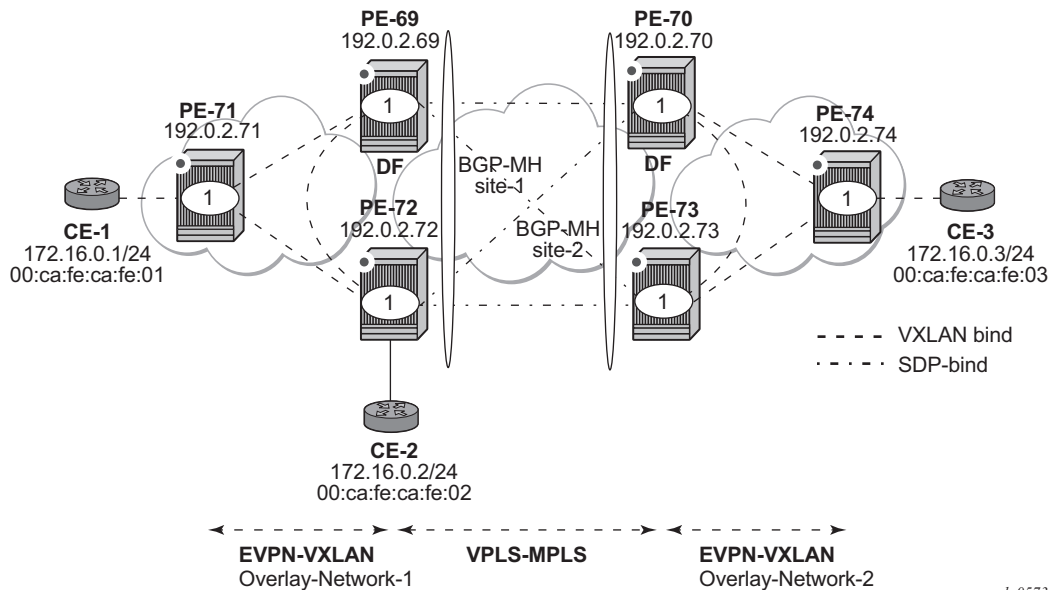


Figure 156: EVPN-VXLAN Topology

The network topology shows two overlay (VXLAN) networks interconnected by an MPLS network:

- PE-69, PE-71 and PE-72 are part of Overlay-Network-1
- PE-70, PE-73 and PE-74 are part of Overlay-Network-2

CE-1, CE-2 and CE-3 belong to the same IP subnet, hence Layer 2 connectivity must be provided to them.

Note that the above topology can illustrate a DCI (Data Center Interconnect) example, where Overlay-Network-1 and Overlay-Network-2 are two Data Centers interconnected through an MPLS WAN. In this application, CE-1, CE-2 and CE-3 would simulate Virtual Machines or appliances, PE-69/70/72/73 would act as DC GWs and PE-71/74 as NVEs (or virtual PEs running on compute infrastructure).

The following protocols and objects are configured beforehand:

- The ports interconnecting the six PEs in [Figure 156](#) are configured as network ports (or hybrid) and have router network interfaces defined on them. Only the ports connected to the CEs are configured as access ports.
- The six PEs shown in the [Figure 156](#) are running IS-IS for the global routing table with the four core PEs interconnected using IS-IS Level-2 point-to-point interfaces and each overlay network is using IS-IS Level-1 point-to-point interfaces.
- LDP is used as the MPLS protocol to signal transport tunnel labels among PE-69, PE-72, PE-70 and PE-73. There is no LDP running in the two overlay networks.
- Note that the network port MTU (in all the ports sending/receiving VXLAN packets) must be at least 50-bytes (54 if dot1q encapsulation is used) greater than the service-mtu in order to accommodate the size of the VXLAN header.

Once the IGP infrastructure and LDP are enabled in the core, BGP has to be configured. In this example, two BGP families have to be enabled: EVPN within each overlay-network for the exchange of MAC/IP addresses and setting up the flooding domains, and L2-VPN for the use of BGP-MH and BGP-AD in the VPLS-MPLS network.

As an example, the following CLI output shows the relevant BGP configuration of PE-71, which only needs the EVPN family. PE-74 would have a similar BGP configuration. Note that the use of Route-Reflectors (RRs) in these type of scenarios is common. Although this example does not use RRs, an EVPN RR could have been used in Overlay-Network-1 and Overlay-Network-2 and an L2-VPN RR could have been used in the core VPLS-MPLS network.

```
A:PE-71>config>router>bgp# info
-----
vpn-apply-import
vpn-apply-export
min-route-advertisement 1
enable-peer-tracking
rapid-withdrawal
```

```

rapid-update evpn
group "DC"
    family evpn
    type internal
    neighbor 192.0.2.69
    exit
    neighbor 192.0.2.72
    exit
exit
no shutdown

```

The BGP configuration of PE-69 and PE-72 follows (PE-70 and PE-73 have an equivalent configuration).

A:PE-69>config>router>bgp# info

```

-----
vpn-apply-import
vpn-apply-export
min-route-advertisement 1
enable-peer-tracking
rapid-withdrawal
rapid-update l2-vpn evpn
group "DC"
    family l2-vpn evpn
    type internal
    neighbor 192.0.2.71
    exit
    neighbor 192.0.2.72
    exit
exit
group "WAN"
    family l2-vpn
    type internal
    neighbor 192.0.2.70
    exit
    neighbor 192.0.2.73
    exit
exit
no shutdown

```

A:PE-72>config>router>bgp# info

```

-----
vpn-apply-import
vpn-apply-export
min-route-advertisement 1
enable-peer-tracking
rapid-withdrawal
rapid-update l2-vpn evpn
group "DC"
    family l2-vpn evpn
    type internal
    neighbor 192.0.2.69
    exit
    neighbor 192.0.2.71
    exit
exit

```

Enabling EVPN-VXLAN in a VPLS Service

```
group "WAN"
  family l2-vpn
  type internal
  neighbor 192.0.2.70
  exit
  neighbor 192.0.2.73
  exit
exit
no shutdown
```

Figure 157 shows the BGP peering sessions among the PEs and the enabled BGP families. Note that, for instance, PE-71 will only establish an EVPN peering session with its peers (only the EVPN family is enabled on PE-71), even though PE-69 and PE-72 have EVPN and L2-VPN families configured.

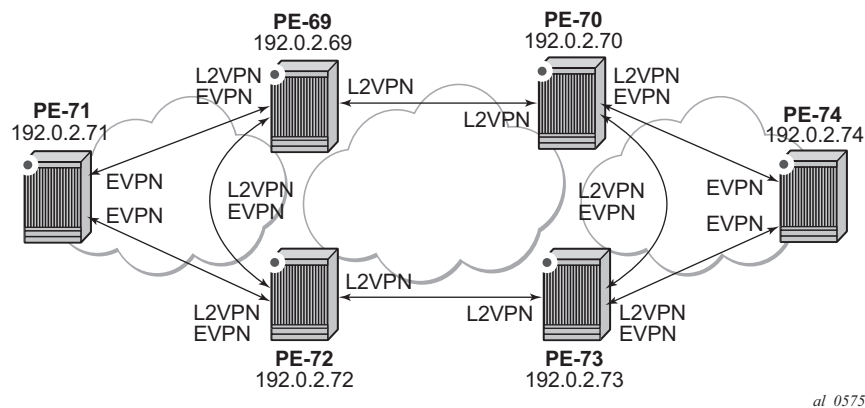


Figure 157: BGP Adjacencies and Enabled Families

Once the network infrastructure is running properly, the actual service configuration can be carried out. The following CLI outputs show the configuration of VPLS 1 in PE-71, PE-69 and PE-72 as per the topology illustrated in Figure 156.

VPLS 1 in those three PEs are interconnected using VXLAN bindings, whereas PE-69 and PE-72 are connected to the remote PEs by means of BGP-AD SDP-bindings. Although BGP-AD SDP-bindings are used in this example for the connectivity of the EVPN-VXLAN PEs to a regular VPLS network, SAPs, manual spoke-SDPs or mesh-SDPs could have been used instead. BGP-VPLS cannot be enabled in an EVPN-VXLAN VPLS service.

VPLS 1 configuration of PE-71 is shown below:

```
A:PE-71>config>service>vpls# info
```

```

-----
vxlan vni 1 create
exit
bgp
    route-distinguisher 192.0.2.71:1
    route-target export target:64500:12 import target:64500:12
exit
bgp-evpn
    vxlan
        no shutdown
    exit
exit
stp
    shutdown
exit
sap 1/1/1:1 create
exit
no shutdown
-----

```

EVPN-VXLAN is enabled by the configuration of a valid VXLAN Network Identifier (VNI) and the **bgp-evpn>vxlan>no shutdown** command. These two commands, along with the required **bgp** route-distinguisher (RD) and route-target (RT) information, are the minimum mandatory attributes:

- The VNI is a 24-bit identifier with valid values in the [1..16777215] range. This defines the VNI that the 7x50 will use in the EVPN routes generated for the VPLS service, and therefore the VNI that the system expects to see in the VXLAN packets destined to that particular VPLS service. Note that the configured VNI determines the VNI that has to be received in the packets for the VPLS service, but not the VNI that will be sent in VXLAN packets to remote PEs for the service. In other words, in this example, VPLS 1 is configured with VNI=1 in all the PEs, however each PE could have used a different VNI. Note that the VNI is a system-wide significant value and two VPLS services cannot be configured with the same VNI.
- The **bgp-evpn>vxlan>no shutdown** command enables the use of EVPN for VXLAN. It requires the previous configuration of the VNI, RD and RT. As soon as this command is executed, EVPN will advertise an inclusive multicast route to all of the BGP EVPN peers (regardless of the existing SAP/SDP-binding operational status). The exchange of inclusive multicast routes allows the establishment of the VXLAN bindings among the PEs.

Upon the reception of the EVPN inclusive multicast routes from PE-69 and PE-72, PE-71 will automatically setup its VXLAN bindings for VPLS-1. A VXLAN binding is represented by an (egress VTEP, egress VNI) pair, where VTEP is a VXLAN Termination End Point. This can be shown with the following show commands:

```

*A:PE-71# show service id 1 vxlan
=====
VPLS VXLAN, Ingress VXLAN Network Id: 1
=====

```

Enabling EVPN-VXLAN in a VPLS Service

```
Egress VTEP, VNI
=====
VTEP Address          Egress VNI    Num. MACs    In Mcast List?  Oper State
-----
192.0.2.69            1             1            Yes             Up
192.0.2.72            1             1            Yes             Up
-----
Number of Egress VTEP, VNI : 2
=====

*A:PE-71# show service vxlan
=====
VXLAN Tunnel Endpoints (VTEPs)
=====
VTEP Address          Number of Egress VNIs  Oper State
-----
192.0.2.69            2                      Up
192.0.2.72            2                      Up
-----
Number of VTEPs: 2
=====
```

As can be seen in the CLI output, PE-71 has two VXLAN bindings, one to PE-69 and one to PE-72. Both use egress VNI=1 (the actual VNI used in its egress VXLAN packets) and both are part of the flooding multicast list for VPLS 1 and are UP.

- The **In Mcast List? = Yes** entry is set when the proper inclusive multicast route is received from the remote VTEP. If the entry is **No**, the VXLAN binding will not be used to flood BUM (Broadcast, Unknown unicast, Multicast) packets.
- The **Oper State** is based on the existence of the VTEP in the global routing table.

The VPLS 1 configuration of PE-69 and PE-72 is shown below:

```
A:PE-69>config>service>vpls# info
-----
vxlan vni 1 create
exit
bgp
  route-distinguisher 192.0.2.69:1
  vsi-export "vsi-policy-1"
  vsi-import "vsi-policy-1"
  pw-template-binding 1 split-horizon-group "CORE"
  exit
exit
bgp-ad
  vpls-id 64500:1
  no shutdown
exit
bgp-evpn
  unknown-mac-route
  vxlan
    no shutdown
  exit
```

```

exit
stp
    shutdown
exit
site "site-1" create
    site-id 1
    split-horizon-group CORE
    no shutdown
exit
no shutdown
-----

A:PE-72>config>service>vpls# info
-----

vxlan vni 1 create
exit
bgp
    route-distinguisher 192.0.2.72:1
    vsi-export "vsi-policy-1"
    vsi-import "vsi-policy-1"
    pw-template-binding 1 split-horizon-group "CORE"
    exit
exit
bgp-ad
    vpls-id 64500:1
    no shutdown
exit
bgp-evpn
    unknown-mac-route
    vxlan
        no shutdown
    exit
exit
proxy-arp
    no age-time
    no send-refresh
    no shutdown
exit
stp
    shutdown
exit
site "site-1" create
    site-id 1
    split-horizon-group CORE
    no shutdown
exit
sap 1/1/1:1 create
exit
no shutdown
-----
    
```

In addition to the VNI and **bgp-evpn>vxlan>no shutdown** commands for enabling EVPN-VXLAN in VPLS 1, PE-69 and PE-72 require the configuration of BGP-AD for the discovery and establishment of FEC129 spoke SDPs to the remote PEs in the core, as well as BGP-MH for redundancy. As outlined in [Figure 156](#), there are two BGP-MH sites defined in the network: site-1 is used on PE-69/PE-72 and site-2 is used on PE-70/PE-73. Only one of the two gateway PEs in

each Overlay-Network will be the Designated Forwarder (DF) for VPLS 1, and only the DF will send/receive traffic for VPLS 1 in the Overlay-Network. The following considerations must be taken into account when configuring the connectivity of EVPN-VXLAN services to regular VPLS objects:

- As discussed, in this example, BGP-AD spoke-SDPs are used but SAPs, manual spoke-SDPs or mesh-SDPs are also supported.
- In this example, BGP-AD spoke-SDPs are auto-instantiated using **pw-template-binding 1 split-horizon-group "CORE"**.
→ Although not shown above, this requires the creation of the pw-template 1 (**config>service>pw-template 1 create**).
- The split-horizon-group CORE is added to the BGP-MH site "site-1". This statement will ensure that all the spoke SDPs automatically established to the remote PEs are part of the BGP-MH site.
- Although the route-targets for the Overlay-Network and the VPLS-MPLS network can have the same value for the same VPLS service, they are usually different. This example assumes the use of RT-DC-1 in Overlay-Network-1 and RT-WAN-1 in the VPLS-MPLS core for VPLS 1. The **vsi-policy-1** allows the system to export and import the right RTs for VPLS 1:

```
A:PE-69>config>router>policy-options# info
```

```
-----
community "RT-DC-1" members "target:64500:12"
community "RT-WAN-1" members "target:64500:11"
policy-statement "vsi-policy-1"
  entry 10 # to import all the evpn routes with RT-DC-1
    from
      community "RT-DC-1"
      family evpn
    exit
    action accept
    exit
  exit
  entry 20 # to import all the bgp-ad/mh routes from the WAN
    from
      community "RT-WAN-1"
      family l2-vpn
    exit
    action accept
    exit
  exit
  entry 30 # to export all the evpn routes with RT-DC-1
    from
      family evpn
    exit
    action accept
      community add "RT-DC-1"
    exit
  exit
  entry 40 # to export all the bgp-ad/mh routes with RT-WAN-1
    from
```



```

        family l2-vpn
    exit
    action accept
        community add "RT-WAN-1"
    exit
    exit
    default-action reject
exit

```

Once PE-69 and PE-72 are configured as above, they will setup the spoke SDPs and will run the DF election algorithm to determine the operational status of those spoke SDPs. Refer to [LDP VPLS using BGP-Auto Discovery](#) and [BGP Multi-Homing for VPLS Networks](#) for more information about the use of BGP-AD and BGP-MH.

Note that in the configuration for VPLS 1, both gateway PEs, PE-69 and PE-72 will attempt to establish two parallel Layer 2 paths between each other (a BGP-AD spoke SDP and a EVPN VXLAN binding). Since that would create a Layer 2 loop, the SR OS implementation gives priority to the EVPN path and only the VXLAN binding will be active. In other words, when an (egress VTEP, VNI) and a spoke SDP are attempted to be set up to the same far-end IP address at the same time, the VXLAN path will prevail and the spoke SDP will be kept down. The spoke SDP will only be brought up if the VXLAN (egress VTEP, VNI) goes down.

This behavior can be easily observed in this setup by using the following show commands. In PE-69, the spoke SDP to far-end PE-72 will be down with a **EvpnRouteConflict** Flag. The (egress VTEP, VNI) = (192.0.2.72, 1) VXLAN bind will be UP.

```

A:PE-69# show service id 1 base
=====
Service Basic Information
=====
Service Id      : 1                      Vpn Id      : 0
Service Type    : VPLS
Name            : (Not Specified)
Description     : (Not Specified)
Customer Id     : 1                      Creation Origin : manual
Last Status Change: 07/17/2014 00:03:52
Last Mgmt Change  : 07/17/2014 19:02:39
Etree Mode      : Disabled
Admin State     : Up                      Oper State      : Up
MTU             : 1514                     Def. Mesh VC Id : 1
...
=====
Service Access & Destination Points
=====
Identifier                               Type      AdmMTU  OprMTU  Adm  Opr
-----
sdp:17405:4294967290 SB(192.0.2.73)      BgpAd     0       8974    Up   Up
sdp:17406:4294967292 SB(192.0.2.72)      BgpAd     0       8974    Up   Down
sdp:17407:4294967294 SB(192.0.2.70)      BgpAd     0       8974    Up   Up
=====
A:PE-69# show service id 1 all | match Flag
Flags                               : None

```

Enabling EVPN-VXLAN in a VPLS Service

```
Flags          : EvpnRouteConflict
Flags          : None
```

```
A:PE-69# show service id 1 vxlan
```

```
=====
VPLS VXLAN, Ingress VXLAN Network Id: 1
=====
```

```
Egress VTEP, VNI
```

```
=====
VTEP Address      Egress VNI    Num. MACs    In Mcast List?  Oper State
-----
192.0.2.71         1              0            Yes             Up
192.0.2.72         1              0            Yes             Up
=====
```

```
Number of Egress VTEP, VNI : 2
=====
```

At the non-DF, PE-72, all the spoke SDPs will be down due to BGP-MH:

```
A:PE-72# show service id 1 base
```

```
=====
Service Basic Information
=====
```

```
Service Id       : 1                      Vpn Id          : 0
Service Type     : VPLS
Name             : (Not Specified)
Description      : (Not Specified)
Customer Id      : 1                      Creation Origin  : manual
Last Status Change: 07/17/2014 00:03:42
Last Mgmt Change : 07/17/2014 19:02:50
Etree Mode       : Disabled
Admin State      : Up                    Oper State      : Up
MTU              : 1514                  Def. Mesh VC Id : 1
SAP Count        : 1                    SDP Bind Count  : 3
...
```

```
-----
Service Access & Destination Points
-----
```

```
Identifier                      Type      AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/1:1                     q-tag     1518    1518    Up   Up
sdp:17405:4294967290 SB(192.0.2.73) BgpAd     0      8974    Up   Down
sdp:17406:4294967292 SB(192.0.2.69) BgpAd     0      8974    Up   Down
sdp:17407:4294967294 SB(192.0.2.70) BgpAd     0      8974    Up   Down
=====
```

```
A:PE-72# show service id 1 all | match Flag
```

```
Flags          : StandbyForMHPProtocol
Flags          : StandbyForMHPProtocol
Flags          : StandbyForMHPProtocol
Flags          : None
```

MAC Learning and unknown-mac-route

Once the VPLS service (VPLS 1) is configured, the network allows the CEs to exchange unicast and BUM traffic over the Overlay and VPLS-MPLS service infrastructure. BUM traffic sent by CE-1 will be ingress-replicated to PE-69 and PE-72 by PE-71, and propagated by PE-69 (the DF) to the remote network. From this point on, MAC addresses will be learned on active SAPs and spoke SDPs and advertised in EVPN MAC routes. No data plane MAC learning is carried out on VXLAN bindings. MACs associated with (egress VTEP, VNI) bindings will always be learned through EVPN.

The following CLI output shows the reception of an EVPN MAC route and how the (CE-2) MAC address appears in the FDB for VPLS 1.

```
33 2014/07/17 22:06:08.48 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.72
"Peer 1: 192.0.2.72: UPDATE
Peer 1: 192.0.2.72 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 88
  Flag: 0x90 Type: 14 Len: 44 Multiprotocol Reachable NLRI:
    Address Family EVPN
    NextHop len 4 NextHop 192.0.2.72
    Type: EVPN-MAC Len: 33 RD: 192.0.2.72:1 ESI: 0:0:0:0:0:0:0:0:0:0, tag: 1
, mac len: 48 mac: 00:ca:fe:ca:fe:02, IP len: 0, IP: NULL, label: 0
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 16 Extended Community:
    target:64500:12
    bgp-tunnel-encap:VXLAN
"
```

```
*A:PE-71# show service id 1 fdb detail
=====
Forwarding Database, Service 1
=====
```

ServId	MAC	Source-Identifier	Type	Last Change
1	00:ca:fe:ca:fe:01	sap:1/1/1:1	L/0	07/17/14 22:06:08
1	00:ca:fe:ca:fe:02	vxlan: 192.0.2.72:1	Evpn	07/17/14 22:06:08

```
-----
No. of MAC Entries: 2
-----
Legend: L=Learned O=Oam P=Protected-MAC C=Conditional S=Static
=====
*A:PE-71#
```

When a frame destined to 00:ca:fe:ca:fe:02 enters SAP 1/1/1:1, it is encapsulated into a VXLAN packet with outer destination IP 192.0.2.72 and VNI 1, and sent on the wire.

In virtualized data center networks where all the MACs are known beforehand (all the virtual machine and appliance MACs are distributed by EVPN before any traffic flows), unknown MAC addresses are always outside the data center. If that is the case, the DC GWs can make use of the **unknown-mac-route** so that the DC NVEs supporting the concept of this route send the unknown unicast traffic only to the DC GW. This minimizes the flooding within the Data Center, as explained in draft-rabadan-l2vpn-dci-evpn-overlay.

In this example the unknown-mac-route is configured in the gateway PEs (PE-69, PE-72 and PE-70, PE-73) in the following way:

```
*A:PE-69>config>service>vpls# bgp-evpn unknown-mac-route
*A:PE-69>config>service>vpls#
27 2014/07/17 22:15:54.94 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.71
"Peer 1: 192.0.2.71: UPDATE
Peer 1: 192.0.2.71 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 88
  Flag: 0x90 Type: 14 Len: 44 Multiprotocol Reachable NLRI:
    Address Family EVPN
    NextHop len 4 NextHop 192.0.2.69
    Type: EVPN-MAC Len: 33 RD: 192.0.2.69:1 ESI: 0:0:0:0:0:0:0:0:0, tag: 1
, mac len: 48 mac: 00:00:00:00:00:00, IP len: 0, IP: NULL, label: 0
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 16 Extended Community:
    target:64500:12
    bgp-tunnel-encap:VXLAN
"
```

...

Note that:

- Although the 7x50 can generate the unknown-mac-route, it will never honor it and normal flooding applies when an unknown unicast packet arrives at an ingress sap/sdp-binding.
- When unknown-mac-route is configured, it will ONLY be generated when: a) no BGP-MH site is configured within the same VPLS service or b) a site is configured AND the site is DF (Designated Forwarder) in the PE. If the site becomes a non-DF site, the unknown-mac-route will be withdrawn.
- If the unknown-mac-route is used in the DC GW and all the NVEs in the DC understand it, the advertisement of MAC addresses can be disabled with the **[no] mac-advertisement** command. If so, the 7x50 will only advertise the unknown-mac-route.

```
*A:PE-72>config>service>vpls>bgp-evpn# info
-----
unknown-mac-route
no mac-advertisement
vxlan
    no shutdown
exit
-----
```

Scaling BGP-MH Resiliency with the Use of Operational Groups

In [Figure 156](#), VPLS 1 in PE-69/PE-72 is configured with a BGP-MH site that controls which of the two PEs forwards the traffic to the remote PEs (in this case PE-69 is the DF and the gateway responsible for forwarding packets to the remote PEs).

When new VPLS services are required in PE-69/PE-72 the same BGP-MH configuration can be used. However, if the number of VPLS services grows significantly, the use of individual BGP-MH sites per service will not scale. Since all the services in these two PEs share the same physical topology, the use of oper-groups can provide a simple and scalable way of providing resiliency to as many services as the user needs (up to the maximum number of VPLS services per system).

The way oper-groups can be used to scale this type of deployments is the following (using the network topology in Figure 1 and focusing on Overlay-Network-1):

- A control-VPLS service is defined in PE-69 and PE-72. For instance, VPLS 1.
 - This service is configured with a BGP-MH site in both PEs.
 - An oper-group **control-vpls-1** is created and associated to the pw-template-binding 1 in VPLS 1.
- Data VPLS services are defined in both PEs. For instance: VPLS 2, VPLS 3,... VPLS 999.
 - In all these services, the pw-template-binding is configured with **monitor-oper-group “control-vpls-1”**.
 - The status of the spoke SDPs in the data VPLS services depends on the status of the oper-group. If there is a DF switchover in VPLS 1 and VPLS 1 spoke SDPs go down on PE-69, all the spoke SDPs in all the data VPLS services controlled by **control-vpls-1** in PE-69 will go down too. In the same way, the spoke SDPs in PE-72 will come up.
- To allow per-service load balancing a second control-VPLS service with a different BGP-MH site should be configured.
 - For instance, VPLS 1 might have PE-69 as the DF and VPLS 1000 might be a second control-VPLS service with PE-72 as the DF.
 - Each control-VPLS would control a group of data VPLS services based on the definition and association of a second oper-group.

The following example shows the configuration of VPLS 1 as the control-VPLS and VPLS 2 as a data-VPLS. VPLS 1 controls the VPLS 2 spoke SDP status.

```
*A:PE-69>config>service# info
-----
customer 1 create
  description "Default customer"
exit
pw-template 1 create
exit
oper-group "control-vpls-1" create
exit
vpls 1 customer 1 create
  description "control-VPLS"
  vxlan vni 1 create
  exit
  bgp
    route-distinguisher 192.0.2.69:1
    vsi-export "vsi-policy-1"
    vsi-import "vsi-policy-1"
    pw-template-binding 1 split-horizon-group "CORE"
    oper-group "control-vpls-1"
  exit
exit
bgp-ad
  vpls-id 64500:1
  no shutdown
exit
bgp-evpn
  unknown-mac-route
  vxlan
    no shutdown
  exit
exit
stp
  shutdown
exit
site "site-1" create
  site-id 1
  split-horizon-group CORE
  no shutdown
exit
no shutdown
exit
vpls 2 customer 1 create
  description "data-VPLS"
  vxlan vni 2 create
  exit
  bgp
    route-distinguisher 192.0.2.69:2
    vsi-export "vsi-policy-2"
    vsi-import "vsi-policy-2"
    pw-template-binding 1
    monitor-oper-group "control-vpls-1"
  exit
exit
bgp-ad
```

Enabling EVPN-VXLAN in a VPLS Service

```
vpls-id 64500:2
no shutdown
exit
bgp-evpn
  unknown-mac-route
  vxlan
    no shutdown
  exit
exit
stp
  shutdown
exit
no shutdown
exit
```

Use of Proxy-ARP in EVPN-VXLAN Services

EVPN-VXLAN services support proxy-ARP functionality that is enabled by the **proxy-arp [no] shutdown** command. The default value is shutdown. When proxy-arp is enabled:

- MAC and IP addresses contained in the received valid EVPN MAC routes are populated in the proxy-ARP table.
- ARP-request messages received on SAPs and SDP-binds are intercepted and the target IP address is looked up. If the IP address is found, an ARP reply will be issued based on the information found in the proxy-ARP table, otherwise the ARP request would be flooded in the VPLS service (except for the source SAP/SDP binding).
- ARP-reply messages received on SAPs and SDP-bindings are also intercepted and sent to the CPM. These ARP-reply messages are re-injected in the data plane and forwarded based on the FDB information to the destination MAC address. If the destination MAC address is not in the FDB, the ARP-reply message will be flooded in the VPLS service (except for the source SAP/SDP binding).

The following CLI output shows the proxy-ARP configuration in PE-72 and a received valid MAC route that includes the MAC and IP of CE-1. This MAC-IP pair is installed in the proxy-ARP table for VPLS 1.

```
*A:PE-72>config>service>vpls# info
```

```
vxlan vni 1 create
exit
bgp
  route-distinguisher 192.0.2.72:1
  vsi-export "vsi-policy-1"
  vsi-import "vsi-policy-1"
  pw-template-binding 1 split-horizon-group "CORE"
    oper-group "control-vpls-1"
  exit
exit
```



```

    bgp-ad
      vpls-id 64500:1
      no shutdown
    exit
    bgp-evpn
      unknown-mac-route
      vxlan
        no shutdown
      exit
    exit
    proxy-arp
      no shutdown
    exit
  ...

27 2014/07/17 23:15:54.85 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.71
"Peer 1: 192.0.2.71: UPDATE
Peer 1: 192.0.2.71 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 92
  Flag: 0x90 Type: 14 Len: 48 Multiprotocol Reachable NLRI:
    Address Family EVPN
    NextHop len 4 NextHop 192.0.2.71
    Type: EVPN-MAC Len: 37 RD: 192.0.2.71:1 ESI: 0:0:0:0:0:0:0:0:0, tag: 1
, mac len: 48 mac: 00:ca:fe:ca:fe:01, IP len: 4, IP: 172.16.0.1, label: 0
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 16 Extended Community:
    target:64500:12
    bgp-tunnel-encap:VXLAN
"

*A:PE-72# show service id 1 proxy-arp
-----
VPLS Proxy Arp Table
-----
IP Address          Mac Address
-----
172.16.0.1          00:ca:fe:ca:fe:01
-----
Number of entries : 1
-----
=====
*A:PE-72#

```

Note that in the tested release, the 7x50 does not include a host IP address in any EVPN MAC advertisement for a MAC learned on a SAP or SDP-bind. Host IP addresses are only included in the EVPN MAC advertisements corresponding to R-VPLS IP interfaces. When deployed as DC GW in a Nuage architecture, the Nuage Networks VSC (Virtual Services Controller) or VSG (Virtual Services Gateway) will send virtual machine and host MAC/IP pairs in EVPN MAC routes. Please refer to the Alcatel-Lucent Nuage documentation for more information about the Nuage DC architecture. The 7x50 DC GW will populate the proxy-ARP tables with those MAC/IP pairs. In the CLI excerpt above, assume that PE-71 is replaced by a Nuage VSC that sends the

pair <172.16.0.1, 00:ca:fe:ca:fe:01> in an EVPN MAC route. PE-72 receives the advertisement and adds the entry to its proxy-ARP table for VPLS 1.

MAC Mobility, MAC Duplication and MAC Protection in EVPN

MAC mobility, duplication and protection are fully supported as specified in draft-ietf-l2vpn-evpn. [Figure 158](#) illustrates the concept of mobility (Virtual Machine VM-1 moves from PE-71 to PE-72).

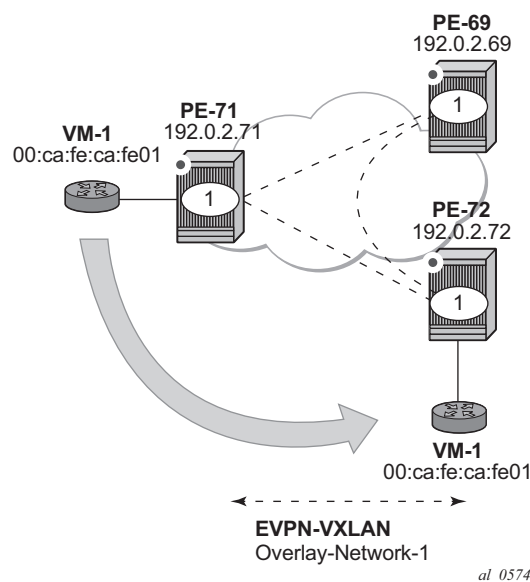


Figure 158: EVPN MAC Mobility

MAC mobility is handled in EVPN by the use of sequence numbers in the MAC routes. When 00:ca:fe:ca:fe:01 moves from PE-71 to PE-72, SR OS will gracefully handle it in this way:

- 00:ca:fe:ca:fe:01 moves to PE-72 SAP 1/1/1:1
- PE-72 advertises 00:ca:fe:ca:fe:01 using a higher sequence number (the first time a MAC is advertised, EVPN uses sequence number 0).
- PE-69 at this point has two valid MAC routes for 00:ca:fe:ca:fe:01. It picks up the one coming from PE-72 since the sequence number is higher.
- PE-71 receives the MAC route, and since the sequence number is higher than the one for its own route, it updates the FDB and withdraws its own MAC route.

However, if MAC 00:ca:fe:ca:fe:01 is constantly learned on the PE-71 and PE-72 SAPs, the process above causes an endless exchange of MAC route advertisements and withdraws that has a negative impact on all the PEs in the EVPN network. This issue is known as “MAC duplication” and is originated by a loop at the access or a duplicated MAC address in two hosts of the same service. SR OS solves this issue through the use of the mac-duplication detection feature. Note that mac-duplication is always enabled with the following default settings:

```
*A:PE-71>config>service>vpls>bgp-evpn# info detail
-----
no unknown-mac-route
mac-advertisement
no ip-route-advertisement
mac-duplication
    detect num-moves 5 window 3
    retry 9
exit
vxlan
    no shutdown
exit
-----
```

Where:

- **num-moves** — Identifies the number of MAC moves in a VPLS service. The counter is incremented when a given MAC is locally relearned in the FDB or flushed from the FDB due to the reception of a better remote EVPN route for that MAC. When the threshold is reached for a given MAC, this MAC is put in hold-down state (this ‘hold-down’ state is described below). Range: <3..10>. Default value: 5.
- **window** — Identifies the timer within which a MAC is considered duplicate if it reaches the configured num-moves. Range: <1..15> minutes. Default value: 3 minutes.
- **Retry** — The timer after which the MAC in hold-down state is automatically flushed and the mac-duplication process starts again. This value is expected to be equal to two times or more than the window. If no retry is configured, this implies that, once mac-duplication is detected, MAC updates for that MAC will be held down until the user intervenes or a network event (that flushes the MAC) occurs. Range: <2..60> minutes. Default value: 9 minutes.

When a MAC is considered a duplicate or in the ‘hold-down’ state, no further BGP advertisements are issued for this MAC and an alarm is triggered (by the first MAC in hold-down state). The following CLI output shows how PE-72 detects that MAC 00:ca:fe:ca:fe:01 is a duplicate (after reaching the **num-moves** in **window**) and the corresponding alarm. The **show service id bgp-evpn** command shows the mac-duplication settings and the list of duplicate MACs on hold-down.

MAC Mobility, MAC Duplication and MAC Protection in EVPN

```
41 2014/07/17 23:50:45.83 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.71
"Peer 1: 192.0.2.71: UPDATE
Peer 1: 192.0.2.71 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 96
  Flag: 0x90 Type: 14 Len: 44 Multiprotocol Reachable NLRI:
    Address Family EVPN
    NextHop len 4 NextHop 192.0.2.72
    Type: EVPN-MAC Len: 33 RD: 192.0.2.72:1 ESI: 0:0:0:0:0:0:0:0:0, tag: 1
, mac len: 48 mac: 00:ca:fe:ca:fe:01, IP len: 0, IP: NULL, label: 0
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 24 Extended Community:
    target:64500:12
    bgp-tunnel-encap:VXLAN
    mac-mobility:Seq:4
"
```

```
2 2014/07/17 23:50:46.62 UTC MINOR: SVCMGR #2331 Base
"VPLS Service 1 has MAC(s) detected as duplicates by EVPN mac-duplication detect
ion."
```

```
*A:PE-72# show service id 1 bgp-evpn
```

```
=====
BGP EVPN Table
=====
MAC Advertisement      : Enabled          Unknown MAC Route      : Enabled
VXLAN Admin Status    : Enabled          Creation Origin        : manual
MAC Dup Detn Moves     : 5                MAC Dup Detn Window: 3
MAC Dup Detn Retry     : 9                Number of Dup MACs    : 1
IP Route Advertise*    : Disabled
```

```
-----
Detected Duplicate MAC Addresses          Time Detected
-----
00:ca:fe:ca:fe:01                        07/17/2014 23:50:47
-----
* indicates that the corresponding row element may have been truncated.
```

The 7x50 stops sending and processing any BGP MAC Advertisement routes for that MAC address until:

- The MAC is flushed due to a local event (SAP/SDP-binding associated to the MAC fails) or the reception of a remote withdraw for the MAC (due to a MAC flush at the remote 7x50) or
- The **retry <in_minutes>** timer expires, which flushes the MAC and restart the process.

When the last duplicate MAC address is removed from the duplicate list, the system will show the following message:

```
*A:PE-72#
3 2014/07/17 23:56:21.71 UTC MINOR: SVCMGR #2332 Base
"VPLS Service 1 no longer has MAC(s) detected as duplicates by EVPN mac-duplicate
ion detection."
```

EVPN also provides a mechanism to protect certain MACs that do not move for which connectivity must be guaranteed. These addresses must be protected in case there is an attempt to dynamically learn them in a different place in the EVPN-VXLAN VPLS service (on the same or different PE).

The protected MACs are configured in SR OS as conditional static MACs. A conditional static MAC defined in an EVPN-VXLAN VPLS service is advertised by BGP-EVPN as a static address. An example of the configuration of a conditional static MAC is shown below:

```
*A:PE-71>config>service>vpls# info
-----
vxlan vni 1 create
exit
bgp
  route-distinguisher 192.0.2.71:1
  route-target export target:64500:12 import target:64500:12
exit
bgp-evpn
  vxlan
    no shutdown
  exit
exit
proxy-arp
  no shutdown
exit
sap 1/1/1:1 create
exit
static-mac
  mac 00:ca:fe:ca:fe:05 create sap 1/1/1:1 monitor fwd-status
exit
no shutdown
-----
```

The protected MACs advertised in EVPN are shown in the receiving BGP RIB as Static (MAC mobility extended community with Sequence 0 and sticky bit set) and **EvpnS** (Evpn Static) in the FDB. The advertising PE shows the protected MAC as **CStatic** (Conditional Static) in the FDB:

MAC Mobility, MAC Duplication and MAC Protection in EVPN

```
# advertising PE
*A:PE-71>config>service>vpls# show service id 1 fdb detail
=====
Forwarding Database, Service 1
=====
ServId      MAC                Source-Identifier      Type      Last Change
              Age
-----
1           00:ca:fe:ca:fe:05  sap:1/1/1:1          CStatic   07/18/14 00:29:34
-----
No. of MAC Entries: 1
-----
Legend:  L=Learned O=Oam P=Protected-MAC C=Conditional S=Static
=====

# receiving PE

*A:PE-72# show service id 1 fdb detail
=====
Forwarding Database, Service 1
=====
ServId      MAC                Source-Identifier      Type      Last Change
              Age
-----
1           00:ca:fe:ca:fe:05  vxlan:
              192.0.2.71:1          EvpnS     07/18/14 00:29:35
-----
No. of MAC Entries: 1
-----
Legend:  L=Learned O=Oam P=Protected-MAC C=Conditional S=Static
=====

*A:PE-72# show router bgp routes evpn mac mac-address 00:ca:fe:ca:fe:05 hunt
=====
BGP Router ID:192.0.2.72      AS:64500      Local AS:64500
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP EVPN Mac Routes
=====
-----
RIB In Entries
-----
Network      : N/A
Nexthop      : 192.0.2.71
From         : 192.0.2.71
Res. Nexthop : N/A
Local Pref.  : 100
Aggregator AS : None
Atomic Aggr. : Not Atomic
AIGP Metric  : None
Connector    : None
Community    : target:64500:12 bgp-tunnel-encap:VXLAN
               mac-mobility:Seq:0/Static
Cluster      : No Cluster Members
Originator Id : None
Peer Router Id : 192.0.2.71
Interface Name : NotAvailable
Aggregator    : None
MED           : 0
```

```

Flags          : Used Valid Best IGP
Route Source   : Internal
AS-Path        : No As-Path
EVPN type      : MAC
ESI            : 0:0:0:0:0:0:0:0:0      Tag          : 1
IP Address     : N/A                   Route Dist.   : 192.0.2.71:1
Mac Address    : 00:ca:fe:ca:fe:05
MPLS Label1    : 0                     MPLS Label2   : N/A
Route Tag      : 0
Neighbor-AS    : N/A
Orig Validation: N/A
Source Class   : 0                     Dest Class    : 0

```

```

-----
RIB Out Entries
-----

```

```

Routes : 1

```

```

=====
*A:PE-72#

```

The following procedures are supported in order to protect the configured static MAC addresses:

- All the SAP/SDP-bindings are internally configured as MAC protect restrict-protected-src as soon as bgp-evpn is enabled in the VPLS service.
- Local static MACs or remote EVPN Static MACs are considered as protected.
- If a frame with a source MAC address matching one of the protected MACs is received on a different SAP/SDP-binding than the owner of the protected MAC, the frame is discarded and an alarm triggered. Note that this MAC protection is not performed for frames received on VXLAN bindings.
- The same throttled alarm mechanism used in MAC protect for restrict-protected-src with discard-frame is used here: the offending frames are captured to a list to be polled by the CPM every ~10min.

In this example, PE-72 has 00:ca:fe:ca:fe:05 in its FDB as EvpnS. If SAP 1/1/1:1 receives a frame with source MAC address 00:ca:fe:ca:fe:05, the frame is discarded and an alarm triggered:

```

*A:PE-72#
4 2014/07/18 00:33:49.05 UTC MINOR: SVCMDR #2208 Base Slot 1
"Protected MAC 00:ca:fe:ca:fe:05 received on SAP 1/1/1:1 in service 1. "

```

Debug and Show Commands

In addition to the previously mentioned **show service id vxlan**, **show service id bgp-evpn** and **show service id fdb detail** commands, the following commands provide valuable information when troubleshooting an EVPN-VXLAN VPLS service.

The **show router bgp routes evpn** command supports filtering by route type as well as many other route fields.

```
*A:PE-72# show router bgp routes evpn
- evpn <evpn-type>

    inclusive-mcast - Display BGP EVPN Inclusive-Mcast Routes
    ip-prefix       - Display BGP EVPN IP-Prefix Routes
    mac             - Display BGP EVPN Mac Routes

*A:PE-72# show router bgp routes evpn mac
{hunt|detail}
  hunt      detail
rd <rd>
next-hop <ip-address>
mac-address <mac-address>
community <comm-id>
tag <vni-id>

*A:PE-72# show router bgp routes evpn mac tag 1
=====
BGP Router ID:192.0.2.72      AS:64500      Local AS:64500
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP EVPN Mac Routes
=====
Flag  Route Dist.      ESI              Tag      MacAddr
      NextHop
      IpAddr
      Mac Mobility
-----
u*>i  192.0.2.69:1      0:0:0:0:0:0:0:0  1        00:00:00:00:00:00
      192.0.2.69      N/A
      Seq:0

u*>i  192.0.2.71:1      0:0:0:0:0:0:0:0  1        00:ca:fe:ca:fe:05
      192.0.2.71      N/A
      Static

-----
Routes : 2
=====
```


The **tools dump service id vxlan** displays the number of times a service could not add a VXLAN binding or <VTEP, Egress VNI> due to the following limits:

- The per System VTEP limit has been reached
- The per System (egress VTEP, egress VNI) limit has been reached
- The per Service (egress VTEP, egress VNI) limit has been reached
- The per System Bind limit: Total bind limit or VXLAN bind limit has been reached.

Tools dump service vxlan usage displays the consumed VXLAN resources in the system, whereas **tools dump service vxlan dup-vtep-egrvni** displays the (egress VTEP, egress VNI) bindings that have been detected as duplicate attempts, in other words, an attempt to add the same binding to more than one service:

```
*A:PE-72# tools dump service id 1 vxlan

VTEP, Egress VNI Failure statistics at 001 01:06:55.950:

statistics last cleared at 000 00:00:00.000:

Failures: None

*A:PE-72# tools dump service vxlan usage

VXLAN usage statistics at 001 01:07:59.790:

VTEP                               :      2/8191
VTEP, Egress VNI                   :      4/131071
Sdp Bind + VTEP, Egress VNI       :     10/196607
RVPLS Egress VNI                   :      0/40959

*A:PE-72# tools dump service vxlan dup-vtep-egrvni

Duplicate VTEP, Egress VNI usage attempts at 001 01:08:04.080:

1. 192.0.2.71:100
```

Conclusion

SR OS supports the EVPN control plane for VXLAN tunnels terminated in VPLS services. VXLAN is an overlay IP tunneling mechanism that is being used in data center, data center interconnect and other applications. EVPN is a scalable and flexible control plane that provides control over the MACs being learned and advertised, as well as other mechanisms to optimize Layer 2 services such as proxy-ARP, MAC mobility, MAC duplication detection and MAC protection. SR OS provides a resilient and scalable EVPN-VXLAN solution for Layer 2 services, including interoperability to existing VPLS networks. This example showed all of those functions and how they are configured and operated.

EVPN for VXLAN Tunnels (Layer 3)

In This Chapter

This section provides information about EVPN for VXLAN tunnels (Layer 3).

Topics in this section include:

- [Applicability on page 1064](#)
- [Overview on page 1065](#)
- [Configuration on page 1066](#)
- [Conclusion on page 1102](#)

Applicability

This example is applicable to the 7950 XRS, 7750 SR-c4/c12, 7750 SR-7/12 and 7450 ESS-6/6v/7/12, but it is not supported on 7750 SR-1, 7450 ESS-1 or 7710 SR. Virtual eXtensible Local Area Network (VXLAN) requires IOM3-XP/IMM or higher-based line cards and chassis-mode D. Ethernet Virtual Private Networks (EVPN) is a control plane technology and does not have line card hardware dependencies.

The configuration was tested in release 12.0.R4. The [EVPN for VXLAN Tunnels \(Layer 2\) on page 1033](#) example is pre-requisite reading.

Overview

As discussed in the [EVPN for VXLAN Tunnels \(Layer 2\) on page 1033](#) example, EVPN and VXLAN can be enabled on VPLS or R-VPLS services in SR OS. While that example focuses on the use of EVPN-VXLAN layer 2 services, that is how EVPN-VXLAN is configured in VPLS services, this example describes how EVPN-VXLAN can be used to provide inter-subnet forwarding in R-VPLS and VPRN services. Inter-subnet forwarding can be provided by regular R-VPLS and VPRN services, however EVPN provides an efficient and unified way to populate FDBs (Forwarding Data Bases), ARP (Address Resolution Protocol) tables and routing tables using a single BGP address family. Inter-subnet forwarding in overlay networks would otherwise require data plane learning and the use of routing protocols on a per VPRN basis.

The SR OS solution for inter-subnet forwarding using EVPN is based on building blocks described in draft-sajassi-l2vpn-evpn-inter-subnet-forwarding and the use of the EVPN ip-prefix routes (routes type-5) as explained in draft-rabadan-l2vpn-evpn-prefix-advertisement. This example describes three supported common scenarios and provides the CLI configuration and required tools to troubleshoot EVPN-VXLAN in each case. The scenarios tested and explained are:

- EVPN-VXLAN in R-VPLS services
- EVPN-VXLAN in IRB (Integrated Routing Bridging) Backhaul R-VPLS services
- EVPN-VXLAN in EVPN Tunnel R-VPLS services

In all these scenarios redundant PEs are usually deployed. If that is the case, the interaction of EVPN, IP-VPN and the routing table manager (RTM) may lead to some routing loop situations that must be avoided by the use of routing policies (note that this also may happen in traditional IP-VPN deployments when eBGP and MP-BGP interact to populate VPRN routing tables in multi-homed networks). This section explains when those routing loops can happen and how to avoid them.

The term IRB interface refers to an R-VPLS service bound to a VPRN IP interface. The terms IRB interface and R-VPLS interface are used interchangeably throughout this example.

Configuration

This section describes the configuration of EVPN-VXLAN for Layer 3 services on the 7x50, as well as the available troubleshooting and show commands. The three scenarios described in the overview are analyzed independently.

EVPN-VXLAN in an R-VPLS Service

Figure 159 shows the topology used in the first scenario.

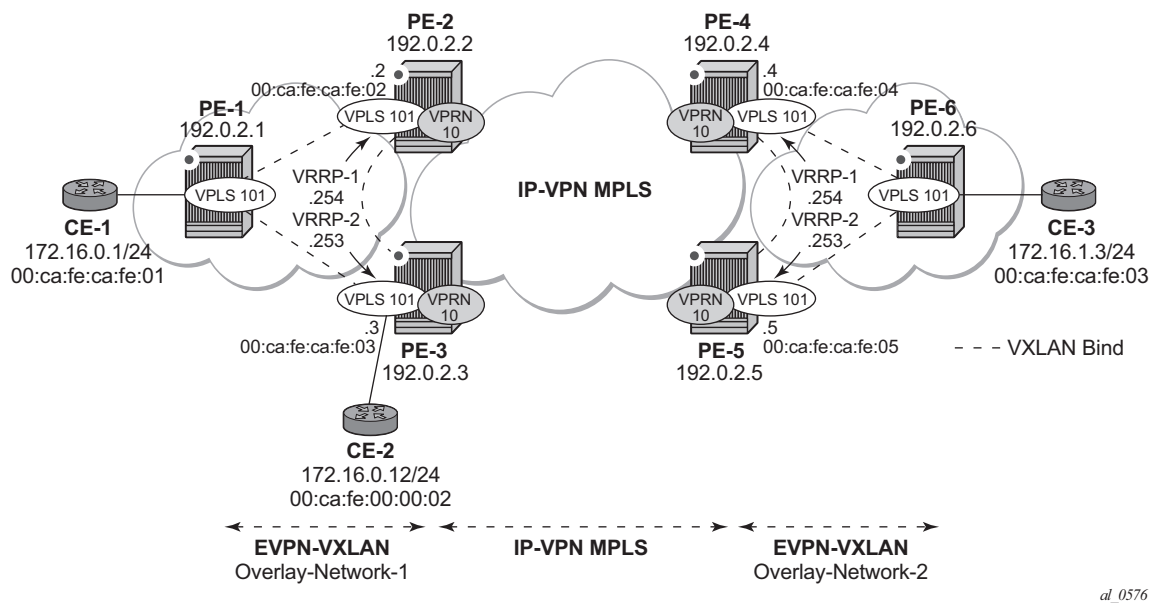


Figure 159: EVPN-VXLAN for R-VPLS Services

The network topology shows two overlay (VXLAN) networks interconnected by an MPLS network:

- PE-1, PE-2 and PE-3 are part of Overlay-Network-1
- PE-4, PE-5 and PE-6 are part of Overlay-Network-2

A Layer 2/Layer 3 service is provided to a customer to connect CE-1, CE-2 and CE-3. In this scenario, Layer 2 connectivity is provided within each overlay network and inter-subnet connectivity (Layer 3) is provided between the overlay networks, hence VPLS 101 is defined within each overlay network and VPRN 10 connects both Layer 2 services through an IP-VPN MPLS network.

Note that the above topology can illustrate a Data Center Interconnect (DCI) example, where Overlay-Network-1 and Overlay-Network-2 are two data centers interconnected through an MPLS WAN. In this application, CE-1, CE-2 and CE-3 would simulate virtual machines or appliances, PE-2/3/4/5 would act as Data Center gateways (DC GW) GWs and PE-1/6 as Network Virtualization Edge devices (or virtual PEs running on a compute infrastructure).

The following protocols and objects are configured beforehand:

- The ports interconnecting the six PEs in [Figure 159](#) are configured as network ports (or hybrid) and will have router network interfaces defined in them. Only the ports connected to the CEs are configured as access ports.
- The six PEs are running IS-IS for the global routing table with the four core PEs interconnected using IS-IS Level-2 point-to-point interfaces and each overlay network using IS-IS Level-1 point-to-point interfaces.
- LDP is used as the MPLS protocol to signal transport tunnel labels among PE-2, PE-3, PE-4 and PE-5. There is no LDP running within each overlay network.
- Note that the network port MTU (in all the ports sending/receiving VXLAN packets) must be at least 50-bytes (54 if dot1q encapsulation is used) greater than the service-mtu in order to accommodate the size of the VXLAN header.

Once the IGP infrastructure and LDP in the core are enabled, BGP has to be configured. In this scenario, two BGP families have to be enabled: EVPN within each overlay-network for the exchange of MAC/IP addresses and setting up the flooding domains, and VPN-IPv4 among the four core PEs so that IP-prefixes can be exchanged and resolved to MPLS tunnels in the core.

As an example, the following CLI output shows the relevant BGP configuration of PE-1, which only needs the EVPN family. PE-6 has a similar BGP configuration, that is, only EVPN family is configured for its peers. Note that the use of Route-Reflectors (RRs) in these type of scenarios is common. Although this scenario does not use RRs, an EVPN RR could have been used in Overlay-Network-1 and Overlay-Network-2 and a separate VPN-IPv4 RR could have been used in the core IP-VPN MPLS network.

```
A:PE-1>config>router>bgp# info
-----
vpn-apply-import
vpn-apply-export
enable-peer-tracking
rapid-withdrawal
rapid-update evpn
group "DC"
    family evpn
        type internal
        neighbor 192.0.2.2
        exit
        neighbor 192.0.2.3
        exit
    exit
no shutdown
-----
```

The BGP configuration of PE-2 and PE-3 follows (PE-4 and PE-5 have an equivalent configuration).

```
A:PE-2>config>router>bgp# info
```

```
-----
vpn-apply-import
vpn-apply-export
min-route-advertisement 1
enable-peer-tracking
rapid-withdrawal
rapid-update evpn
group "DC"
    family vpn-ipv4 evpn
    type internal
    neighbor 192.0.2.1
    exit
    neighbor 192.0.2.3
    exit
exit
group "WAN"
    family vpn-ipv4
    type internal
    neighbor 192.0.2.4
    exit
    neighbor 192.0.2.5
    exit
exit
no shutdown
-----
```

```
A:PE-3>config>router>bgp# info
```

```
-----
vpn-apply-import
vpn-apply-export
min-route-advertisement 1
enable-peer-tracking
rapid-withdrawal
rapid-update evpn
group "DC"
    family vpn-ipv4 evpn
    type internal
    neighbor 192.0.2.1
    exit
    neighbor 192.0.2.2
    exit
exit
group "WAN"
    family vpn-ipv4
    type internal
    neighbor 192.0.2.4
    exit
    neighbor 192.0.2.5
    exit
exit
no shutdown
-----
```


Figure 160 shows the BGP peering sessions among the PEs and the enabled BGP families. Note that PE-1 and PE-6 only establish an EVPN peering session with their peers (only the EVPN family is enabled on both PEs, even if the peer PEs are VPN-IPv4 capable as well).

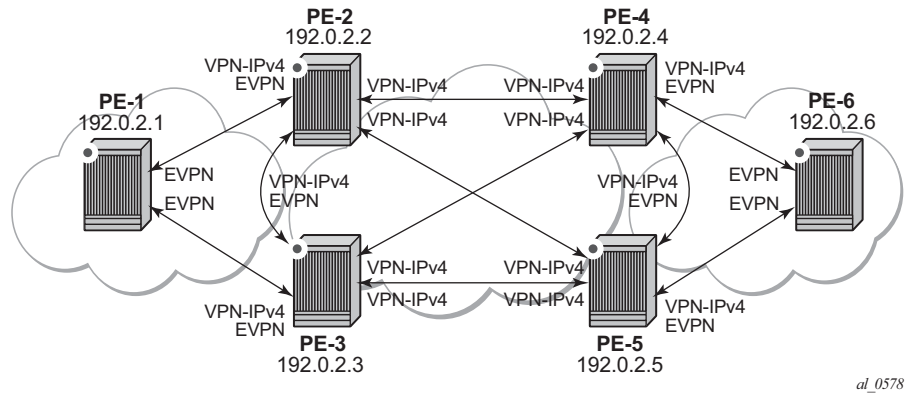


Figure 160: BGP adjacencies and enabled families

Once the network infrastructure is running properly, the actual service configuration, as illustrated in Figure 159, can be carried out. The following CLI output shows the configuration for VPLS 101 and VPRN 10 in PE-1, PE-2 and PE-3. The other overlay network has a similar configuration.

*A:PE-1# configure service vpls 101

```
*A:PE-1>config>service>vpls# info
-----
vxlan vni 101 create
exit
bgp
  route-distinguisher 192.0.2.1:101
  route-target export target:64500:101 import target:64500:101
exit
bgp-evpn
  vxlan
    no shutdown
  exit
exit
proxy-arp
  no shutdown
exit
stp
  shutdown
exit
service-name "evi-101"
sap 1/1/1:101 create
exit
no shutdown
-----
```

EVPN-VXLAN in an R-VPLS Service

```
*A:PE-2# configure service vpls 101
*A:PE-2>config>service>vpls# info
-----
      allow-ip-int-binding
      vxlan vni 101 create
      exit
      bgp
        route-distinguisher 192.0.2.2:101
        route-target export target:64500:101 import target:64500:101
      exit
      bgp-evpn
        vxlan
          no shutdown
        exit
      exit
      proxy-arp
        shutdown
      exit
      stp
        shutdown
      exit
      service-name "evi-101"
      no shutdown
-----

*A:PE-2# configure service vprn 10
*A:PE-2>config>service>vprn# info
-----
      ecmp 2
      route-distinguisher 192.0.2.2:10
      auto-bind mpls
      vrf-target target:64500:10
      interface "int-1" create
        address 172.16.0.2/24
        mac 00:ca:fe:ca:fe:02
        vrrp 1
          backup 172.16.0.254
          priority 254
          ping-reply
          traceroute-reply
          mac 00:ca:fe:ca:fe:54
        exit
        vrrp 2
          backup 172.16.0.253
          ping-reply
          traceroute-reply
          mac 00:ca:fe:ca:fe:53
        exit
        vpls "evi-101"
      exit
      exit
      no shutdown
-----

*A:PE-3# configure service vpls 101
*A:PE-3>config>service>vpls# info
-----
      allow-ip-int-binding
      vxlan vni 101 create
```

```

exit
bgp
    route-distinguisher 192.0.2.3:101
    route-target export target:64500:101 import target:64500:101
exit
bgp-evpn
    vxlan
        no shutdown
    exit
exit
proxy-arp
    shutdown
exit
stp
    shutdown
exit
service-name "evi-101"
no shutdown
-----
*A:PE-3# configure service vprn 10
*A:PE-3>config>service>vprn# info
-----
ecmp 2
route-distinguisher 192.0.2.3:10
auto-bind mpls
vrf-target target:64500:10
interface "int-1" create
    address 172.16.0.3/24
    mac 00:ca:fe:ca:fe:03
    vrrp 1
        backup 172.16.0.254
        ping-reply
        traceroute-reply
        mac 00:ca:fe:ca:fe:54
    exit
    vrrp 2
        backup 172.16.0.253
        priority 254
        ping-reply
        traceroute-reply
        mac 00:ca:fe:ca:fe:53
    exit
vpls "evi-101"
    exit
exit
no shutdown
-----

```

For details about the EVPN and VXLAN configuration on PE-1 VPLS 101, refer to [EVPN for VXLAN Tunnels \(Layer 2\) on page 1033](#). The configuration of VPLS 101 on PE-2 and PE-3 has the following important aspects:

- The **allow-ip-int-binding** command is required so that the R-VPLS can be bound to VPRN 10.
- The **service-name** command is required and the configured name must match the name configured in the VPRN 10 VPLS interface.
- Even though EVPN and VXLAN are properly configured, **proxy-arp** cannot be enabled in VPLS 101. In an R-VPLS with EVPN-VXLAN, proxy-arp is not supported and the VPRN ARP table is used instead. When an EVPN MAC route that includes an IP address is received in an R-VPLS, the MAC-IP pair encoded in the route is added to the VPRN's ARP table, as opposed to the proxy-arp table.

```
*A:PE-2>config>service>vpls# proxy-arp no shutdown
MINOR: SVCNMR #8007 Cannot modify proxy arp - Not supported on routed vpls services
```

When configuring VPRN 10 on PE-2 and PE-3 the following considerations must be taken into account:

- When trying to enable existing VPRN features on interfaces linked to EVPN-VXLAN R-VPLS interfaces, the following commands are not supported:
 - arp-populate.
 - authentication-policy.
 - IPv6 and ingress>v6-routed-override-filter.
- Dynamic routing protocols such as ISIS, RIP or OSPF are not supported.
- In general, no 7x50 control plane generated packets are sent to the egress VXLAN bindings except for ARP, VRRP, ICMP and BFD.
- As depicted in [Figure 159](#) and shown in the CLI excerpts, VRRP can be configured on the VPRN 10 VPLS interfaces to provide default gateway redundancy to the hosts connected to VPLS 101. Note that two VRRP instances are configured so that VPLS 101 upstream traffic can be load-balanced to PE-2 and PE-3. With VRRP on EVPN-VXLAN R-VPLS interfaces:
 - Ping and traceroute reply can be configured and are supported. BFD is also supported to speed up the fault detection.
 - Note that **standby-forwarding**, even if it were configured for VRRP, would not have any effect in this configuration: the standby PE will never see any flooded traffic sent to it, therefore this command is not applicable to this scenario.
- When a VPRN 10 VPLS interface is bound to VPLS 101, EVPN advertises all the IP addresses configured for that VPLS interface as MAC routes with a static MAC indication. For the remote EVPN peers, that means that those MAC addresses linked to

remote IP interfaces are protected. Note that VRRP virtual IP/MACs are also advertised by EVPN as “static” and so protected. In the example of [Figure 159](#), the VPLS 101 FDB in PE-1 shows the IP interface MACs and VRRP MACs as **EvpnS** (Static) as shown in the following output. VPRN 10 VRRP instances and ARP entries for PE-2 are also shown:

```
*A:PE-1# show service id 101 fdb detail
=====
Forwarding Database, Service 101
=====
```

ServId	MAC	Source-Identifier	Type	Last Change
101	00:ca:fe:ca:fe:53	vxlan: 192.0.2.3:101	EvpnS	07/05/14 00:02:16
101	00:ca:fe:ca:fe:54	vxlan: 192.0.2.2:101	EvpnS	07/05/14 00:02:16
101	00:ca:fe:ca:fe:01	vxlan: 192.0.2.1:101	Evpn	07/05/14 00:02:16
101	00:ca:fe:ca:fe:02	vxlan: 192.0.2.2:101	EvpnS	07/05/14 00:02:16
101	00:ca:fe:ca:fe:03	vxlan: 192.0.2.3:101	EvpnS	07/05/14 00:01:54

```
-----
No. of MAC Entries: 5
-----
Legend: L=Learned O=Oam P=Protected-MAC C=Conditional S=Static
=====
*A:PE-2# show router 10 vrrp instance
=====
VRRP Instances
=====
```

Interface Name	VR Id	Own	Adm	State	Base Pri	Msg Int
	IP		Opr	Pol Id	InUse Pri	Inh Int
int-1	1	No	Up	Master	254	1
	IPv4		Up	n/a	254	No
Backup Addr: 172.16.0.254						
int-1	2	No	Up	Backup	100	1
	IPv4		Up	n/a	100	No
Backup Addr: 172.16.0.253						

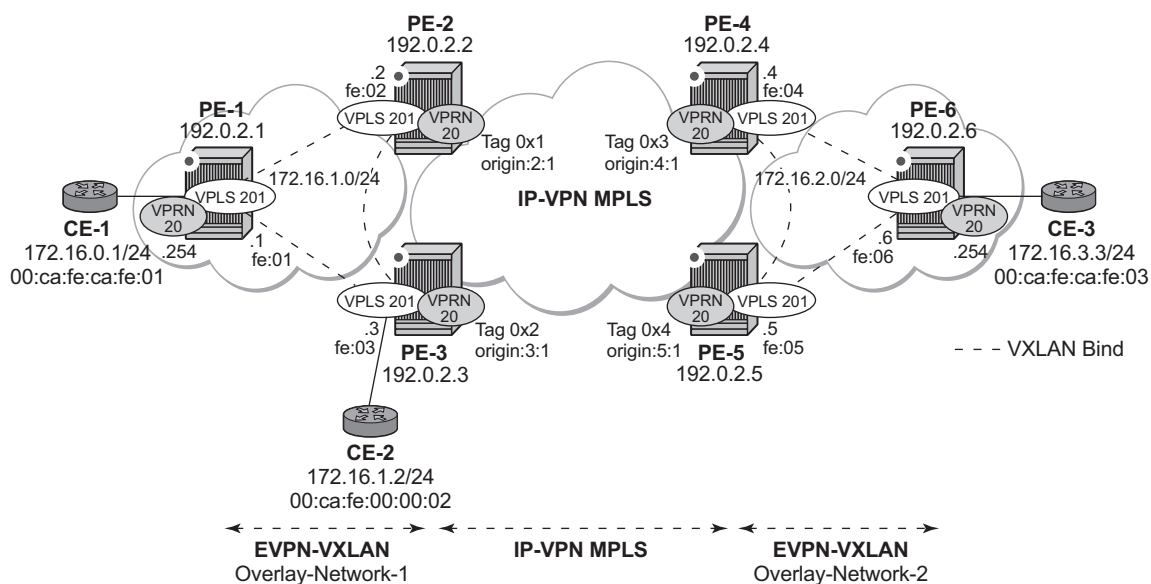
```
-----
Instances : 2
=====
*A:PE-2# show router 10 arp
=====
ARP Table (Service: 10)
=====
```

IP Address	MAC Address	Expiry	Type	Interface
172.16.0.2	00:ca:fe:ca:fe:02	00h00m00s	Oth[I]	int-1
172.16.0.3	00:ca:fe:ca:fe:03	00h00m00s	Evp[I]	int-1
172.16.0.253	00:ca:fe:ca:fe:53	00h00m00s	Oth	int-1
172.16.0.254	00:ca:fe:ca:fe:54	00h00m00s	Oth[I]	int-1

```
-----
No. of ARP Entries: 4
=====
```

EVPN-VXLAN in IRB Backhaul R-VPLS Services

Figure 161 illustrates the second inter-subnet forwarding scenario, where Layer 3 connectivity must be provided not only between the overlay networks but also within each overlay network. In the example depicted in Figure 161, a given customer (tenant) has different subnets and connectivity must be provided across all of them (CE-1, CE-2 and CE-3 must be able to communicate), bearing in mind that EVPN-VXLAN is enabled in each overlay network and IP-VPN MPLS is used to inter-connect both overlay networks. VPLS 201 is an IRB Backhaul R-VPLS service since it provides connectivity to the VPRN instances. Only the two least significant octets of the R-VPLS interface MAC addresses are shown.



al_0579

Figure 161: EVPN-VXLAN for IRB Backhaul R-VPLS Services

From a BGP peering perspective, there is no change in this scenario compared to the previous one: PE-1 and PE-6 only support the EVPN address family. However in this scenario CE-1 is not connected to an R-VPLS directly linked to the VPRN instances in PE-2/PE-3. As a result of that, IP prefixes must be exchanged between PE-1 and PE-2/PE-3. EVPN is able to advertise not only MAC routes and Inclusive Multicast routes, but also IP prefix routes that contain IP prefixes that can be installed in the attached VPRN routing table.

As an example, the VPRN 20 and VPLS 201 configurations on PE-1, PE-2 and PE-3 are shown below. Similar configurations are needed in PE-3, PE-4 and PE-6.

```

*A:PE-1# configure service vprn 20
*A:PE-1>config>service>vprn# info
-----
route-distinguisher 192.0.2.1:20
vrf-target target:64500:20
interface "int-evi-201" create
    address 172.16.1.1/24
    vpls "evi-201"
    exit
exit
interface "int-PE-1-CE-1" create
    address 172.16.0.254/24
    sap 1/1/1:20 create
    exit
exit
no shutdown
-----

*A:PE-1# configure service vpls 201
*A:PE-1>config>service>vpls# info
-----
allow-ip-int-binding
vxlan vni 201 create
exit
bgp
    route-distinguisher 192.0.2.1:201
    route-target export target:64500:201 import target:64500:201
exit
bgp-evpn
    ip-route-advertisement
    vxlan
        no shutdown
    exit
exit
stp
    shutdown
exit
service-name "evi-201"
no shutdown
-----

*A:PE-2# configure service vprn 20
*A:PE-2>config>service>vprn# info
-----
route-distinguisher 192.0.2.2:20
auto-bind mpls
vrf-target target:64500:20
interface "int-evi-201" create
    address 172.16.1.2/24
    vpls "evi-201"
    exit
exit
no shutdown
-----

*A:PE-2# configure service vpls 201
*A:PE-2>config>service>vpls# info
-----
allow-ip-int-binding

```

```

vxlan vni 201 create
exit
bgp
    route-distinguisher 192.0.2.2:201
    route-target export target:64500:201 import target:64500:201
exit
bgp-evpn
    ip-route-advertisement
    vxlan
        no shutdown
    exit
exit
stp
    shutdown
exit
service-name "evi-201"
no shutdown
-----

*A:PE-3# configure service vprn 20
*A:PE-3>config>service>vprn# info
-----
    route-distinguisher 192.0.2.3:20
    auto-bind mpls
    vrf-target target:64500:20
    interface "int-evi-201" create
        address 172.16.1.3/24
        vpls "evi-201"
    exit
exit
no shutdown
-----

*A:PE-3# configure service vpls 201
*A:PE-3>config>service>vpls# info
-----
    allow-ip-int-binding
    vxlan vni 201 create
    exit
    bgp
        route-distinguisher 192.0.2.3:201
        route-target export target:64500:201 import target:64500:201
    exit
    bgp-evpn
        ip-route-advertisement
        vxlan
            no shutdown
        exit
    exit
    stp
        shutdown
    exit
    service-name "evi-201"
    sap 1/1/1:20 create
    exit
    no shutdown
    -----

```


As shown in the CLI excerpt, the configuration in the three nodes (PE-1/2/3) for VPLS 201 and VPRN 20 is very similar. The main difference is the **auto-bind mpls** command existing in PE-2/3's VPRN 20. This command allows the VPRN 20 on PE-2/3 to receive IP-VPN routes from the core and resolve them to MPLS tunnels. VPRN 20 on PE-1 does not require such command since all its IP prefixes are resolved to local interfaces or to EVPN peers.

The **ip-route-advertisement** command enables:

- The advertisement of IP prefixes in EVPN, in routes type 5. All the existing IP prefixes in the attached VPRN 20 routing table are advertised in EVPN within the VPLS 201 context (except for the ones associated to VPLS 201 itself).
- The installation of IP prefixes in the attached VPRN 20 routing table with a preference of 169 (bgp-vpn routes for IP-VPN have a preference of 170) and a next-hop of the GW-IP (Gateway IP) address included in the EVPN IP prefix route.

For instance, the following output shows that PE-1 advertises the IP prefix 172.16.0.0/24 as a EVPN route to PE-3 (similar route is sent to PE-2), captured by a **debug router bgp update** session. The VPRN 20 routing tables in PE-1, PE-2 and PE-3 are also shown.

```
4 2014/07/05 23:58:54.88 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 89
  Flag: 0x90 Type: 14 Len: 45 Multiprotocol Reachable NLRI:
    Address Family EVPN
    NextHop len 4 NextHop 192.0.2.1
    Type: EVPN-IP-Prefix Len: 34 RD: 192.0.2.1:201, tag: 201, ip_prefix: 17
2.16.0.0/24 gw_ip 172.16.1.1 Label: 0
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 16 Extended Community:
    target:64500:201
    bgp-tunnel-encap:VXLAN
"
```

```
*A:PE-1# show router 20 route-table
```

```
=====
Route Table (Service: 20)
=====
```

Dest Prefix[Flags]	Type	Proto	Age	Pref
Next Hop[Interface Name]			Metric	
172.16.0.0/24	Local	Local	23h57m35s	0
int-PE-1-CE-1			0	
172.16.1.0/24	Local	Local	23h57m48s	0
int-evi-201			0	
172.16.2.0/24	Remote	BGP EVPN	00h00m17s	169
172.16.1.2			0	
172.16.3.0/24	Remote	BGP EVPN	00h00m17s	169
172.16.1.2			0	

```
-----
```

EVPN-VXLAN in IRB Backhaul R-VPLS Services

```
No. of Routes: 4
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====

*A:PE-2# show router 20 route-table
=====
Route Table (Service: 20)
=====
Dest Prefix[Flags]                Type    Proto    Age          Pref
Next Hop[Interface Name]          Metric
-----
172.16.0.0/24                    Remote  BGP EVPN  00h11m04s    169
      172.16.1.1                  0
172.16.1.0/24                    Local   Local    01d00h08m    0
      int-evi-201                  0
172.16.2.0/24                    Remote  BGP VPN   01d00h07m    170
      192.0.2.4 (tunneled)         0
172.16.3.0/24                    Remote  BGP VPN   01d00h07m    170
      192.0.2.4 (tunneled)         0
-----

No. of Routes: 4
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====

*A:PE-3# show router 20 route-table
=====
Route Table (Service: 20)
=====
Dest Prefix[Flags]                Type    Proto    Age          Pref
Next Hop[Interface Name]          Metric
-----
172.16.0.0/24                    Remote  BGP EVPN  00h11m23s    169
      172.16.1.1                  0
172.16.1.0/24                    Local   Local    01d00h09m    0
      int-evi-201                  0
172.16.2.0/24                    Remote  BGP VPN   01d00h08m    170
      192.0.2.4 (tunneled)         0
172.16.3.0/24                    Remote  BGP VPN   01d00h08m    170
      192.0.2.4 (tunneled)         0
-----

No. of Routes: 4
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====
```

When checking the operation of EVPN in this scenario, it is important to observe that the right next hops and prefixes are successfully installed in the VPRN 20 routing table:

- EVPN IP prefixes are sent using a GW-IP matching the primary IP interface address of the R-VPLS for which the routes are sent. For instance, as shown above, IP prefix 172.16.0.0/24 is advertised from PE-1 with GW-IP 172.16.1.1 (the IP address configured for the VPRN 20 VPLS interface in PE-1). In the PE-2/3 VPRN 20 routing tables, IP prefix 172.16.0.0/24 is installed with next hop 172.16.1.1. Traffic arriving at PE-2/3 on VPRN 20 with IP Destination Address (DA) in the 172.16.0.0 subnet matches the mentioned routing table entry. As usual, the next hop is resolved by the ARP table to a MAC and the MAC resolved by the FDB table to an egress VTEP, VNI.
- IP prefixes in the VPRN 20 routing table are advertised in IP-VPN to the remote IP-VPN MPLS peers. Received IP-VPN prefixes are installed in the VPRN 20 routing table using the remote PE system IP address as the next hop, as usual. For instance, 172.16.3.0/24 is installed in PE-2 VPRN 20's routing table with next hop (tunneled) 192.0.2.4 and preference 170.

The following considerations of how the routing table manager (RTM) handles EVPN and IP-VPN prefixes must be taken into account:

- Only VPRN interface primary addresses are advertised as GW-IP in EVPN IP prefix routes. Secondary addresses are never sent as GW-IP addresses.
- EVPN IP prefixes are advertised by default as soon as the **ip-route-advertisement** command is enabled and there are active IP prefixes in the attached VPRN routing table.
- If the same IP prefix is received on a PE via EVPN and IP-VPN at the same time for the same VPRN, by default the EVPN prefix is selected since its preference (169) is better than the IP-VPN preference (170).
- Since EVPN has a better preference compared to IP-VPN, when the VPRNs on redundant PEs are attached to the same R-VPLS service, routing loops may occur. The use case described here is an example where routing loops can occur. Check [Use of Routing Policies to Avoid Routing Loops in Redundant PEs on page 1092](#) to avoid routing loops in redundant PEs for more information.
- When the command **ip-route-advertisement** is enabled, the subnet IP prefixes are advertised in EVPN but not the "host" IP prefixes (/32 prefixes associated with the local interfaces). If the user wants to advertise the host IP prefixes as well, the **incl-host** keyword must be added to the **ip-route-advertisement** command. The following example illustrates this. The host routes can be shown with the **show router route-table all** command. When the **incl-host** keyword is added to PE-1's VPLS 201, PE-1 advertises the host routes as well and these are installed in the remote PEs' routing tables.

A:PE-1# show router 20 route-table

Route Table (Service: 20)

Dest Prefix[Flags] Next Hop[Interface Name]	Type	Proto	Age Metric	Pref
172.16.0.0/24 int-PE-1-CE-1	Local	Local	01d04h17m 0	0
172.16.1.0/24 int-evi-201	Local	Local	01d04h18m 0	0
172.16.2.0/24 172.16.1.2	Remote	BGP EVPN	04h20m31s 0	169
172.16.3.0/24 172.16.1.2	Remote	BGP EVPN	04h20m31s 0	169

No. of Routes: 4

Flags: n = Number of times nexthop is repeated

B = BGP backup route available

L = LFA nexthop available

S = Sticky ECMP requested

A:PE-1# show router 20 route-table all

Route Table (Service: 20)

Dest Prefix[Flags] Next Hop[Interface Name]	Type	Proto Active	Age Metric	Pref
172.16.0.0/24 int-PE-1-CE-1	Local	Local Y	01d04h17m 0	0
172.16.0.254/32 int-PE-1-CE-1	Local	Host Y	01d04h17m 0	0
172.16.1.0/24 int-evi-201	Local	Local Y	01d04h18m 0	0
172.16.1.1/32 int-evi-201	Local	Host Y	01d04h18m 0	0
172.16.2.0/24 172.16.1.2	Remote	BGP EVPN Y	04h20m34s 0	169
172.16.3.0/24 172.16.1.2	Remote	BGP EVPN Y	04h20m34s 0	169

No. of Routes: 6

Flags: n = Number of times nexthop is repeated

B = BGP backup route available

L = LFA nexthop available

S = Sticky ECMP requested

E = Inactive best-external BGP route

A:PE-1# configure service vpls 201 bgp-evpn ip-route-advertisement incl-host

A:PE-2# show router 20 route-table

Route Table (Service: 20)

Dest Prefix[Flags] Next Hop[Interface Name]	Type	Proto	Age Metric	Pref
--	------	-------	---------------	------

```

172.16.0.0/24 Remote BGP EVPN 04h25m22s 169
    172.16.1.1 0
172.16.0.254/32 Remote BGP EVPN 00h03m52s 169
    172.16.1.1 0
172.16.1.0/24 Local Local 01d04h22m 0
    int-evi-201 0
172.16.2.0/24 Remote BGP VPN 01d04h22m 170
    192.0.2.4 (tunneled) 0
172.16.3.0/24 Remote BGP VPN 01d04h22m 170
    192.0.2.4 (tunneled) 0
-----

```

No. of Routes: 5

Flags: n = Number of times nexthop is repeated

B = BGP backup route available

L = LFA nexthop available

S = Sticky ECMP requested

- ECMP is fully supported for the VPRN for EVPN IP prefix routes coming from different GW-IP next-hops. However ECMP is not supported for IP prefixes routes belonging to different owners (EVPN and IP-VPN). ECMP behavior for EVPN is illustrated in the following output. When **ecmp** is enabled in PE-1's VPRN 20, an additional route with a different GW-IP as next-hop is installed in the routing table for the IP-prefixes 172.16.2.0/24 and 172.16.3.0/24.

```
*A:PE-1# configure service vprn 20 ecmp 2
```

```
*A:PE-1# show router 20 route-table
```

```
Route Table (Service: 20)
```

```

=====
Dest Prefix[Flags]      Type  Proto  Age      Pref
Next Hop[Interface Name] Metric
-----
172.16.0.0/24           Local  Local  01d04h50m 0
    int-PE-1-CE-1      0
172.16.1.0/24           Local  Local  01d04h50m 0
    int-evi-201        0
172.16.2.0/24           Remote BGP EVPN 00h00m01s 169
    172.16.1.2         0
172.16.2.0/24           Remote BGP EVPN 00h00m01s 169
    172.16.1.3         0
172.16.3.0/24           Remote BGP EVPN 00h00m01s 169
    172.16.1.2         0
172.16.3.0/24           Remote BGP EVPN 00h00m01s 169
    172.16.1.3         0
-----

```

No. of Routes: 6

Flags: n = Number of times nexthop is repeated

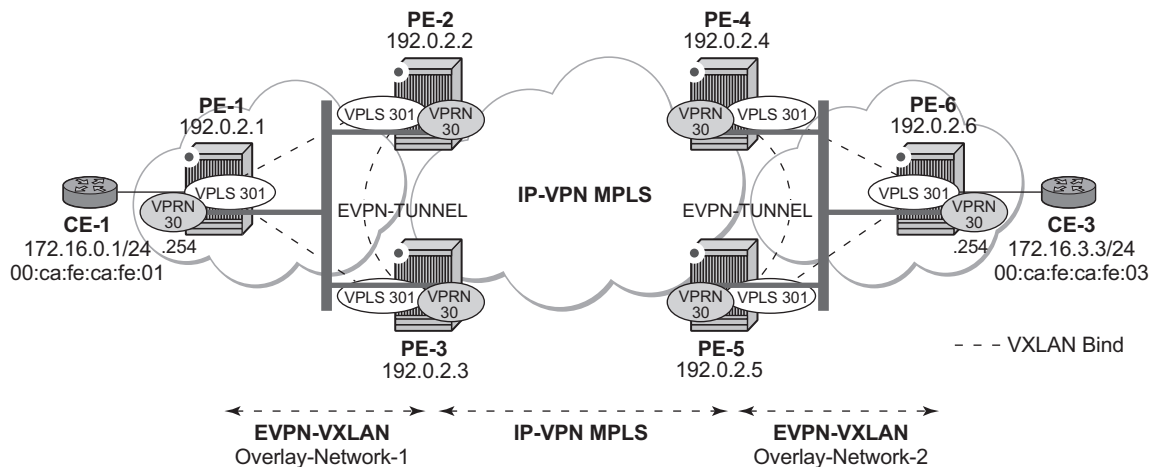
B = BGP backup route available

L = LFA nexthop available

S = Sticky ECMP requested

EVPN-VXLAN in EVPN Tunnel R-VPLS Services

The previous scenario shows how to use EVPN-VXLAN to provide inter-subnet forwarding for a given tenant, where R-VPLS services can contain hosts and also offer transit services between VPRN instances. For example, in the use case depicted in [Figure 161](#), VPLS 201 in Overlay-Network-1 is an R-VPLS that can provide intra-subnet connectivity to all the hosts in subnet 172.16.1.0/24 (for example, CE-2 belongs to this subnet) but it can also provide “transit” or “backhaul” connectivity to hosts in subnet 172.16.0.0/24 (for example, CE-1) sending packets to subnets 172.16.2.0/24 or 172.16.3.0/24. In some cases, the R-VPLS where EVPN-VXLAN is enabled does not need to provide intra-subnet connectivity and it is purely a transit or backhaul service where VPRN IRB interfaces are connected. [Figure 162](#) illustrates this use case.



al_0581

Figure 162: EVPN-VXLAN in EVPN-tunnel R-VPLS Services

Compared to the use case in [Figure 161](#), in this case the R-VPLS connecting the IRB interfaces in Overlay-Network-1 (VPLS 301) does not have any connected host. If that is the case, VPLS 301 can be configured as an EVPN tunnel.

EVPN tunnels are enabled using the `evpn-tunnel` command under the R-VPLS interface configured on the VPRN. EVPN tunnels bring the following benefits to EVPN-VXLAN IRB backhaul R-VPLS services:

- Easier and simpler provisioning of the tenant service: if an EVPN tunnel is configured in an IRB backhaul R-VPLS there is no need to provision the IRB IP addresses in the VPRN. This makes the provisioning easier to automate and saves IP addresses from the tenant IP space.
- Higher scalability of the IRB backhaul R-VPLS: if EVPN tunnels are enabled, BUM traffic is suppressed in the EVPN-VXLAN IRB backhaul R-VPLS service (it is not

required). As a result, the number of VXLAN bindings in IRB backhaul R-VPLS services with EVPN tunnels can be much higher.

As an example, the VPRN 30 and VPLS 301 configurations on PE-1, PE-2 and PE-3 are shown below. Note that similar configurations are needed in PE-4, PE-5 and PE-6.

```
A:PE-1# configure service vprn 30
```

```
A:PE-1>config>service>vprn# info
```

```
-----
route-distinguisher 192.0.2.1:30
vrf-target target:64500:30
interface "int-PE-1-CE-1" create
    address 172.16.0.254/24
    sap 1/1/1:30 create
    exit
exit
interface "int-evi-301" create
    vpls "evi-301"
        evpn-tunnel
    exit
exit
no shutdown
-----
```

```
A:PE-1# configure service vpls 301
```

```
A:PE-1>config>service>vpls# info
```

```
-----
allow-ip-int-binding
vxlan vni 301 create
exit
bgp
    route-distinguisher 192.0.2.1:301
    route-target export target:64500:301 import target:64500:301
exit
bgp-evpn
    ip-route-advertisement
    vxlan
        no shutdown
    exit
exit
stp
    shutdown
exit
service-name "evi-301"
no shutdown
-----
```

```
A:PE-2# configure service vprn 30
```

```
A:PE-2>config>service>vprn# info
```

```
-----
route-distinguisher 192.0.2.2:30
auto-bind mpls
vrf-target target:64500:30
interface "int-evi-301" create
    vpls "evi-301"
        evpn-tunnel
    exit
exit
no shutdown
-----
```

EVPN-VXLAN in EVPN Tunnel R-VPLS Services

```
-----
A:PE-2# configure service vpls 301
A:PE-2>config>service>vpls# info
-----
        allow-ip-int-binding
        vxlan vni 301 create
        exit
        bgp
            route-distinguisher 192.0.2.2:301
            route-target export target:64500:301 import target:64500:301
        exit
        bgp-evpn
            ip-route-advertisement
            vxlan
                no shutdown
            exit
        exit
        stp
            shutdown
        exit
        service-name "evi-301"
        no shutdown
-----

A:PE-3# configure service vprn 30
A:PE-3>config>service>vprn# info
-----
        route-distinguisher 192.0.2.3:30
        auto-bind mpls
        vrf-target target:64500:30
        interface "int-evi-301" create
            vpls "evi-301"
            evpn-tunnel
        exit
        exit
        no shutdown
-----

A:PE-3# configure service vpls 301
A:PE-3>config>service>vpls# info
-----
        allow-ip-int-binding
        vxlan vni 301 create
        exit
        bgp
            route-distinguisher 192.0.2.3:301
            route-target export target:64500:301 import target:64500:301
        exit
        bgp-evpn
            ip-route-advertisement
            vxlan
                no shutdown
            exit
        exit
        stp
            shutdown
        exit
        service-name "evi-301"
        no shutdown
-----
```


As shown in the output above, the configuration in the three nodes (PE-1/2/3) for VPLS 301 and VPRN 30 is similar to the configuration of VPLS 201 and VPRN 20 in the previous scenario, however, when the **evpn-tunnel** command is added to the VPRN interface, there is no need to configure an IP interface address. Note that **evpn-tunnel** can be enabled independently of **ip-route-advertisement** (although no route-type 5 advertisements are sent in that case).

A given VPRN supports regular IRB backhaul R-VPLS services as well as EVPN tunnel R-VPLS services. A maximum of eight R-VPLS services with **ip-route-advertisement** enabled per VPRN is supported (in any combination of regular IRB R-VPLS or EVPN tunnel R-VPLS services). Note that EVPN tunnel R-VPLS services do not support SAPs or SDP-binds. No frames are flooded in an EVPN tunnel R-VPLS service, and, in fact no inclusive multicast routes are exchanged in R-VPLS services that are configured as EVPN tunnels. The show service id vxlan command for an R-VPLS service configured as an EVPN tunnel shows <egress VTEP, VNI> bindings excluded from the “multicast list”, in other words, the VXLAN bindings are not used to flood BUM traffic:

```
*A:PE-2# show service id 301 vxlan
=====
VPLS VXLAN, Ingress VXLAN Network Id: 301

=====
Egress VTEP, VNI
=====
VTEP Address          Egress VNI      Num. MACs      In Mcast List?  Oper State
-----
192.0.2.1              301              1               No               Up
192.0.2.3              301              1               No               Up
-----
Number of Egress VTEP, VNI : 2
=====
```

The process followed upon receiving a route-type 5 on a regular IRB R-VPLS interface (previous scenario) differs from the one for an EVPN tunnel type (this scenario):

- IRB backhaul R-VPLS VPRN interface:
 - When a route-type 2 that includes an IP address is received and it becomes active, the MAC/IP information is added to the FDB and ARP tables. This can be checked with the **show>router>arp** command and the **show>service>id>fdb detail** command.
 - When a route-type 5 is received on (for instance) PE-2, and becomes active for the R-VPLS service, the IP prefix is added to the VPRN routing table regardless of the existence of a route-type 2 that can resolve the GW IP address. If a packet is received from the WAN side and the IP lookup hits an entry for which the GW IP (IP next-hop) does not have an active ARP entry, the system will ARP to get the MAC. If the ARP is resolved but the MAC is unknown in the FDB table, the system will flood the ARP message into the R-VPLS multicast list. Routes type 5 can be checked in the routing table with the **show>router>route-table** command and the **show>router>fib** command.

- EVPN tunnel R-VPLS VPRN interface:
 - When a route-type 2 is received and becomes active, the MAC address is added to the FDB (only). This MAC address is normally a GW-MAC.
 - When a route-type 5 is received on (for instance) PE-1, the system looks for the GW-MAC. The IP prefix is added to the VPRN routing table with next hop equal to EVPN-tunnel-GW-MAC; for example (see below), ET-d8:45:ff:00:00:6a is an EVPN tunnel with GW-MAC d8:45:ff:00:00:6a. The GW-MAC is added from the GW-MAC extended community sent along with the route-type 5 for prefix 172.16.3.0/24. If a packet is received from the CE-1 and the IP lookup hits an entry for which the next hop is a EVPN tunnel:GW-MAC, the system looks up the GW-MAC in the FDB. Normally a route-type 2 with the GW-MAC has already been received so that the GW-MAC has been added to the FDB. If the GW-MAC is not present in the FDB, the packet will be dropped.
 - Note that the IP prefixes with GW-MACs as next hops are displayed in the show router route-table command, as shown below for the setup in Figure 4. The show service id fdb detail command can be used to look for the forwarding information for a given GW-MAC:

```
A:PE-1# show router 30 route-table
=====
Route Table (Service: 30)
=====
Dest Prefix[Flags]                                Type   Proto   Age           Pref
Next Hop[Interface Name]                        Metric
-----
172.16.0.0/24                                     Local  Local   00h06m15s    0
int-PE-1-CE-1                                     0
172.16.3.0/24                                     Remote BGP EVPN 00h05m31s    169
int-evi-301 (ET-d8:45:ff:00:00:6a)               0
-----
No. of Routes: 2
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====
A:PE-1# show service id 301 fdb detail
=====
Forwarding Database, Service 301
=====
ServId  MAC                Source-Identifier  Type   Last Change
Age
-----
301      d8:45:ff:00:00:6a vxlan:            EvpnS  07/05/14 00:02:46
192.0.2.2:301
301      d8:47:ff:00:00:6a cpm              Intf   07/05/14 00:01:48
301      d8:48:ff:00:00:6a vxlan:            EvpnS  07/05/14 00:02:18
192.0.2.3:301
-----
No. of MAC Entries: 3
-----
Legend: L=Learned O=Oam P=Protected-MAC C=Conditional S=Static
```

Note that IP prefix routes sent for EVPN tunnel R-VPLS services do not contain a GW-IP (the GW-IP will be zero) but convey a GW-MAC address that is used in the peer VPRN routing table. The following output shows PE-2's VPRN 30 interface MAC address and the route-type 5 sent to PE-1 using the MAC as GW-MAC:

```

=====
*A:PE-2# show router 30 interface detail | match MAC
MAC Address      : d8:45:ff:00:00:6a    Mac Accounting    : Disabled

*A:PE-2# configure service vpls 301 bgp-evpn ip-route-advertisement
*A:PE-2#
6 2014/07/05 00:29:41.79 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.1
"Peer 1: 192.0.2.1: UPDATE
Peer 1: 192.0.2.1 - Send BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 105
    Flag: 0x90 Type: 14 Len: 45 Multiprotocol Reachable NLRI:
        Address Family EVPN
        NextHop len 4 NextHop 192.0.2.2
        Type: EVPN-IP-Prefix Len: 34 RD: 192.0.2.2:301, tag: 301, ip_prefix: 17
2.16.3.0/24 gw_ip 0.0.0.0 Label: 0
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 32 Extended Community:
        origin:69:1
        target:64500:301
        mac-nh:d8:45:ff:00:00:6a
        bgp-tunnel-encap:VXLAN
"

```

Looking at the VPRN 30 routing table, since IP prefixes are shown with an EVPN tunnel next-hop (GW-MAC) as opposed to an IP next-hop, the user may think that no ARP entries are consumed by VPRN 30. However internal ARP entries are still consumed in VPRN 30. Although not shown in the show router 30 arp command, the **summary** option shows the consumption of internal ARP entries for EVPN.

```

*A:PE-2# show router 30 route-table
=====
Route Table (Service: 30)
=====
Dest Prefix[Flags]                                Type    Proto    Age          Pref
      Next Hop[Interface Name]                      Metric
-----
172.16.0.0/24                                     Remote  BGP EVPN   00h31m34s    169
      int-evi-301 (ET-d8:47:ff:00:00:6a)              0
172.16.3.0/24                                     Remote  BGP VPN    00h59m09s    170
      192.0.2.4 (tunneled)                             0
-----
No. of Routes: 2
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available

```

EVPN-VXLAN in EVPN Tunnel R-VPLS Services

```

      L = LFA nexthop available
      S = Sticky ECMP requested
=====
*A:PE-2# show router 30 arp

=====
ARP Table (Service: 30)
=====
IP Address      MAC Address      Expiry   Type   Interface
-----
No Matching Entries Found
=====
*A:PE-2# show router 30 arp summary

=====
ARP Table Summary (Service: 30)
=====
Local ARP Entries      : 1
Static ARP Entries     : 0
Dynamic ARP Entries    : 0
Managed ARP Entries   : 0
Internal ARP Entries   : 0
BGP-EVPN ARP Entries : 1
-----
No. of ARP Entries     : 2
=====
```

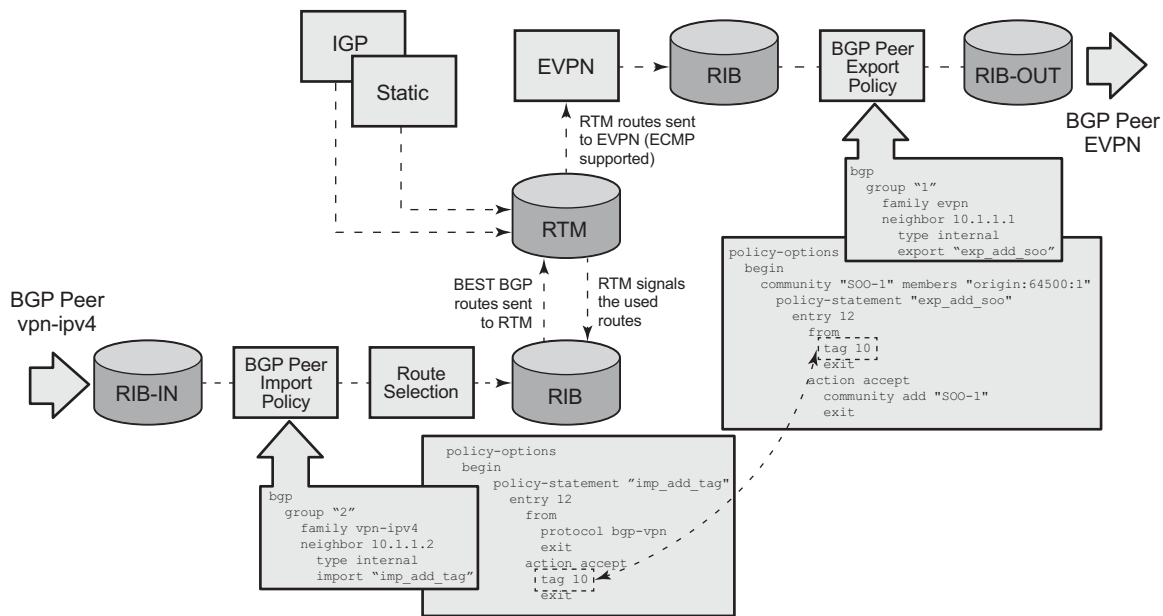
The number of BGP-EVPN ARP Entries in the **show router 30 arp summary** command matches the number of remote valid GW-MACs for VPRN 30.

Routing Policies for IP Prefixes in EVPN

Routing policies are supported for IP prefixes imported/exported through BGP EVPN. The default import/export behavior for IP prefixes in EVPN can be modified by the use of routing policies applied either at peer level (**config>router>bgp>group/group>neighbor>import/export**) or VPLS level (**config>service>vpls>bgp>vsi-import/vsi-export**).

When applying routing policies to control the distribution of prefixes between EVPN and IP-VPN, the user must take into account that both families are completely separated as far as BGP is concerned and that when prefixes from a family are imported in the RTM, the BGP attributes are lost to the other family. The use of tags allows the controlled distribution of prefixes across the two families.

Figure 163 illustrates how vpn-ipv4 routes are imported into the RTM and then passed onto EVPN for its own processing. Note that vpn-ipv4 routes can be tagged at ingress and this tag is preserved throughout the RTM and EVPN processing so that the tag can be “matched” by the egress BGP routing policy. In this particular example, egress EVPN routes matching tag 10, are modified to add a site-of-origin community origin:64500:1.



al_0583

Figure 163: Routing Policies for Egress EVPN Routes

Policy TAGS can be used to match EVPN IP-prefixes that were learned not only from BGP vpn-ipv4 but also from other routing protocols. Note that the tag range supported for each protocol is different:

```
<tag> : accepts in decimal or hex
        [0x1..0xFFFFFFFF]H (for OSPF and ISIS)
        [0x1..0xFFFF]H (for RIP)
        [0x1..0xFF]H (for BGP)
```

Figure 164 illustrates the reverse workflow: routes imported from EVPN and exported from RTM to BGP vpn-ipv4. In this example, EVPN routes received with community VM-mob are tagged with TAG 200. At the egress vpn-ipv4 peers, only the routes with TAG 200 are advertised.

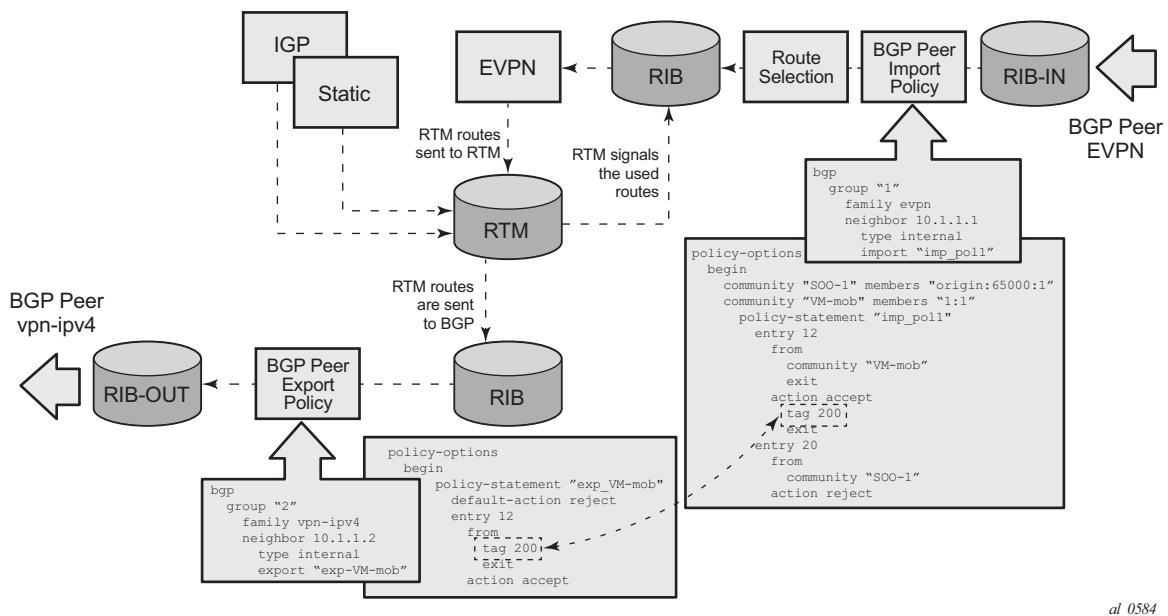


Figure 164: Routing Policies for Ingress EVPN Routes

The above behavior and the use of tags is also valid for **vsi-import** and **vsi-export** policies. The behavior can be summarized in the following statements:

- For EVPN prefix routes received and imported in RTM:
 - Routes can be matched on communities and tags can be added to them. This works at peer level or vsi-import level.
 - Well-known communities (no-export|no-export-subconfed|no-advertise) also require that the routing policies add a tag if the user wants to modify the behavior when exporting to BGP.
 - Routes can be matched based on family evpn.
 - Routes cannot be matched on prefix-list.
- For exporting RTM to EVPN prefix routes:
 - Routes can be matched on tags and based on that, communities added, or routes accepted or rejected, etc. This works at peer level or vsi-export level.
 - Tags can be added for static-routes, rip, ospf, isis and bgp and then be matched in the vsi-export policy for EVPN IP-prefix route advertisement.
 - Tags cannot be added for direct routes.

Use of Routing Policies to Avoid Routing Loops in Redundant PEs

When redundant PE VPRN instances are connected to the same R-VPLS service (IRB backhaul or EVPN tunnel R-VPLS) with the `ip-route-advertisement` command enabled, routing loops can occur in two different use-cases:

1. Routing loop caused by EVPN and IP-VPN interaction in the RTM.
2. Routing loop caused by EVPN in “parallel” R-VPLS services.

Policy configuration examples for both cases are provided below.

Routing loop use-case 1: EVPN and IP-VPN interaction

This use case refers to scenarios with redundant PEs and VPRNs connected to the same R-VPLS with **ip-route-advertisement**. The scenarios in [Figure 161](#) (EVPN-VXLAN for IRB Backhaul R-VPLS services) and [Figure 162](#) (EVPN-VXLAN in EVPN tunnel R-VPLS services) are examples of this use case. In both scenarios the following process causes a routing loop:

1. IP prefix 172.16.3.0/24 is advertised by PE-4 to PE-2 and PE-3.
2. PE-2 imports that prefix in the VPRN routing table and re-advertises the IP prefix in EVPN to PE-1 and PE-3 (the same thing happens in PE-3).
3. PE-3 already has the 172.16.3.0/24 prefix in the VPRN routing table with preference 170 (IP-VPN) but since it receives the IP prefix from EVPN with lower preference (169), the RTM will install the EVPN prefix in the VPRN routing table (the same thing happens in PE-2).
4. PE-3 advertises the EVPN learned IP prefix to all MP-BGP vpn-ipv4 peers (also PE-2).
5. PE-2 receives the IP prefix again from PE-3 and will advertise it in EVPN again, creating a routing loop (PE-3 will do the same thing as well).

This routing loop also happens in traditional multi-homed IP-VPN scenarios where the PE-CE eBGP and MP-BGP vpn-ipv4/v6 protocols interact in the same VPRN RTM, with different router preferences. In either case (EVPN or eBGP interaction with MP-BGP) the issue can be solved by the use of routing policies and site-of-origin communities.

Routing policies are applied to PE-2 and PE-3 (also to PE-4 and PE-5) and allow the redundant PEs to reject their own generated routes in order to avoid the loops. These routing policies can be applied at vsi-import/export level or BGP group/neighbor level. The following output shows an example of routing policies applied at BGP neighbor level for PE-2 (similar policies are applied on PE-3/4/5). Note that neighbor or group level policies are the preferred way in this kind of use case: a single set of policies is sufficient, as opposed to a set of policies per service (if the policies are applied at vsi-import/export level).


```

*A:PE-2>config>router>bgp# info
-----
vpn-apply-import
vpn-apply-export
min-route-advertisement 1
enable-peer-tracking
rapid-withdrawal
rapid-update evpn
group "DC"
    family vpn-ipv4 evpn
    type internal
    neighbor 192.0.2.1
        import "add-tag_to_bgp-evpn_routes"
    exit
    neighbor 192.0.2.3
        import "reject_based_on_SOO"
        export "add-SOO_on_export"
    exit
exit
group "WAN"
    family vpn-ipv4
    type internal
    neighbor 192.0.2.4
        import "add-tag_to_bgp-vpn_routes"
    exit
    neighbor 192.0.2.5
        import "add-tag_to_bgp-vpn_routes"
    exit
exit
no shutdown
-----

*A:PE-2>config>router>policy-options# info
-----
community "SOO-PE-2" members "origin:2:1"
community "SOO-PE-3" members "origin:3:1"
policy-statement "add-SOO_on_export"
    entry 10
        from
            tag 0x1
        exit
        action accept
            community add "SOO-PE-2"
        exit
    exit
    entry 20
        from
            tag 0x2
        exit
        action accept
            community add "SOO-PE-3"
        exit
    exit
exit
policy-statement "reject_based_on_SOO"
    entry 10
        from
            community "SOO-PE-2"
        exit
        action reject
    
```

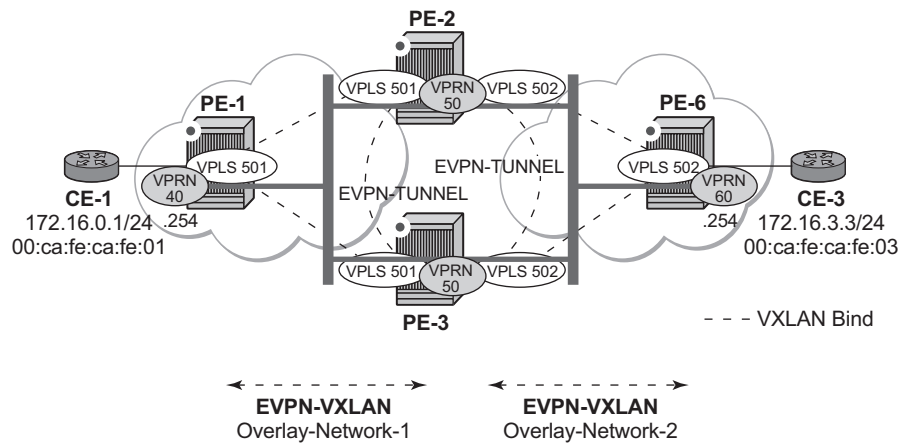
Use of Routing Policies to Avoid Routing Loops in Redundant PEs

```
exit
entry 20
  from
    community "SOO-PE-3"
  exit
  action reject
exit
exit
policy-statement "add-tag_to_bgp-vpn_routes"
  entry 10
    from
      protocol bgp-vpn
    exit
    action accept
      tag 0x1
    exit
  exit
exit
policy-statement "add-tag_to_bgp-evpn_routes"
  entry 10
    from
      family evpn
    exit
    action accept
      tag 0x1
    exit
  exit
exit
exit
```

EVPN and MP-BGP routes are tagged at import and add a site-of-origin community. Routes exchanged between the two redundant PEs are rejected if they are received by a PE with its own site-of-origin.

Routing loop use-case 2: EVPN in parallel R-VPLS services

If a given VPRN is connected to more than one R-VPLS with **ip-route-advertisement** enabled, IP prefixes that belong to one R-VPLS are advertised into the other R-VPLS and vice versa. When redundant PEs are used, a routing loop will occur. [Figure 165](#) illustrates this use case. Note that the example shows R-VPLS with an EVPN tunnel configuration but the same routing loop occurs for regular IRB backhaul R-VPLS services.



al_0585

Figure 165: EVPN in Parallel R-VPLS Services

The configuration of VPRN 50 as well as VPLS 501/502 and the required policies are shown below. For this use case, policies must be applied at vsi-import/export level since more granularity is required when modifying the imported/exported routes.

```
*A:PE-2# configure service vprn 50
*A:PE-2>config>service>vprn# info
-----
route-distinguisher 192.0.2.2:50
interface "int-evi-501" create
    vpls "evi-501"
    evpn-tunnel
    exit
exit
interface "int-evi-502" create
    vpls "evi-502"
    evpn-tunnel
    exit
exit
no shutdown
-----
*A:PE-2# configure service vpls 501
*A:PE-2>config>service>vpls# info
-----
```

Use of Routing Policies to Avoid Routing Loops in Redundant PEs

```
allow-ip-int-binding
vxlan vni 501 create
exit
bgp
    route-distinguisher 192.0.2.2:501
    vsi-export "vsi-export-policy-501"
    vsi-import "vsi-import-policy-501"
exit
bgp-evpn
    ip-route-advertisement
    vxlan
        no shutdown
    exit
exit
stp
    shutdown
exit
service-name "evi-501"
no shutdown
-----
*A:PE-2>config>service>vpls# info
-----
allow-ip-int-binding
vxlan vni 502 create
exit
bgp
    route-distinguisher 192.0.2.2:502
    vsi-export "vsi-export-policy-502"
    vsi-import "vsi-import-policy-502"
exit
bgp-evpn
    ip-route-advertisement
    vxlan
        no shutdown
    exit
exit
stp
    shutdown
exit
service-name "evi-502"
no shutdown
-----
*A:PE-2>config>router>policy-options# info
-----
community "exp_RVPLS501" members "origin:2:11" "target:64500:501"
community "exp_RVPLS502" members "origin:2:11" "target:64500:502"
community "SOO-PE-2-RVPLS" members "origin:2:11"
community "SOO-PE-3-RVPLS" members "origin:3:11"
community "SOO_PE-3_RVPLS501" members "origin:3:11" "target:64500:501"
community "SOO_PE-3_RVPLS502" members "origin:3:11" "target:64500:502"
policy-statement "vsi-export-policy-501"
    entry 10
        from
            tag 0x5
        exit
        action accept
            community add "SOO_PE-3_RVPLS501"
        exit
    exit
```

```

        entry 20
            action accept
            community add "exp_RVPLS501"
        exit
    exit
exit
policy-statement "vsi-export-policy-502"
    entry 10
        from
            tag 0x5
        exit
        action accept
        community add "SOO_PE-3_RVPLS502"
        exit
    exit
    entry 20
        action accept
        community add "exp_RVPLS502"
        exit
    exit
exit
policy-statement "vsi-import-policy-501"
    entry 10
        from
            community "SOO-PE-2-RVPLS"
        exit
        action reject
    exit
    entry 20
        from
            community "SOO_PE-3_RVPLS501"
        exit
        action accept
        tag 0x5
        exit
    exit
    default-action accept
    exit
exit
policy-statement "vsi-import-policy-502"
    entry 10
        from
            community "SOO-PE-2-RVPLS"
        exit
        action reject
    exit
    entry 20
        from
            community "SOO_PE-3_RVPLS502"
        exit
        action accept
        tag 0x5
        exit
    exit
    default-action accept
    exit
exit

```

Troubleshooting and Debug Commands

For general information on EVPN and VXLAN troubleshooting and debug commands, please refer to chapter [EVPN for VXLAN Tunnels \(Layer 2\) on page 1033](#). This information below focuses on specific commands for Layer-3 applications.

When troubleshooting and operating a EVPN-VXLAN scenario with inter-subnet forwarding, it is important to check the IP prefixes and next-hops, as well as ARP tables and FDB tables (**show router x route-table**, **show router x arp**, **show service id y fdb detail**).

ICMP commands can also help checking the connectivity. When traceroute is used on EVPN-VXLAN in EVPN tunnel interfaces, EVPN tunnel interface hops in the traceroute commands are showing the VPRN loopback address or the other non evpn-tunnel interface address. In VPRN services where all of the interfaces are of type EVPN tunnel, ICMP packets fail until an IP address is configured. The following output shows a traceroute from VPRN 30 in PE-1 to CE-3 and from PE-2 to CE-1 (see [Figure 162](#)):

```
A:PE-1# traceroute router 30 172.16.3.3
traceroute to 172.16.3.3, 30 hops max, 40 byte packets
 1  192.0.2.2 (192.0.2.2)      1.79 ms  1.60 ms  1.51 ms
 2  0.0.0.0 * * *
 3  192.0.2.6 (192.0.2.6)      3.15 ms  3.20 ms  2.93 ms
 4  172.16.3.3 (172.16.3.3)    4.24 ms  3.28 ms  3.31 ms

*A:PE-2# traceroute router 30 172.16.0.1
traceroute to 172.16.0.1, 30 hops max, 0 byte packets

Send failed. Unable to find local ip address
```

When troubleshooting R-VPLS services, specifically R-VPLS services configured as EVPN tunnels, the limit of peer PEs per EVPN tunnel service is much higher than for a regular R-VPLS service since the egress <VTEP, VNI> bindings do not have to be added to the multicast flooding list. For this reason, the following **tools dump** command has been added to check the consumed/total EVPN tunnel next hops. Note that the number of EVPN tunnel next hops matches the number of remote GW-MAC addresses per EVPN tunnel R-VPLS service.

```
A:PE-1# tools dump service id 501 evpn usage
```

```
Evpn Tunnel Interface IP Next Hop: 2/8189
```

Finally, when troubleshooting EVPN routes and routing policies, the **show router bgp routes evpn** command and its filters can help:

- Check that the expected routes are received, properly imported and communities/tags added/replaced/removed.
- Check that the expected routes are sent, properly exported and communities added/replaced/removed.

Examples of EVPN IP prefix routes including communities and tags are shown below.

```
*A:PE-2# show router bgp routes evpn ?
- evpn <evpn-type>
```

```
    inclusive-mcast - Display BGP EVPN Inclusive-Mcast Routes
    ip-prefix       - Display BGP EVPN IP-Prefix Routes
    mac             - Display BGP EVPN Mac Routes
```

```
*A:PE-2# show router bgp routes evpn ip-prefix ?
- ip-prefix [hunt|detail] [rd <rd>] [prefix <ip-prefix/mask>] [community
  <comm-id>] [tag <vni-id>] [next-hop <ip-address>]
```

```
...
```

```
*A:PE-2# show router bgp routes evpn ip-prefix prefix 172.16.0.0/24 hunt community ori-
gin:69:11
```

```
=====
BGP Router ID:192.0.2.2      AS:64500      Local AS:64500
=====
```

```
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
```

```
BGP EVPN IP-Prefix Routes
```

```
=====
-----
```

```
RIB In Entries
```

```
-----
-----
```

```
RIB Out Entries
```

```
-----
-----
```

```
...
```

```
Network      : N/A
Nexthop      : 192.0.2.2
To           : 192.0.2.1
Res. Nexthop : n/a
Local Pref.  : 100
Aggregator AS : None
Atomic Aggr. : Not Atomic
AIGP Metric  : None
Connector    : None
Community    : origin:2:11 target:64500:502

Interface Name : NotAvailable
Aggregator     : None
MED            : 0
```

Troubleshooting and Debug Commands

```
mac-nh:d8:45:ff:00:01:33 bgp-tunnel-encap:VXLAN
Cluster      : No Cluster Members
Originator Id : None                      Peer Router Id : 192.0.2.1
Origin       : IGP
AS-Path      : No As-Path
EVPN type    : IP-PREFIX
ESI          : N/A                      Tag           : 502
Gateway Address: d8:45:ff:00:01:33
Prefix       : 172.16.0.0/24             Route Dist.    : 192.0.2.2:502
MPLS Label   : 0
Route Tag    : 0
Neighbor-AS  : N/A
Orig Validation: N/A
Source Class : 0                      Dest Class     : 0
-----
Routes : 2
=====

*A:PE-2# show router bgp routes evpn ip-prefix prefix 172.16.0.0/24 detail
=====
BGP Router ID:192.0.2.2      AS:64500      Local AS:64500
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP EVPN IP-Prefix Routes
=====
-----
Original Attributes

Network      : N/A
Nexthop      : 192.0.2.1
From         : 192.0.2.1
Res. Nexthop : N/A
Local Pref.  : 100                      Interface Name : NotAvailable
Aggregator AS : None                   Aggregator    : None
Atomic Aggr. : Not Atomic              MED           : 0
AIGP Metric  : None
Connector    : None
Community    : target:64500:201 bgp-tunnel-encap:VXLAN
Cluster      : No Cluster Members
Originator Id : None                      Peer Router Id : 192.0.2.1
Flags        : Used Valid Best IGP
Route Source : Internal
AS-Path      : No As-Path
EVPN type    : IP-PREFIX
ESI          : N/A                      Tag           : 201
Gateway Address: 172.16.1.1
Prefix       : 172.16.0.0/24             Route Dist.    : 192.0.2.1:201
MPLS Label   : 0
Route Tag    : 0
Neighbor-AS  : N/A
Orig Validation: N/A
Source Class : 0                      Dest Class     : 0

Modified Attributes

Network      : N/A
```


EVPN for VXLAN Tunnels (Layer 3)

```

Nexthop      : 192.0.2.1
From         : 192.0.2.1
Res. Nexthop : N/A
Local Pref.  : 100
Aggregator AS : None
Atomic Aggr. : Not Atomic
AIGP Metric  : None
Connector    : None
Community    : target:64500:201 bgp-tunnel-encap:VXLAN
Cluster      : No Cluster Members
Originator Id : None
Flags        : Used Valid Best IGP
Route Source  : Internal
AS-Path       : No As-Path
EVPN type     : IP-PREFIX
ESI           : N/A
Gateway Address: 172.16.1.1
Prefix        : 172.16.0.0/24
MPLS Label    : 0
Route Tag      : 1
Neighbor-AS   : N/A
Orig Validation: N/A
Source Class   : 0
Interface Name : NotAvailable
Aggregator     : None
MED            : 0
Tag            : 201
Route Dist.    : 192.0.2.1:201
Dest Class     : 0

```

...

Conclusion

SR OS supports not only the EVPN control plane for VXLAN tunnels in Layer 2 applications but also the simultaneous use of EVPN and VXLAN for VPN customers (tenants) with intra and inter-subnet connectivity requirements. R-VPLS services can be configured to provide default gateway connectivity to hosts, IRB backhaul connectivity to VPRN services and EVPN tunnel connectivity to VPRN services. When configured to do so, EVPN can advertise IP prefixes and interact with the VPRN RTM to propagate IP prefix connectivity between EVPN and other routing protocols in the VPRN, including IP-VPN. This example has shown how to configure R-VPLS services for all these functions, as well as how to configure routing policies for EVPN-based IP prefixes.

Inter-AS Model C for VLL

In This Chapter

This section describes advanced inter-AS model C for VLL configurations.

Topics in this section include:

- [Applicability on page 1104](#)
- [Overview on page 1105](#)
- [Configuration on page 1107](#)
- [Conclusion on page 1125](#)

Applicability

This chapter is applicable to all of the 7750 SR and 7450 ESS series (including mixed mode). The information was tested with SROS 13.0 R3.

Overview

SR OS supports RFC 3107, Carrying Label Information in BGP-4, including VLL/VPLS. BGP SDPs can also be used with PBB-VPLS services.

ISPs are looking for mechanisms to implement the VLL and VPLS services outside an autonomous system (AS). A service provider may have inter-AS operation as a consequence of delivering inter- provider VLL/VPLS or because they use multiple ASs as a result of acquisitions and merges.

The objective of this chapter is to describe the interconnection of VLL services across multiple ASs.

Network Setup

Figure 166 shows the network setup used.

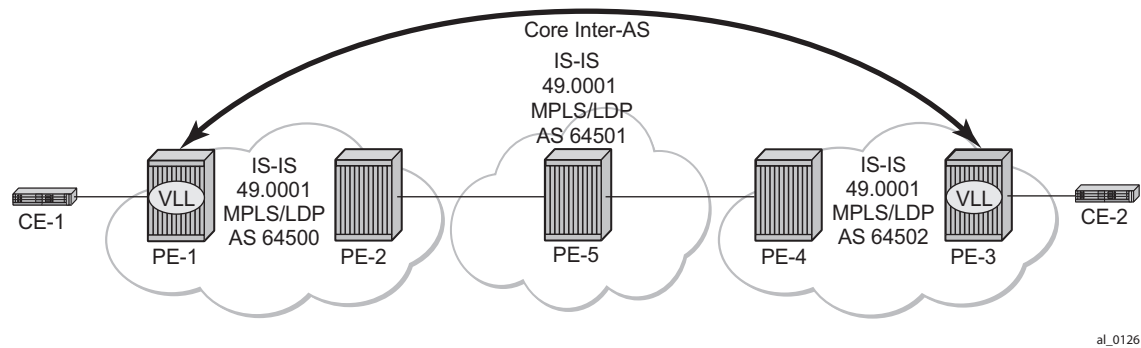


Figure 166: Network Setup - Inter-AS Model C for VLL

The network topology displayed in Figure 166 consists of three sites in different ASs with each site using 7750 SRs.

In AS 64500, there are PE-1 and PE-2, AS 64501 has PE-5 and AS 64502 has PE-4 and PE-3. There is a business customer with two remote locations, Site A and Site B, with customer edge (CE) devices CE-1 connected to the AS 64500 via PE-1 and CE-2 connected to the AS 64502 via PE-3. A VLL service is configured between PE-1 and PE-3 to connect site A and site B.

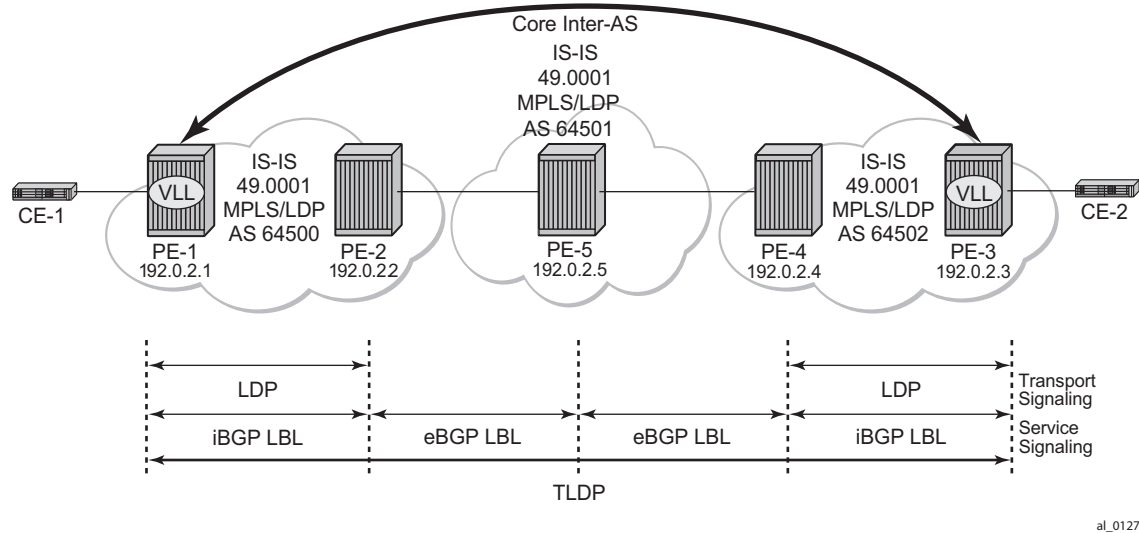


Figure 167: Inter-AS Model C for VLL

Configuration

This section describes all of the relevant configuration tasks for the detailed setup shown in the [Figure 168](#). In this particular example the following protocols are assumed to be already configured.

- IS-IS as the IGP with all the nodes being level Level1/Level 2.
- LDP as the MPLS protocol to signal the transport tunnels.

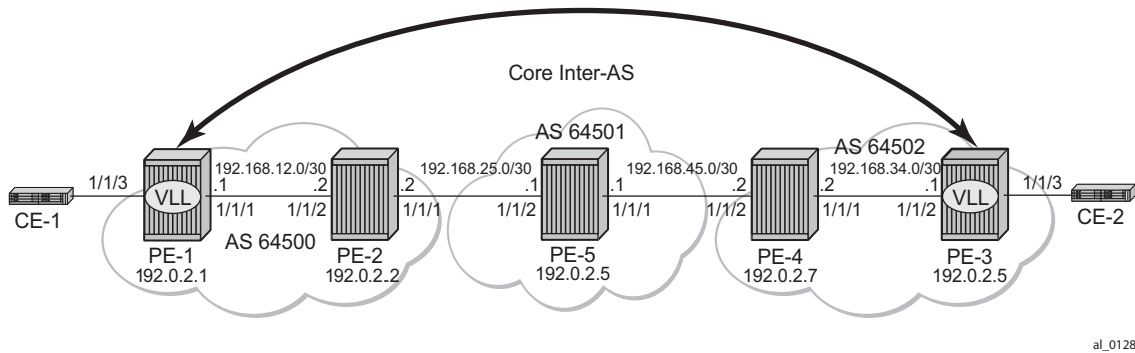


Figure 168: Network Setup Configuration

BGP Configuration

The following CLI output shows the BGP configuration — iBGP and eBGP — required for the PE routers to implement VLL Inter-AS.

The configuration on PE-5 in AS 64501 is the following:

```
*A:PE-5# configure router
      bgp
        min-route-advertisement 1
        rapid-withdrawal
        group "EBGP"
          family ipv4
            type external
            local-as 64501
            neighbor 192.168.25.1
              peer-as 64500
              advertise-label ipv4
            exit
            neighbor 192.168.45.1
              peer-as 64502
              advertise-label ipv4
            exit
          exit
        no shutdown
      exit
```

The **advertise-label ipv4** statement must be configured so that MPLS labels are carried with IPv4 NLRIs.

The configuration of PE-2 in AS 64500 is displayed below:

```
*A:PE-2# configure router
      bgp
        min-route-advertisement 1
        rapid-withdrawal
        group "EBGP"
          family ipv4
            type external
            export "EXPORT-SYSTEMS"
            local-as 64500
            neighbor 192.168.25.2
              peer-as 64501
              advertise-label ipv4
            exit
          exit
        group "IBGP"
          family ipv4
            type internal
            neighbor 192.0.2.1
              next-hop-self
              advertise-label ipv4
            exit
          exit
        no shutdown
      exit
```



```
exit
```

The configuration of PE-4 in AS 64502 is displayed below:

```
*A:PE-4# configure router
      bgp
        min-route-advertisement 1
        export "EXPORT-SYSTEMS"
        rapid-withdrawal
        group "EBGP"
          family ipv4
            type external
            local-as 64502
            neighbor 192.168.45.2
              peer-as 64501
              advertise-label ipv4
            exit
          exit
        group "IBGP"
          family ipv4
            type internal
            neighbor 192.0.2.3
              next-hop-self
              advertise-label ipv4
            exit
          exit
        no shutdown
      exit
```

To complement the BGP setup, the configurations of PE-1 and PE-3 (the PEs to which the CEs are connected in AS 64500 and AS 64502), respectively, are displayed:

PE-1:

```
*A:PE-1# configure router
      bgp
        min-route-advertisement 1
        rapid-withdrawal
        group "IBGP"
          family ipv4
            type internal
            neighbor 192.0.2.2
              next-hop-self
              advertise-label ipv4
            exit
          exit
        no shutdown
      exit
```

PE-3:

```
*A:PE-3# configure router
      bgp
        min-route-advertisement 1
```

Configuration

```
rapid-withdrawal
group "IBGP"
  family ipv4
  type internal
  neighbor 192.0.2.4
    next-hop-self
    advertise-label ipv4
  exit
exit
no shutdown
exit
```

Policy Configuration

The export policy on the PE-2 and PE-4 peering determines the system addresses leaked to the remote AS. It is worth noting here that it is important to modify the default origin attribute from incomplete to IGP in dual-homing scenarios otherwise the iBGP route will be always preferred over the eBGP route.

This is the configuration on PE-2:

```
*A:PE-2# configure router
  policy-options
    begin
    prefix-list "SYSTEMS-AS64500"
      prefix 192.0.2.1/32 exact
      prefix 192.0.2.2/32 exact
    exit
    policy-statement "EXPORT-SYSTEMS"
      entry 10
        from
          prefix-list "SYSTEMS-AS64500"
        exit
        action accept
          origin igp
        exit
      exit
    exit
  commit
exit
```

This is the configuration on PE-4:

```
*A:PE-4# configure router
  policy-options
    begin
    prefix-list "SYSTEMS-AS64502"
      prefix 192.0.2.3/32 exact
      prefix 192.0.2.4/32 exact
    exit
    policy-statement "EXPORT-SYSTEMS"
      entry 10
        from
          prefix-list "SYSTEMS-AS64502"
        exit
        action accept
          origin igp
        exit
      exit
    exit
  commit
exit
```

Service Configuration

Once BGP is configured, the configuration requires the service to be defined (Epipe 1). The focus here is a VLL service (noting that it is also possible to have a similar configuration with VPLS services).

The following CLI shows the service level configuration on PE-1:

```
*A:PE-1# configure service
      sdp 13 mpls create
        far-end 192.0.2.3
        bgp-tunnel
        no shutdown
      exit
    epipe 1 customer 1 create
      description "Tunnel-PE-1-PE-3"
      sap 1/1/3:1 create
      exit
    spoke-sdp 13:1 create
    exit
  no shutdown
exit
```

The following CLI shows the service level configuration on PE-3:

```
*A:PE-3# configure service
      sdp 31 mpls create
        far-end 192.0.2.1
        bgp-tunnel
        no shutdown
      exit
    epipe 1 customer 1 create
      description "Tunnel-PE-3-PE-1"
      sap 1/1/3:1 create
      exit
    spoke-sdp 31:1 create
    exit
  no shutdown
exit
```

Show Commands and Troubleshooting

On PE-3 BGP tunnels do exist to the remote AS system addresses that are using LDP as a transport mechanism and the configuration of end-to-end SDPs over which TLDP service labels are exchanged.

The following output shows the SDP and LDP information:

```
*A:PE-1# show service sdp
=====
Services: Service Destination Points
=====
SdpId  AdmMTU  OprMTU  Far End          Adm  Opr          Del    LSP    Sig
-----
13      0        1552    192.0.2.3        Up   Up           MPLS   B      TLDP
-----
Number of SDPs : 1
-----
Legend: R = RSVP, L = LDP, B = BGP, M = MPLS-TP, n/a = Not Applicable
       I = SR-ISIS, O = SR-OSPF
=====
*A:PE-1#
*A:PE-1# show service service-using
=====
Services
=====
ServiceId  Type      Adm  Opr  CustomerId Service Name
-----
1          Epipe     Up   Up   1          1
2147483648 IES       Up   Down 1          _tmnx_InternalIesService
2147483649 intVpls   Up   Down 1          _tmnx_InternalVplsService
-----
Matching Services : 3
-----
*A:PE-1#
*A:PE-1# show router ldp session
=====
LDP IPv4 Sessions
=====
Peer LDP Id          Adj Type  State          Msg Sent  Msg Recv  Up Time
-----
192.0.2.2:0          Link      Established    95        98        0d 00:04:02
192.0.2.3:0          Targeted  Established    9         10        0d 00:00:27
-----
No. of IPv4 Sessions: 2
=====
LDP IPv6 Sessions
=====
Peer LDP Id
Adj Type          State          Msg Sent      Msg Recv      Up Time
-----
No Matching Entries Found
=====
*A:PE-1#
```

The route-table shows that the system IP address of PE-3 is reachable using a BGP tunnel:

```
*A:PE-1# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type  Proto  Age           Pref
Next Hop[Interface Name]                        Metric
-----
192.0.2.1/32                                     Local  Local   00h06m03s    0
system
192.0.2.2/32                                     Remote  ISIS    00h05m10s   15
192.168.12.2
192.0.2.3/32                                     Remote  BGP     00h00m53s   170
192.0.2.2 (tunneled)
192.0.2.4/32                                     Remote  BGP     00h00m53s   170
192.0.2.2 (tunneled)
192.168.12.0/30                                  Local  Local   00h06m03s    0
int-PE-1-PE-2
-----
No. of Routes: 5
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====
*A:PE-1#
```

The tunnel-table below shows the details of the LDP, SDP and BGP tunnels. This is followed by the service details, noting that Epipe 1 is using SDP 13. A ping is used to show that there is IP connectivity from PE-1 to the system IP address of PE-3:

```
*A:PE-1# show router tunnel-table
=====
IPv4 Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId  Pref  Nexthop      Metric
-----
192.0.2.2/32     ldp       MPLS  65537      9     192.168.12.2  10
192.0.2.3/32     sdp       MPLS  13         5     192.0.2.3     0
192.0.2.3/32     bgp       MPLS  262145     12    192.0.2.2     1000
192.0.2.4/32     bgp       MPLS  262146     12    192.0.2.2     1000
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
*A:PE-1#
*A:PE-1# show service id 1 base
=====
Service Basic Information
=====
Service Id       : 1                Vpn Id          : 0
Service Type     : Epipe
Name             : (Not Specified)
Description      : Tunnel-PE-1-PE-3
Customer Id      : 1                Creation Origin  : manual
Last Status Change: 07/09/2015 14:46:09
```

```

Last Mgmt Change   : 07/09/2015 14:45:55
Test Service       : No
Admin State        : Up                      Oper State      : Up
MTU                : 1514
Vc Switching       : False
SAP Count          : 1                      SDP Bind Count     : 1
Per Svc Hashing    : Disabled
Force QTag Fwd     : Disabled

```

Service Access & Destination Points

Identifier	Type	AdmMTU	OprMTU	Adm	Opr
sap:1/1/3:1	q-tag	1518	1518	Up	Up
sdp:13:1 S(192.0.2.3)	Spok	0	1552	Up	Up

```

=====
*A:PE-1#
*A:PE-1# ping 192.0.2.3
PING 192.0.2.3 56 data bytes
64 bytes from 192.0.2.3: icmp_seq=1 ttl=64 time=1.91ms.
64 bytes from 192.0.2.3: icmp_seq=2 ttl=64 time=2.06ms.
64 bytes from 192.0.2.3: icmp_seq=3 ttl=64 time=2.02ms.
64 bytes from 192.0.2.3: icmp_seq=4 ttl=64 time=2.01ms.
64 bytes from 192.0.2.3: icmp_seq=5 ttl=64 time=2.02ms.

---- 192.0.2.3 PING Statistics ----
5 packets transmitted, 5 packets received, 0.00% packet loss
round-trip min = 1.91ms, avg = 2.01ms, max = 2.06ms, stddev = 0.050ms
*A:PE-1#

```

The same commands on PE-3 are shown below:

```

*A:PE-3# show service sdp
=====
Services: Service Destination Points
=====
SdpId  AdmMTU  OprMTU  Far End      Adm  Opr      Del    LSP    Sig
-----
31      0        1552    192.0.2.1    Up   Up        MPLS   B      TLDP
-----

Number of SDPs : 1
-----

Legend: R = RSVP, L = LDP, B = BGP, M = MPLS-TP, n/a = Not Applicable
        I = SR-ISIS, O = SR-OSPF
=====
*A:PE-3#
*A:PE-3# show service service-using
=====
Services
=====
ServiceId  Type      Adm  Opr  CustomerId  Service Name
-----
1          Epipe     Up   Up    1           _tmnx_InternalIesService
2147483648 IES        Up   Down  1           _tmnx_InternalVplsService
2147483649 intVpls    Up   Down  1           _tmnx_InternalVplsService
-----

Matching Services : 3

```

Configuration

```
-----
=====
*A:PE-3#
*A:PE-3# show router ldp session
=====
LDP IPv4 Sessions
=====
Peer LDP Id          Adj Type  State          Msg Sent  Msg Recv  Up Time
-----
192.0.2.1:0          Targeted  Established    12        13        0d 00:00:44
192.0.2.4:0          Link      Established    98        100       0d 00:04:11
-----
No. of IPv4 Sessions: 2
=====

LDP IPv6 Sessions
=====
Peer LDP Id
Adj Type          State          Msg Sent  Msg Recv  Up Time
-----
No Matching Entries Found
=====
*A:PE-3#
*A:PE-3# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]          Type  Proto  Age          Pref
Next Hop[Interface Name]    Metric
-----
192.0.2.1/32                Remote BGP    00h01m16s  170
192.0.2.4 (tunneled)        0
192.0.2.2/32                Remote BGP    00h01m16s  170
192.0.2.4 (tunneled)        0
192.0.2.3/32                Local  Local  00h06m10s  0
system                      0
192.0.2.4/32                Remote ISIS  00h05m20s  15
192.168.34.2                10
192.168.34.0/30             Local  Local  00h06m11s  0
int-PE-3-PE-4               0
-----
No. of Routes: 5
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====
*A:PE-3#
*A:PE-3# show router tunnel-table
=====
IPv4 Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId  Pref  Nexthop      Metric
-----
192.0.2.1/32      sdp        MPLS 31          5      192.0.2.1     0
192.0.2.1/32      bgp        MPLS 262145     12     192.0.2.4     1000
192.0.2.2/32      bgp        MPLS 262146     12     192.0.2.4     1000
192.0.2.4/32      ldp        MPLS 65537      9      192.168.34.2  10
```



```

-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
*A:PE-3#
*A:PE-3# show service id 1 base
=====
Service Basic Information
=====
Service Id       : 1                Vpn Id       : 0
Service Type     : Epipe
Name             : (Not Specified)
Description      : Tunnel-PE-3-PE-1
Customer Id      : 1                Creation Origin : manual
Last Status Change: 07/09/2015 14:48:39
Last Mgmt Change : 07/09/2015 14:48:33
Test Service     : No
Admin State      : Up               Oper State     : Up
MTU              : 1514
Vc Switching     : False
SAP Count        : 1                SDP Bind Count : 1
Per Svc Hashing  : Disabled
Force QTag Fwd   : Disabled
-----

Service Access & Destination Points
-----
Identifier                               Type      AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/3:1                             q-tag     1518    1518    Up    Up
sdp:31:1 S(192.0.2.1)                   Spok      0       1552    Up    Up
=====
*A:PE-3#
*A:PE-3# ping 192.0.2.1
PING 192.0.2.1 56 data bytes
64 bytes from 192.0.2.1: icmp_seq=1 ttl=64 time=1.83ms.
64 bytes from 192.0.2.1: icmp_seq=2 ttl=64 time=2.06ms.
64 bytes from 192.0.2.1: icmp_seq=3 ttl=64 time=2.01ms.
64 bytes from 192.0.2.1: icmp_seq=4 ttl=64 time=2.08ms.
64 bytes from 192.0.2.1: icmp_seq=5 ttl=64 time=2.15ms.

---- 192.0.2.1 PING Statistics ----
5 packets transmitted, 5 packets received, 0.00% packet loss
round-trip min = 1.83ms, avg = 2.03ms, max = 2.15ms, stddev = 0.107ms
*A:PE-3#

```

On PE-3, the BGP route to the system IP address of PE-1 can be seen with PE-4 as the next hop:

```

*A:PE-3# show router bgp routes
=====
BGP Router ID:192.0.2.3      AS:64502      Local AS:64502
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
              l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

```

Configuration

```

=====
BGP IPv4 Routes
=====
Flag   Network                               LocalPref  MED
      Nexthop (Router)                   Path-Id    Label
      As-Path
-----
u*>i   192.0.2.1/32                           100        None
      192.0.2.4                           None       262140
      64501 64500
u*>i   192.0.2.2/32                           100        None
      192.0.2.4                           None       262141
      64501 64500
*i     192.0.2.3/32                           100        10
      192.0.2.4                           None       262139
      No As-Path
*i     192.0.2.4/32                           100        None
      192.0.2.4                           None       262138
      No As-Path
-----
Routes : 4
=====
*A:PE-3#

```

Again on PE-3, the FIB on slot 1 shows that the system IP address of PE-1 is reachable using BGP over an LDP transport to PE-4:

```

*A:PE-3# show router fib 1
=====
FIB Display
=====
Prefix [Flags]                               Protocol
      NextHop
-----
192.0.2.1/32                                BGP
      192.0.2.4 (Transport:LDP)
192.0.2.2/32                                BGP
      192.0.2.4 (Transport:LDP)
192.0.2.3/32                                LOCAL
      192.0.2.3 (system)
192.0.2.4/32                                ISIS
      192.168.34.2 (int-PE-3-PE-4)
192.168.34.0/30                             LOCAL
      192.168.34.0 (int-PE-3-PE-4)
-----
Total Entries : 5
=====
*A:PE-3#

```

The show commands on PE-5 router in AS 64501 are as follows:

```
*A:PE-9# show router bgp summary
=====
BGP Router ID:192.0.2.9          AS:64501          Local AS:64501
=====
BGP Admin State      : Up          BGP Oper State      : Up
T*A:PE-5# show router bgp summary
=====
BGP Router ID:192.0.2.5          AS:64501          Local AS:64501
=====
BGP Admin State      : Up          BGP Oper State      : Up
Total Peer Groups    : 1          Total Peers          : 2
Total BGP Paths       : 9          Total Path Memory     : 1720
Total IPv4 Remote Rts : 8          Total IPv4 Rem. Active Rts : 4
Total McIPv4 Remote Rts : 0        Total McIPv4 Rem. Active Rts : 0
Total McIPv6 Remote Rts : 0        Total McIPv6 Rem. Active Rts : 0
Total IPv6 Remote Rts : 0          Total IPv6 Rem. Active Rts : 0
Total IPv4 Backup Rts : 0          Total IPv6 Backup Rts  : 0

Total Supressed Rts   : 0          Total Hist. Rts      : 0
Total Decay Rts       : 0

Total VPN Peer Groups : 0          Total VPN Peers      : 0
Total VPN Local Rts   : 0
Total VPN-IPv4 Rem. Rts : 0        Total VPN-IPv4 Rem. Act. Rts : 0
Total VPN-IPv6 Rem. Rts : 0        Total VPN-IPv6 Rem. Act. Rts : 0
Total VPN-IPv4 Bkup Rts : 0        Total VPN-IPv6 Bkup Rts  : 0

Total VPN Supp. Rts   : 0          Total VPN Hist. Rts  : 0
Total VPN Decay Rts   : 0

Total L2-VPN Rem. Rts : 0          Total L2VPN Rem. Act. Rts : 0
Total MVPN-IPv4 Rem Rts : 0        Total MVPN-IPv4 Rem Act Rts : 0
Total MDT-SAFI Rem Rts : 0          Total MDT-SAFI Rem Act Rts : 0
Total MSPW Rem Rts    : 0          Total MSPW Rem Act Rts   : 0
Total RouteTgt Rem Rts : 0          Total RouteTgt Rem Act Rts : 0
Total McVpnIPv4 Rem Rts : 0        Total McVpnIPv4 Rem Act Rts : 0
Total MVPN-IPv6 Rem Rts : 0        Total MVPN-IPv6 Rem Act Rts : 0
Total EVPN Rem Rts     : 0          Total EVPN Rem Act Rts   : 0
Total FlowIpv4 Rem Rts : 0          Total FlowIpv4 Rem Act Rts : 0
Total FlowIpv6 Rem Rts : 0          Total FlowIpv6 Rem Act Rts : 0
=====
BGP Summary
=====
Neighbor
Description
AS PktRcvd InQ Up/Down State|Rcv/Act/Sent (Addr Family)
PktSent OutQ
-----
192.168.25.1
64500      14    0 00h03m44s 4/2/4 (IPv4)
           14    0
192.168.45.1
64502      15    0 00h03m44s 4/2/4 (IPv4)
           14    0
-----
*A:PE-5#
```

Configuration

```
*A:PE-5# show router bgp routes
=====
BGP Router ID:192.0.2.5      AS:64501      Local AS:64501
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP IPv4 Routes
=====
Flag  Network                               LocalPref  MED
      Nexthop (Router)                     Path-Id    Label
      As-Path
-----
u*>i  192.0.2.1/32                          None       10
      192.168.25.1                          None       262141
      64500
i      192.0.2.1/32                          None       None
      192.168.45.1                          None       262140
      64502 64501 64500
u*>i  192.0.2.2/32                          None       None
      192.168.25.1                          None       262140
      64500
i      192.0.2.2/32                          None       None
      192.168.45.1                          None       262141
      64502 64501 64500
u*>i  192.0.2.3/32                          None       10
      192.168.45.1                          None       262139
      64502
i      192.0.2.3/32                          None       None
      192.168.25.1                          None       262138
      64500 64501 64502
u*>i  192.0.2.4/32                          None       None
      192.168.45.1                          None       262138
      64502
i      192.0.2.4/32                          None       None
      192.168.25.1                          None       262139
      64500 64501 64502
-----
Routes : 8
=====
*A:PE-5#
*A:PE-5# show router bgp inter-as-label
=====
BGP Inter-AS labels
Flags: B - entry has backup, P - entry is promoted
=====
NextHop                               Received   Advertised   Label
                                      Label      Label        Origin
-----
192.168.25.1                         262140    262142       External
192.168.25.1                         262141    262143       External
192.168.45.1                         262138    262140       External
192.168.45.1                         262139    262141       External
-----
Total Labels allocated: 4
=====
*A:PE-5#
```

```

*A:PE-5# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]                                Type   Proto   Age           Pref
  Next Hop[Interface Name]                        Metric
-----
192.0.2.1/32                                     Remote BGP      00h03m17s    170
      192.168.25.1                                0
192.0.2.2/32                                     Remote BGP      00h03m17s    170
      192.168.25.1                                0
192.0.2.3/32                                     Remote BGP      00h03m04s    170
      192.168.45.1                                0
192.0.2.4/32                                     Remote BGP      00h03m04s    170
      192.168.45.1                                0
192.0.2.5/32                                     Local  Local    00h07m44s     0
      system                                         0
192.168.25.0/30                                  Local  Local    00h07m44s     0
      int-PE-5-PE-2                                0
192.168.45.0/30                                  Local  Local    00h07m44s     0
      int-PE-5-PE-4                                0
-----
No. of Routes: 7
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====
*A:PE-5#

```

The commands on PE-2 are shown below:

```

*A:PE-2# show router bgp summary
=====
BGP Router ID:192.0.2.2      AS:64500      Local AS:64500
=====
BGP Admin State      : Up      BGP Oper State      : Up
Total Peer Groups    : 2      Total Peers          : 2
Total BGP Paths       : 7      Total Path Memory    : 1320
Total IPv4 Remote Rts : 4      Total IPv4 Rem. Active Rts : 2
Total McIPv4 Remote Rts : 0    Total McIPv4 Rem. Active Rts : 0
Total McIPv6 Remote Rts : 0    Total McIPv6 Rem. Active Rts : 0
Total IPv6 Remote Rts  : 0    Total IPv6 Rem. Active Rts  : 0
Total IPv4 Backup Rts  : 0    Total IPv6 Backup Rts    : 0

Total Suppressed Rts  : 0      Total Hist. Rts      : 0
Total Decay Rts       : 0

Total VPN Peer Groups : 0      Total VPN Peers      : 0
Total VPN Local Rts   : 0
Total VPN-IPv4 Rem. Rts : 0    Total VPN-IPv4 Rem. Act. Rts : 0
Total VPN-IPv6 Rem. Rts : 0    Total VPN-IPv6 Rem. Act. Rts : 0
Total VPN-IPv4 Bkup Rts : 0    Total VPN-IPv6 Bkup Rts   : 0

Total VPN Supp. Rts   : 0      Total VPN Hist. Rts   : 0
Total VPN Decay Rts   : 0

Total L2-VPN Rem. Rts : 0      Total L2VPN Rem. Act. Rts : 0
Total MVPN-IPv4 Rem Rts : 0    Total MVPN-IPv4 Rem Act Rts : 0

```

Configuration

```
Total MDT-SAFI Rem Rts : 0          Total MDT-SAFI Rem Act Rts : 0
Total MSPW Rem Rts      : 0          Total MSPW Rem Act Rts      : 0
Total RouteTgt Rem Rts  : 0          Total RouteTgt Rem Act Rts  : 0
Total McVpnIPv4 Rem Rts : 0          Total McVpnIPv4 Rem Act Rts : 0
Total MVPN-IPv6 Rem Rts : 0          Total MVPN-IPv6 Rem Act Rts : 0
Total EVPN Rem Rts      : 0          Total EVPN Rem Act Rts      : 0
Total FlowIpv4 Rem Rts  : 0          Total FlowIpv4 Rem Act Rts  : 0
Total FlowIpv6 Rem Rts  : 0          Total FlowIpv6 Rem Act Rts  : 0

=====
BGP Summary
=====
Neighbor
Description
          AS PktRcvd InQ  Up/Down  State|Rcv/Act/Sent (Addr Family)
          PktSent OutQ
-----
192.0.2.1
          64500      14    0 00h05m07s 0/0/2 (IPv4)
                   16    0
192.168.25.2
          64501      13    0 00h03m34s 4/2/4 (IPv4)
                   15    0
-----
*A:PE-2#
*A:PE-2# show router bgp inter-as-label
=====
BGP Inter-AS labels
Flags: B - entry has backup, P - entry is promoted
=====
NextHop                Received      Advertised      Label
                        Label              Label           Origin
-----
192.0.2.1                0              262141          Internal
192.0.2.2                0              262140          Edge
192.168.25.2             262140         262139          External
192.168.25.2             262141         262138          External
-----
Total Labels allocated:  4
=====
*A:PE-2#
*A:PE-2# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix[Flags]      Type   Proto   Age      Pref
Next Hop[Interface Name] Metric
-----
192.0.2.1/32            Remote ISIS   00h07m12s 15
      192.168.12.1              10
192.0.2.2/32            Local  Local   00h07m56s 0
      system                    0
192.0.2.3/32            Remote BGP     00h02m55s 170
      192.168.25.2              0
192.0.2.4/32            Remote BGP     00h02m55s 170
      192.168.25.2              0
192.168.12.0/30         Local  Local   00h07m56s 0
      int-PE-2-PE-1            0
192.168.25.0/30         Local  Local   00h07m56s 0
```

```

int-PE-2-PE-5
-----
No. of Routes: 6
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
      S = Sticky ECMP requested
=====
*A:PE-2#

```

The show commands on PE-4:

```

*A:PE-4# show router bgp summary
=====
BGP Router ID:192.0.2.4      AS:64502      Local AS:64502
=====
BGP Admin State      : Up      BGP Oper State      : Up
Total Peer Groups    : 2      Total Peers          : 2
Total BGP Paths      : 7      Total Path Memory    : 1320
Total IPv4 Remote Rts : 4      Total IPv4 Rem. Active Rts : 2
Total McIPv4 Remote Rts : 0    Total McIPv4 Rem. Active Rts: 0
Total McIPv6 Remote Rts : 0    Total McIPv6 Rem. Active Rts: 0
Total IPv6 Remote Rts : 0      Total IPv6 Rem. Active Rts : 0
Total IPv4 Backup Rts : 0      Total IPv6 Backup Rts  : 0

Total Suppressed Rts : 0      Total Hist. Rts      : 0
Total Decay Rts      : 0

Total VPN Peer Groups : 0      Total VPN Peers      : 0
Total VPN Local Rts   : 0
Total VPN-IPv4 Rem. Rts : 0    Total VPN-IPv4 Rem. Act. Rts: 0
Total VPN-IPv6 Rem. Rts : 0    Total VPN-IPv6 Rem. Act. Rts: 0
Total VPN-IPv4 Bkup Rts : 0    Total VPN-IPv6 Bkup Rts : 0

Total VPN Supp. Rts   : 0      Total VPN Hist. Rts   : 0
Total VPN Decay Rts   : 0

Total L2-VPN Rem. Rts : 0      Total L2VPN Rem. Act. Rts : 0
Total MVPN-IPv4 Rem Rts : 0    Total MVPN-IPv4 Rem Act Rts : 0
Total MDT-SAFI Rem Rts : 0      Total MDT-SAFI Rem Act Rts : 0
Total MSPW Rem Rts     : 0      Total MSPW Rem Act Rts     : 0
Total RouteTgt Rem Rts : 0      Total RouteTgt Rem Act Rts : 0
Total McVpnIPv4 Rem Rts : 0     Total McVpnIPv4 Rem Act Rts : 0
Total MVPN-IPv6 Rem Rts : 0     Total MVPN-IPv6 Rem Act Rts : 0
Total EVPN Rem Rts      : 0      Total EVPN Rem Act Rts      : 0
Total FlowIpv4 Rem Rts  : 0      Total FlowIpv4 Rem Act Rts  : 0
Total FlowIpv6 Rem Rts  : 0      Total FlowIpv6 Rem Act Rts  : 0
=====
BGP Summary
=====
Neighbor
Description
AS PktRcvd InQ Up/Down State|Rcv/Act/Sent (Addr Family)
PktSent OutQ
-----
192.0.2.3
64502 11 0 00h03m47s 0/0/4 (IPv4)
16 0

```

Configuration

```

192.168.45.2
        64501      13      0 00h03m38s 4/2/4 (IPv4)
                  16      0
-----
*A:PE-4#
*A:PE-4# show router bgp routes
=====
BGP Router ID:192.0.2.4      AS:64502      Local AS:64502
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP IPv4 Routes
=====
Flag  Network                               LocalPref  MED
      Nexthop (Router)                     Path-Id    Label
      As-Path
-----
u*>i  192.0.2.1/32                           None       None
      192.168.45.2                           None       262143
      64501 64500
u*>i  192.0.2.2/32                           None       None
      192.168.45.2                           None       262142
      64501 64500
i     192.0.2.3/32                           None       None
      192.168.45.2                           None       262141
      64501 64502
i     192.0.2.4/32                           None       None
      192.168.45.2                           None       262140
      64501 64502
-----
Routes : 4
=====
*A:PE-4#
*A:PE-4# show router bgp inter-as-label
=====
BGP Inter-AS labels
Flags: B - entry has backup, P - entry is promoted
=====
NextHop                               Received    Advertised  Label
                                      Label       Label       Origin
-----
192.168.45.2                         262142     262141     External
192.168.45.2                         262143     262140     External
192.0.2.3                            0          262139     Internal
192.0.2.4                            0          262138     Edge
-----
Total Labels allocated: 4
=====
*A:PE-4#

```


Conclusion

The BGP tunnel based SDP binding is allowed for VLL and VPLS services, including PBB-VPLS. Using RFC 3107, it is possible to implement inter-AS Model C VLLs.

The example used in this chapter illustrates the configuration of VLL Inter-AS for access CE sites. Troubleshooting commands also have been shown to verify all the procedures.

LDP VPLS using BGP-Auto Discovery

In This Chapter

This section provides information about LDP VPLS using BGP-Auto Discovery.

Topics in this section include:

- [Applicability on page 1128](#)
- [Summary on page 1129](#)
- [Overview on page 1130](#)
- [Configuration on page 1132](#)
- [Conclusion on page 1157](#)

Applicability

This example is applicable to all of the 7x50 series and was tested on release SR OS 13.0.R1.
There are no pre-requisites for this configuration.

Summary

MPLS-based Virtual Private LAN Services (VPLS) may have many different provisioning models to allow the signaling of pseudowires between PE routers containing VPLS instances.

Network Management System (NMS) provisioning using LDP signaling is a well understood method of provisioning of Layer 2 VPLS services as is described in RFC 4762. This relies on the provisioning of pseudowires between VPLS instances using Label Distribution Protocol (LDP) signaling with a common virtual circuit (VC) identifier within the label mapping message to instantiate pseudowires.

Border Gateway Protocol (BGP) Auto Discovery (RFC 6074) is an alternative method of provisioning of Layer 2 Provider Edge routers containing VPLS service instances to those described above where PEs in a common VPLS instance are automatically discovered using BGP Auto Discovery (BGP-AD) techniques.

Each PE router advertises the presence of VPLS instances to other PE routers using defined parameters within a BGP update message.

LDP is used as the pseudowire signaling protocol and relies on the auto-discovery of VPLS endpoints to instantiate pseudowires instead of manually provisioning virtual circuits. Locally configured parameters, along with BGP learned parameters, are used to determine local and remote VPLS endpoints, which are used by LDP to signal service labels to peer routers.

Knowledge of BGP-Auto-discovery RFC 6074 architecture and functionality, RFC 4447 Pseudo-wire Set-up using Label Distribution Protocol is assumed throughout this section, as well as knowledge of Multi-Protocol BGP (MP-BGP).

Overview

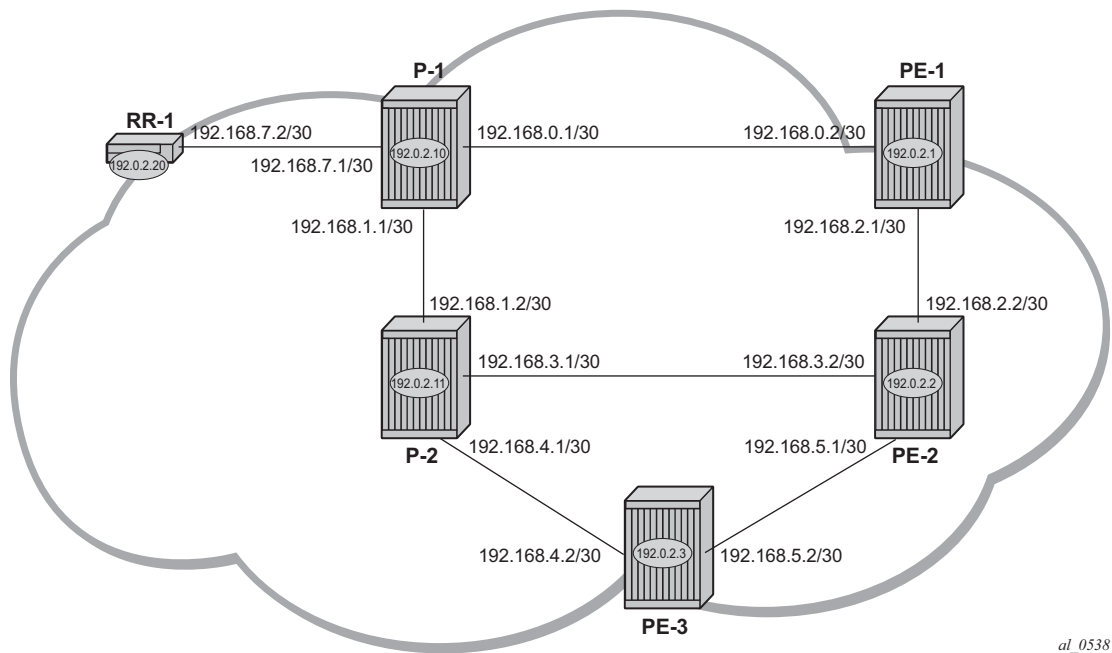


Figure 169: Network Topology

The network topology is displayed in [Figure 169](#). The setup uses six 7x50 nodes located in the same Autonomous System (AS). There are three PEs and RR-1 will act as a Route Reflector for the AS. The Provider Edge routers are all VPLS aware. The Provider (P) routers are VPLS unaware and do not take part in the BGP process. A full mesh VPLS between PE-1, PE-2 and PE-3 is described.

The following configuration tasks should be completed as a pre-requisite:

- ISIS or OSPF should be enabled on all network interfaces between each of the PE/P routers and route reflector.
- MPLS should be configured on all interfaces between PE and P routers; MPLS is not required between P-1 and RR-1.
- LDP should be configured on interfaces between PE and P routers. It is not required between P-1 and RR-1.
- RSVP protocol is disabled by default, so the RSVP protocol should be enabled.

BGP-AD

In this architecture a VPLS service is a collection of local VPLS instances present on a number of PEs in a provider network. In this context, VPLS-aware devices are PE routers. Each VPLS instance has a unique identifier known as the VPLS identifier (VPLS-id). All PEs that have this VPLS instance present will have a common VPLS-id configured.

Each VPLS instance within a PE contains a Virtual Switching Instance (VSI). The VPLS attachment circuits and pseudowires are associated with the VSI. Each VSI within a given VPLS has a unique identifier called the VSI identifier (VSI-id) and is a concatenation of the VPLS-id plus an IP address, usually the system IP address.

The PEs communicate with each other at the control plane level by means of BGP updates containing BGP Layer 2 Network Layer Reachability Information (NLRI). Each update contains enough information for a PE to determine the presence of other local VPLS instances on peering PEs. In turn, this allows peer PE routers to setup pseudowire connectivity using LDP signaling for data flow between peers containing a local VPLS within the same VPLS instances.

Each update contains parameters usually associated with Multi-Protocol BGP updates:

- NLRI encoded as route-target (usually the VPLS-id) and PE system address.
- Next-Hop — The system IP address of the sending PE router.
- Extended communities — Contains the route target extended community and the VPLS-id as community values.

Each VPLS instance is configured with import and export route target extended communities to create the required pseudowire topology by controlling the distribution of each NLRI.

The purpose of this section is to describe the provisioning of a VPLS instance across three PE routers. A full mesh of pseudowires interconnects the VSI of each PE within the VPLS instance. A single attachment circuit is also configured on each VSI.

Configuration

The first step is to configure an MP-iBGP session using the L2VPN address family between each of the PEs and the route reflector.

The configuration for PE-1 is:

```
configure
  router
    bgp
      group "internal"
        family l2-vpn
        type internal
        neighbor 192.0.2.20
      exit
    exit
    no shutdown
  exit
exit
```

The configuration for the other PE nodes is identical. The IP addresses can be derived from [Figure 169](#).

The configuration for route reflector RR-1 is:

```
configure
  router
    bgp
      cluster 1.1.1.1
      group "rr-internal"
        family l2-vpn
        type internal
        neighbor 192.0.2.1
      exit
        neighbor 192.0.2.2
      exit
        neighbor 192.0.2.3
      exit
    exit
    no shutdown
  exit
exit
```

On PE-1, verify that the BGP session with RR-1 is established with address family l2-vpn capability negotiated:

```
*A:PE-1# show router bgp neighbor 192.0.2.20
```

```
=====
BGP Neighbor
=====
```



```

-----
Peer                : 192.0.2.20
Description         : (Not Specified)
Group              : internal
-----
Peer AS             : 65536           Peer Port          : 51039
Peer Address        : 192.0.2.20
Local AS            : 65536           Local Port          : 179
Local Address       : 192.0.2.1
Peer Type           : Internal
State               : Established      Last State          : Established
Last Event          : recvKeepAlive
Last Error          : Cease (Connection Collision Resolution)
Local Family        : L2-VPN
Remote Family       : L2-VPN
Hold Time           : 90              Keep Alive          : 30
Min Hold Time       : 0
Active Hold Time    : 90              Active Keep Alive    : 30
Cluster Id          : None
Preference          : 170             Num of Update Flaps : 0
Recd. Paths         : 6
IPv4 Recd. Prefixes : 0              IPv4 Active Prefixes : 0
IPv4 Suppressed Pfxs : 0             VPN-IPv4 Suppr. Pfxs : 0
VPN-IPv4 Recd. Pfxs : 0             VPN-IPv4 Active Pfxs : 0
Mc IPv4 Recd. Pfxs. : 0             Mc IPv4 Active Pfxs. : 0
Mc IPv4 Suppr. Pfxs : 0             IPv6 Suppressed Pfxs : 0
IPv6 Recd. Prefixes : 0             IPv6 Active Prefixes : 0
VPN-IPv6 Recd. Pfxs : 0             VPN-IPv6 Active Pfxs : 0
VPN-IPv6 Suppr. Pfxs : 0
Mc IPv6 Recd. Pfxs. : 0             Mc IPv6 Active Pfxs. : 0
Mc IPv6 Suppr. Pfxs : 0             L2-VPN Suppr. Pfxs   : 0
L2-VPN Recd. Pfxs   : 6             L2-VPN Active Pfxs   : 4
MVPN-IPv4 Suppr. Pfxs : 0           MVPN-IPv4 Recd. Pfxs : 0
MVPN-IPv4 Active Pfxs : 0           MDT-SAFI Suppr. Pfxs : 0
MDT-SAFI Recd. Pfxs : 0             MDT-SAFI Active Pfxs : 0
Flow-IPv4 Suppr. Pfxs : 0           Flow-IPv4 Recd. Pfxs : 0
Flow-IPv4 Active Pfxs : 0           Rte-Tgt Suppr. Pfxs  : 0
Rte-Tgt Recd. Pfxs  : 0             Rte-Tgt Active Pfxs  : 0
Backup IPv4 Pfxs     : 0             Backup IPv6 Pfxs      : 0
Mc Vpn Ipv4 Recd. Pf* : 0           Mc Vpn Ipv4 Active P* : 0
Mc Vpn Ipv4 Suppr. P* : 0
Backup Vpn Ipv4 Pfxs : 0           Backup Vpn Ipv6 Pfxs : 0
Input Queue          : 0             Output Queue          : 0
i/p Messages         : 31            o/p Messages          : 26
i/p Octets           : 1380          o/p Octets            : 651
i/p Updates          : 9             o/p Updates           : 2
MVPN-IPv6 Suppr. Pfxs : 0           MVPN-IPv6 Recd. Pfxs : 0
MVPN-IPv6 Active Pfxs : 0
Flow-IPv6 Suppr. Pfxs : 0           Flow-IPv6 Recd. Pfxs : 0
Flow-IPv6 Active Pfxs : 0
Evpn Suppr. Pfxs     : 0             Evpn Recd. Pfxs       : 0
Evpn Active Pfxs     : 0
MS-PW Suppr. Pfxs    : 0             MS-PW Recd. Pfxs      : 0
MS-PW Active Pfxs    : 0
TTL Security         : Disabled      Min TTL Value         : n/a
Graceful Restart      : Disabled      Stale Routes Time     : n/a
Restart Time          : n/a
Advertise Inactive    : Disabled      Peer Tracking          : Disabled
Advertise Label       : None

```

Configuration

```
Auth key chain          : n/a
Disable Cap Nego        : Disabled
Flowspec Validate       : Disabled
Aigp Metric             : Disabled
Damp Peer Oscillatio*   : Disabled
GR Notification         : Disabled
Rem Idle Hold Time      : 00h00m00s
Next-Hop Unchanged      : None
L2 VPN Cisco Interop    : Disabled
Local Capability        : RtRefresh MPBGP 4byte ASN
Remote Capability       : RtRefresh MPBGP 4byte ASN
Local AddPath Capabi*   : Disabled
Remote AddPath Capab*   : Send - None
                        : Receive - None
Import Policy           : None Specified / Inherited
Export Policy           : None Specified / Inherited
Origin Validation       : N/A
EBGP Link Bandwidth     : n/a
IPv4 Rej. Pfxs          : 0
VPN-IPv4 Rej. Pfxs      : 0
Mc IPv4 Rej. Pfxs       : 0
MVPN-IPv4 Rej. Pfxs     : 0
Flow-IPv4 Rej. Pfxs     : 0
L2-VPN Rej. Pfxs        : 0
Rte-Tgt Rej. Pfxs       : 0
Mc Vpn Ipv4 Rej. Pfxs   : 0
Bfd Enabled             : Disabled
Default Route Tgt       : Disabled
Split Horizon           : Disabled
Update Errors           : 0
Fault Tolerance         : Disabled
IPv6 Rej. Pfxs          : 0
VPN-IPv6 Rej. Pfxs      : 0
Mc IPv6 Rej. Pfxs       : 0
MVPN-IPv6 Rej. Pfxs     : 0
Flow-IPv6 Rej. Pfxs     : 0
MDT-SAFI Rej. Pfxs      : 0
MS-PW Rej. Pfxs         : 0
Evpn Rej. Pfxs          : 0
-----
Neighbors : 1
=====
* indicates that the corresponding row element may have been truncated.
*A:PE-1#
```

On RR-1, show that BGP sessions with each PE are established, and have correctly negotiated the l2-vpn address family capability.

```
A:RR-1# show router bgp summary
=====
BGP Router ID:192.0.2.20      AS:65536      Local AS:65536
=====
BGP Admin State      : Up      BGP Oper State      : Up
Total Peer Groups     : 1      Total Peers          : 3
Total BGP Paths       : 26     Total Path Memory    : 4984
Total IPv4 Remote Rts : 0      Total IPv4 Rem. Active Rts : 0
Total McIPv4 Remote Rts : 0    Total McIPv4 Rem. Active Rts : 0
Total McIPv6 Remote Rts : 0    Total McIPv6 Rem. Active Rts : 0
Total IPv6 Remote Rts : 0      Total IPv6 Rem. Active Rts : 0
Total IPv4 Backup Rts : 0      Total IPv6 Backup Rts  : 0

Total Supressed Rts   : 0      Total Hist. Rts      : 0
Total Decay Rts       : 0

Total VPN Peer Groups : 0      Total VPN Peers      : 0
Total VPN Local Rts   : 0
Total VPN-IPv4 Rem. Rts : 0    Total VPN-IPv4 Rem. Act. Rts : 0
Total VPN-IPv6 Rem. Rts : 0    Total VPN-IPv6 Rem. Act. Rts : 0
Total VPN-IPv4 Bkup Rts : 0    Total VPN-IPv6 Bkup Rts  : 0
```

```

Total VPN Supp. Rts      : 0          Total VPN Hist. Rts      : 0
Total VPN Decay Rts      : 0

Total L2-VPN Rem. Rts    : 22         Total L2VPN Rem. Act. Rts    : 0
Total MVPN-IPv4 Rem Rts  : 0          Total MVPN-IPv4 Rem Act Rts  : 0
Total MDT-SAFI Rem Rts   : 0          Total MDT-SAFI Rem Act Rts   : 0
Total MSPW Rem Rts       : 0          Total MSPW Rem Act Rts       : 0
Total RouteTgt Rem Rts   : 0          Total RouteTgt Rem Act Rts   : 0
Total McVpnIPv4 Rem Rts  : 0          Total McVpnIPv4 Rem Act Rts  : 0
Total MVPN-IPv6 Rem Rts  : 0          Total MVPN-IPv6 Rem Act Rts  : 0
Total EVPN Rem Rts       : 0          Total EVPN Rem Act Rts       : 0
Total FlowIpv4 Rem Rts   : 0          Total FlowIpv4 Rem Act Rts   : 0
Total FlowIpv6 Rem Rts   : 0          Total FlowIpv6 Rem Act Rts   : 0

```

```

=====
BGP Summary
=====
Neighbor
          AS PktRcvd InQ  Up/Down  State|Rcv/Act/Sent (Addr Family)
          PktSent OutQ
-----
192.0.2.1
          65536      96    0 00h43m38s 6/0/22 (L2VPN)
              111      0
192.0.2.2
          65536     100    0 00h44m20s 9/0/22 (L2VPN)
              113      0
192.0.2.3
          65536      97    0 00h43m47s 7/0/22 (L2VPN)
              112      0
-----
A:RR-1#

```

A full mesh of RSVP Label Switched Paths (LSPs) is configured between the PE routers. For reference, the MPLS interface configuration and LSPs for PE-1 to PE-2 and PE-3 is:

```

A:PE-1# configure router mpls
      interface "system"
        no shutdown
      exit
      interface "int-PE-1-P-1"
        no shutdown
      exit
      interface "int-PE-1-PE-2"
        no shutdown
      exit
      path "loose"
        no shutdown
      exit
      lsp "LSP-PE-1-PE-2"
        to 192.0.2.2
        primary "loose"
      exit
        no shutdown
      exit
      lsp "LSP-PE-1-PE-3"
        to 192.0.2.3

```

Configuration

```
primary "loose"  
exit  
no shutdown  
exit  
no shutdown
```

VPLS PE Configuration

Pseudowire-Templates

Pseudowire templates are used by BGP to dynamically instantiate Service Distribution Point (SDP) bindings, for a given service they are used to signal the egress service de-multiplexor labels used by remote PEs to reach the local PE.

The template determines the signaling parameters of the pseudowire, control word presence, plus other usage characteristics such as Split Horizon Groups, MAC-pinning, filters, etc.

The MPLS transport tunnel between PE routers can be signaled using either LDP or RSVP.

LDP based pseudowires can be automatically instantiated. RSVP based SDPs have to be pre-provisioned.

Pseudowire Templates for Auto-SDP Creation using LDP

In order to use an LDP transport tunnel for data flow between PEs, it is necessary for link layer LDP to be configured between all PEs/Ps so that a transport label for each PE's system interface address is available. Using this mechanism SDPs can be auto-instantiated with SDP ids starting at 17407. Any subsequent SDPs created use SDP-ids decrementing from this value.

A pseudowire template is required which may contain a split-horizon group. Each SDP created with this template is contained within the configured split horizon group so that traffic cannot be forwarded between them.

```
A:PE-1# configure service
      pw-template 1 create
        split-horizon-group "vpls-shg"
      exit
    exit
```

A pseudowire template can also be created that does not contain a split-horizon group. The split horizon group can then be specified when the pw-template is included within the service.

```
A:PE-1# configure service
      pw-template 2 create
    exit
```

Pseudowire Templates for Provisioned SDPs using RSVP

To use an RSVP tunnel as transport between PEs, it is necessary to bind the RSVP LSPs to the SDPs between each PE.

SDP creation from PE-1 to PE-2:

```
A:PE-1# configure service sdp 43 mpls create
      far-end 192.0.2.2
      lsp "LSP-PE-1-PE-2"
      keep-alive
      shutdown
      exit
      no shutdown
```

To create an SDP within a service that uses the RSVP transport tunnel, a pseudowire template is required that has the **use-provisioned-sdp** parameter.

```
A:PE-1# configure service
      pw-template 3 use-provisioned-sdp create
      exit
      exit
```

VPLS BGP-AD using Auto-Provisioned SDPs

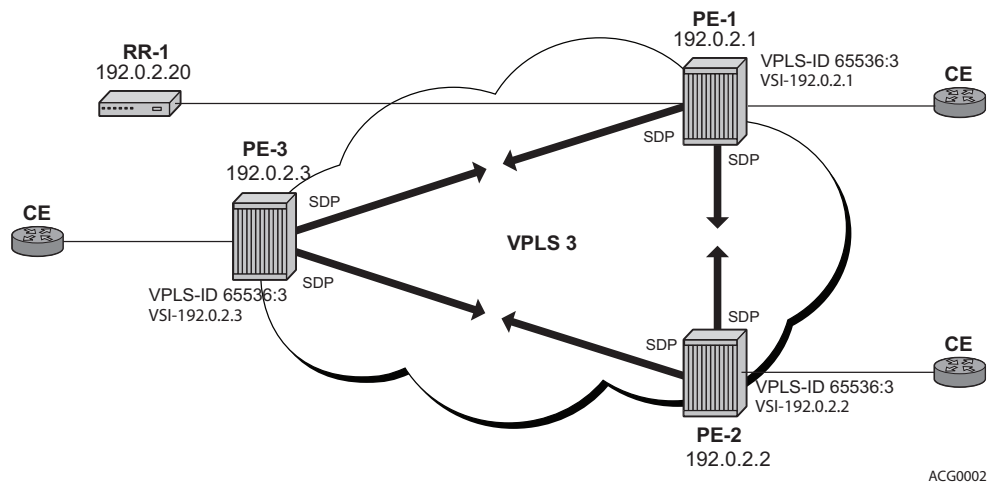


Figure 170: VPLS Instance with Auto-Provisioned SDPs

Figure 170 shows a schematic of a VPLS instance where the SDPs are auto-provisioned. SDPs are instantiated by a PE router using LDP signaling upon receipt of BGP Auto-discovery (BGP-AD) updates from peer PE routers.

PE-1 Configuration:

The following output shows the configuration required for a VPLS service using a pseudowire template configured for auto-provisioning of SDPs.

```
# on PE-1
configure
service
  vpls 3 customer 1 create
    bgp
      route-distinguisher 65536:3
      route-target export target:65536:3 import target:65536:3
      pw-template-binding 2 split-horizon-group "vpls-shg" import-rt "tar-
get:65536:3"
    exit
  exit
  bgp-ad
    vpls-id 65536:3
    vsi-id
      prefix 192.0.2.1
    exit
    no shutdown
  exit
  stp
    shutdown
  exit
```

```

        sap 1/1/4:3.0 create
        exit
        no shutdown
    exit

```

Within the **bgp** context, the pseudowire template is referenced which can be linked to a split-horizon-group and an import route-target, if required.

Within the **bgp-ad** context, the signaling parameters are configured. These are two parameters used by each PE to determine the presence of a given VPLS instance on a PE router. In turn, these are translated into endpoint identifiers for LDP signaling of pseudowires. As previously discussed, these parameters are:

- VPLS-id - a unique identifier of the VPLS instance. Each PE that is a member of a VPLS must share the same VPLS-id. This is inserted as an extended community value in the format AS:n. In this case, the VPLS-id for VPLS 3 is 65536:3. This is a mandatory parameter and if it is not configured it is not possible to enable bgp-ad using no shutdown.
- Virtual Switching Instance (VSI) prefix — This identifies a specific instance of the VPLS. This must be unique within the VPLS instance, and is encoded using the 4 byte dotted decimal notation. Generally the system address is used as the VSI prefix. If this parameter is not configured, then the system address is used automatically.

The VPLS-id and VSI prefix for VPLS 3 on each PE is shown in [Figure 170](#).

The VPLS-id and VSI prefix are concatenated to form a unique VSI-id. In this case, PE-1 has a VSI-id of 65536:3:192.0.2.1. This uniquely identifies the VPLS instance on each individual PE and is advertised as an L2 VPN BGP update.

A BGP-AD update is transmitted to all other PEs via the Route Reflector as follows:

```

*A:PE-1# show router bgp routes l2-vpn rd 65536:3 hunt
=====
BGP Router ID:192.0.2.1      AS:65536      Local AS:65536
=====
Legend -
Status codes  : u - used, s - suppressed, h - history, d - decayed, * - valid
                l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP L2VPN Routes
=====
-----
RIB In Entries
-----
---snipped---
-----
RIB Out Entries
-----
Route Type      : AutoDiscovery
Route Dist.     : 65536:3
Prefix         : 192.0.2.1

```



```

Nexthop      : 192.0.2.1
To           : 192.0.2.20
Res. Nexthop : n/a
Local Pref.  : 100
Aggregator AS : None
Atomic Aggr. : Not Atomic
AIGP Metric  : None
Connector    : None
Community    : target:65536:3 l2-vpn/vrf-imp:65536:3
Cluster      : No Cluster Members
Originator Id : None
Origin       : IGP
AS-Path      : No As-Path
Route Tag    : 0
Neighbor-AS  : N/A
Orig Validation: N/A
Source Class  : 0
Interface Name : NotAvailable
Aggregator    : None
MED           : 0
Peer Router Id : 192.0.2.20
Dest Class    : 0
-----
Routes : 4
=====
A:PE-1#

```

The BGP update shown above is transmitted by PE-1 and has route type Auto Discovery.

In this L2 VPN update, the VPLS-id is encoded as the L2VPN extended community.

The VSI is seen as the prefix. This combination forms the VSI-id and uniquely identifies the VPLS instance within this PE router.

The nexthop is also encoded as the local system IP address, which allows remote PEs to identify a suitable transport tunnel to PE-1 and for the targeted-LDP peer for instantiating the SDP.

As can be seen within the update, the VPLS-id is also used to determine the route target extended community and the route distinguisher.

PE-2 Configuration

On PE-2 create a VPLS Service using pseudowire template 1, with VPLS-id 65536:3 and VSI-id prefix 192.0.2.2 (system IP address).

```

# on PE-2
configure
  service
    vpls 3 customer 1 create
      bgp
        route-distinguisher 65536:3
        route-target export target:65536:3 import target:65536:3
        pw-template-binding 2 split-horizon-group "vpls-shg" import-rt "tar-
get:65536:3"
      exit
    exit
  bgp-ad
    vpls-id 65536:3
    vsi-id

```

VPLS PE Configuration

```
        prefix 192.0.2.2
    exit
    no shutdown
exit
stp
    shutdown
exit
sap 1/1/4:3.0 create
exit
no shutdown
exit
```

PE-3 Configuration

Create a VPLS Instance on PE-3: VPLS-id is the same as that of PE-1 and PE-2, with VSI-id of 192.0.2.3 (system IP address).

```
# on PE-3
configure
    service
        vpls 3 customer 1 create
            bgp
                route-distinguisher 65536:3
                route-target export target:65536:3 import target:65536:3
                pw-template-binding 2 split-horizon-group "vpls-shg" import-rt "tar-
get:65536:3"
            exit
        exit
        bgp-ad
            vpls-id 65536:3
            vsi-id
                prefix 192.0.2.3
            exit
            no shutdown
        exit
        stp
            shutdown
        exit
        sap 1/1/4:3.0 create
        exit
        no shutdown
    exit
```

PE-1 Service Operation Verification

Verify that the service is operationally up on PE-1.

```
*A:PE-1# show service id 3 base
=====
Service Basic Information
=====
Service Id       : 3                Vpn Id           : 0
Service Type     : VPLS
Name             : (Not Specified)
Description      : (Not Specified)
```

```

Customer Id       : 1                      Creation Origin  : manual
Last Status Change: 03/16/2015 12:58:03
Last Mgmt Change  : 03/16/2015 12:59:05
Etree Mode       : Disabled
Admin State      : Up                      Oper State       : Up
MTU              : 1514                    Def. Mesh VC Id   : 3
SAP Count        : 1                      SDP Bind Count    : 2
Snd Flush on Fail: Disabled               Host Conn Verify  : Disabled
Propagate MacFlush: Disabled              Per Svc Hashing   : Disabled
Allow IP Intf Bind: Disabled
Def. Gateway IP   : None
Def. Gateway MAC  : None
Temp Flood Time   : Disabled              Temp Flood        : Inactive
Temp Flood Chg Cnt: 0
VSD Domain       : <none>
SPI load-balance  : Disabled

```

Service Access & Destination Points

Identifier	Type	AdmMTU	OprMTU	Adm	Opr
sap:1/1/4:3.0	qinq	1522	1522	Up	Up
sdp:17406:4294967294 SB(192.0.2.3)	BgpAd	0	1556	Up	Up
sdp:17407:4294967295 SB(192.0.2.2)	BgpAd	0	1556	Up	Up

=====

*A:PE-1#

As seen from the output, the service is operationally up, with the SAPs and SDPs also up. The **SB** flag indicates that the SDP is of type spoke BGP.

BGP is used to discover the VPLS endpoints and exchange network reachability information. LDP is used to signal the pseudowires between the PEs.

LDP signaling occurs when each PE has discovered the endpoints of the VPLS instance. This compares with the use of the provisioned virtual-circuit IDs used in an NMS provisioned VPLS instances as per RFC 4762.

Verification of the ability of PE-1 to reach the other PE routers with VSIs within the VPLS instance can be seen from the Layer 2 routing table as follows:

```

*A:PE-1# show service l2-route-table bgp-ad
=====
Services: L2 Route Information - Summary
=====
Svc Id   L2-Routes (RD-Prefix)      Next Hop      Origin
        Sdp Bind Id          PW Temp Id
-----
3        *65536:3-192.0.2.2      192.0.2.2     BGP-L2
        17407:4294967295      2
3        *65536:3-192.0.2.3      192.0.2.3     BGP-L2
        17406:4294967294      2
-----
No. of L2 Route Entries: 2
=====
*A:PE-1#

```

This output shows the presence of the signaled pseudowire SDPs. SDPs from PE-1 to PE-2 and PE-3 are signaled using LDP Forwarding Equivalence Class (FEC) Element 129.

Each PE router uses targeted LDP to signal the local and remote endpoints. If there is an endpoint match, then SDPs are instantiated. This compares with the use of LDP for NMS provisioned SDPs, which uses virtual-circuit IDs to signal pseudowires using LDP FEC Element 128.

In order to signal the SDPs, the following parameters are required:

1. Attachment Group Identifier (AGI): this is used to carry the VPLS-id of the local PE router VPLS instance. The VPLS-id must be identical for all PEs in the same VPLS instance.
2. Source Attachment Individual Identifier (SAII) and Target Attachment Individual Identifier (TAII): These use AII type 1 (RFC 4446) and are used to carry the NLRI (VSI-id minus the RD) of the remote PE router VPLS instance.

The AGI for each PE must be identical. SAII and TAIID must be different.

The following shows the service LDP bindings for VPLS 3 on PE-1:

```
*A:PE-1# show router ldp bindings services service-id 3
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
              (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
       S - Status Signaled Up,   D - Status Signaled Down
       E - Epipe Service, V - VPLS Service, M - Mirror Service
       A - Apipe Service, F - Fpipe Service, I - IES Service, R - VPRN service
       P - Ipipe Service, WP - Label Withdraw Pending, C - Cpipe Service
       BU - Alternate For Fast Re-Route, TLV - (Type, Length: Value)
=====
LDP Service FEC 128 Bindings
=====
Type          VCId      SDPId      IngLbl  LMTU
Peer          SvcId      EgrLbl  RMTU
-----
No Matching Entries Found
=====
LDP Service FEC 129 Bindings
=====
SAII          AGII          IngLbl  LMTU
TAII          Type          EgrLbl  RMTU
Peer          SvcId      SDPId
-----
192.0.2.1      null          131064U  1500
192.0.2.2      V-Eth        131062S  1500
192.0.2.2:0    3            17407
192.0.2.1      null          131063U  1500
192.0.2.3      V-Eth        131064S  1500
192.0.2.3:0    3            17406
-----
```

```
No. of FEC 129s: 2
```

```
=====
*A:PE-1#
```

This shows the two T-LDP bindings for PE-1 towards PE-2 and PE-3 for VPLS 3.

To list the actual SdpIds use the following command:

```
*A:PE-1# show service id 3 sdp
```

```
=====
Services: Service Destination Points
```

```
=====
SdpId          Type Far End addr   Adm    Opr      I.Lbl    E.Lbl
-----
17406:4294967294 Bgp* 192.0.2.3      Up     Up       131063   131064
17407:4294967295 Bgp* 192.0.2.2      Up     Up       131064   131062
-----
```

```
Number of SDPs : 2
```

```
=====
* indicates that the corresponding row element may have been truncated.
```

```
*A:PE-1#
```

Then the actual AGI, SAII and TAII values are seen in the detailed SDP output.

AGI — 65536:3

SAII — Local system IP address 192.0.2.1

TAII — Remote system IP address 192.0.2.2 or 192.0.2.3

```
*A:PE-1# show service id 3 sdp 17407:4294967295 detail
```

```
=====
Service Destination Point (Sdp Id : 17407:4294967295) Details
```

```
=====
Sdp Id 17407:4294967295 - (192.0.2.2)
```

```
-----
Description      : (Not Specified)
```

```
SDP Id           : 17407:4294967295
```

```
Type            : BgpAd
```

```
PW-Template Id   : 2
```

```
AGI              : 65536:3
```

```
SDP Bind Source  : bgp-12vpn
```

```
Local AII        : 192.0.2.1
```

```
Remote AII       : 192.0.2.2
```

```
Split Horiz Grp  : vpls-shg
```

```
---snipped---
```

The ingress and egress labels can also be seen from the SDP bindings from the service:

```
*A:PE-1# show service id 3 sdp
=====
Services: Service Destination Points
=====
SdpId              Type Far End addr    Adm    Opr      I.Lbl    E.Lbl
-----
17406:4294967294 Bgp* 192.0.2.3      Up     Up       131063   131064
17407:4294967295 Bgp* 192.0.2.2      Up     Up       131064   131062
-----
Number of SDPs : 2
-----
* indicates that the corresponding row element may have been truncated.
*A:PE-1#
```

The SDPs are auto-provisioned SDPs, like SDP 17407 towards PE-2 and 17406 towards PE-3. The label bindings from the SDP and LDP binding outputs are identical.

PE-2 Service Operation Verification

For completeness, verify the service is operationally up on PE-2.

```
*A:PE-2# show service id 3 base
=====
Service Basic Information
=====
Service Id       : 3                Vpn Id       : 0
Service Type     : VPLS
Name             : (Not Specified)
Description      : (Not Specified)
Customer Id      : 1                Creation Origin : manual
Last Status Change: 03/16/2015 13:44:39
Last Mgmt Change : 03/16/2015 13:45:24
Etree Mode       : Disabled
Admin State      : Up               Oper State     : Up
MTU              : 1514             Def. Mesh VC Id : 3
SAP Count        : 1               SDP Bind Count : 2
Snd Flush on Fail : Disabled        Host Conn Verify : Disabled
Propagate MacFlush: Disabled        Per Svc Hashing  : Disabled
Allow IP Intf Bind: Disabled
Def. Gateway IP   : None
Def. Gateway MAC  : None
Temp Flood Time   : Disabled        Temp Flood      : Inactive
Temp Flood Chg Cnt: 0
VSD Domain        : <none>
SPI load-balance  : Disabled

-----
Service Access & Destination Points
-----
Identifier                                     Type      AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/4:3.0                                qinq      1522    1522    Up   Up
sdp:17406:4294967294 SB(192.0.2.3)            BgpAd     0       1556    Up   Up
sdp:17407:4294967295 SB(192.0.2.1)            BgpAd     0       1556    Up   Up
=====
*A:PE-2#
*A:PE-2# show service l2-route-table bgp-ad
=====
Services: L2 Route Information - Summary
=====
Svc Id   L2-Routes (RD-Prefix)                Next Hop      Origin
        Sdp Bind Id                        PW Temp Id
-----
3        *65536:3-192.0.2.1                192.0.2.1     BGP-L2
        17407:4294967295                    2
3        *65536:3-192.0.2.3                192.0.2.3     BGP-L2
        17406:4294967294                    2
-----
No. of L2 Route Entries: 2
=====
*A:PE-2#
*A:PE-2# show router ldp bindings services service-id 3
=====
LDP Bindings (IPv4 LSR ID 192.0.2.2:0)
```

VPLS PE Configuration

```

(IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
       S - Status Signaled Up, D - Status Signaled Down
       E - Epipe Service, V - VPLS Service, M - Mirror Service
       A - Apipe Service, F - Fpipe Service, I - IES Service, R - VPRN service
       P - Ipipe Service, WP - Label Withdraw Pending, C - Cpipe Service
       BU - Alternate For Fast Re-Route, TLV - (Type, Length: Value)
=====
LDP Service FEC 128 Bindings
=====
Type          VCId      SDPIId      IngLbl  LMTU
Peer          SvcId      EgrLbl  RMTU
-----
No Matching Entries Found
=====
LDP Service FEC 129 Bindings
=====
SAII          AGII          IngLbl  LMTU
TAII          Type          EgrLbl  RMTU
Peer          SvcId      SDPIId
-----
192.0.2.2      null          131062U  1500
192.0.2.1      V-Eth        131064S  1500
192.0.2.1:0    3            17407

192.0.2.2      null          131061U  1500
192.0.2.3      V-Eth        131064S  1500
192.0.2.3:0    3            17406

-----
No. of FEC 129s: 2
=====
*A:PE-2#
*A:PE-2# show service id 3 sdp
=====
Services: Service Destination Points
=====
SdpId          Type Far End addr  Adm   Opr    I.Lbl    E.Lbl
-----
17406:4294967294 Bgp* 192.0.2.3    Up    Up     131061    131063
17407:4294967295 Bgp* 192.0.2.1    Up    Up     131062    131064
-----
Number of SDPs : 2
=====
* indicates that the corresponding row element may have been truncated.
*A:PE-2#

```


PE-3 Service Operation Verification

Verify service is operationally up on PE-3.

```
*A:PE-3# show service id 3 base
=====
Service Basic Information
=====
Service Id      : 3                Vpn Id      : 0
Service Type    : VPLS
Name            : (Not Specified)
Description     : (Not Specified)
Customer Id     : 1                Creation Origin : manual
Last Status Change: 03/16/2015 14:45:15
Last Mgmt Change  : 03/16/2015 14:45:49
Etree Mode      : Disabled
Admin State     : Up               Oper State    : Up
MTU             : 1514            Def. Mesh VC Id : 3
SAP Count       : 1              SDP Bind Count : 2
Snd Flush on Fail : Disabled      Host Conn Verify : Disabled
Propagate MacFlush: Disabled      Per Svc Hashing  : Disabled
Allow IP Intf Bind: Disabled
Def. Gateway IP  : None
Def. Gateway MAC : None
Temp Flood Time  : Disabled        Temp Flood     : Inactive
Temp Flood Chg Cnt: 0
VSD Domain      : <none>
SPI load-balance : Disabled

-----
Service Access & Destination Points
-----
Identifier                                     Type      AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/4:3.0                                qinq      1522    1522    Up   Up
sdp:17406:4294967294 SB(192.0.2.2)           BgpAd     0        1556    Up   Up
sdp:17407:4294967295 SB(192.0.2.1)           BgpAd     0        1556    Up   Up
=====
*A:PE-3#
*A:PE-3# show service l2-route-table bgp-ad
=====
Services: L2 Route Information - Summary
=====
Svc Id   L2-Routes (RD-Prefix)                Next Hop      Origin
        Sdp Bind Id                        PW Temp Id
-----
3        *65536:3-192.0.2.1                192.0.2.1     BGP-L2
        17407:4294967295                    2
3        *65536:3-192.0.2.2                192.0.2.2     BGP-L2
        17406:4294967294                    2
-----
No. of L2 Route Entries: 2
=====
*A:PE-3#
*A:PE-3# show router ldp bindings services service-id 3
=====
LDP Bindings (IPv4 LSR ID 192.0.2.3:0)
```

VPLS PE Configuration

```

(IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
       S - Status Signaled Up, D - Status Signaled Down
       E - Epipe Service, V - VPLS Service, M - Mirror Service
       A - Apipe Service, F - Fpipe Service, I - IES Service, R - VPRN service
       P - Ipipe Service, WP - Label Withdraw Pending, C - Cpipe Service
       BU - Alternate For Fast Re-Route, TLV - (Type, Length: Value)
=====
LDP Service FEC 128 Bindings
=====
Type          VCId      SDPIId      IngLbl  LMTU
Peer          SvcId      EgrLbl      RMTU
-----
No Matching Entries Found
=====
LDP Service FEC 129 Bindings
=====
SAII          AGII          IngLbl      LMTU
TAII          Type          EgrLbl      RMTU
Peer          SvcId      SDPIId
-----
192.0.2.3      null          131064U     1500
192.0.2.1      V-Eth        131063S     1500
192.0.2.1:0    3            17407
-----
192.0.2.3      null          131063U     1500
192.0.2.2      V-Eth        131061S     1500
192.0.2.2:0    3            17406
-----
No. of FEC 129s: 2
=====
*A:PE-3#
*A:PE-3# show service id 3 sdp
=====
Services: Service Destination Points
=====
SdpId          Type Far End addr  Adm   Opr    I.Lbl    E.Lbl
-----
17406:4294967294 Bgp* 192.0.2.2      Up    Up     131063    131061
17407:4294967295 Bgp* 192.0.2.1      Up    Up     131064    131063
-----
Number of SDPs : 2
=====
* indicates that the corresponding row element may have been truncated.
*A:PE-3#

```

BGP AD using Pre-Provisioned SDPs

It is possible to configure BGP-AD instances that use RSVP transport tunnels. In this case, the LSPs and SDPs must be manually created.

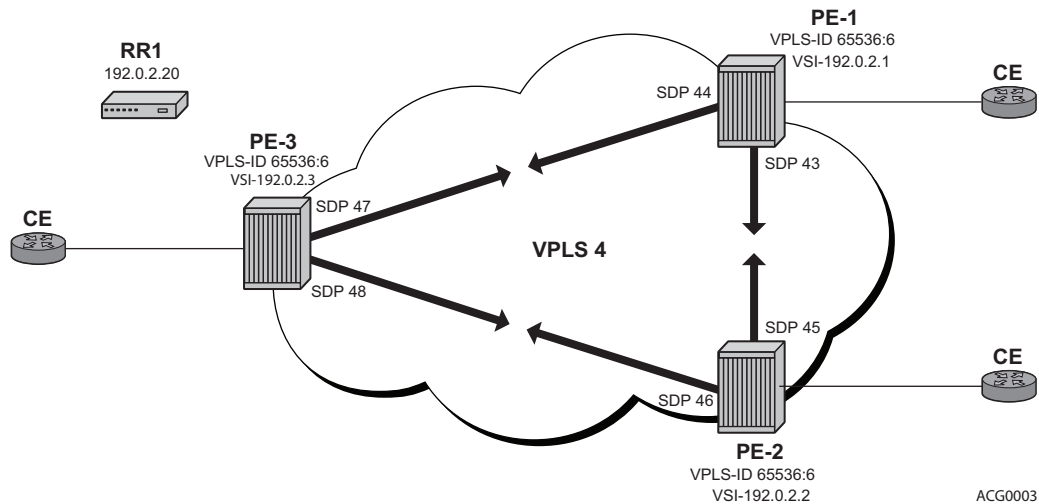


Figure 171: VPLS Instance using Pre-Provisioned SDPs

Figure 171 shows a VPLS instance configured across three Provider Edge routers as before.

The SDP configurations for the three PEs are shown below:

SDPs on PE-1

```
configure service
  sdp 43 mpls create
    far-end 192.0.2.2
    lsp "LSP-PE-1-PE-2"
    keep-alive
    shutdown
  exit
  no shutdown
exit
sdp 44 mpls create
  far-end 192.0.2.3
  lsp "LSP-PE-1-PE-3"
  keep-alive
  shutdown
exit
  no shutdown
exit
exit
```

SDPs on PE-2

```
configure service
  sdp 45 mpls create
    far-end 192.0.2.1
    lsp "LSP-PE-2-PE-1"
    keep-alive
    shutdown
  exit
  no shutdown
exit
sdp 46 mpls create
  far-end 192.0.2.3
  lsp "LSP-PE-2-PE-3"
  keep-alive
  shutdown
exit
  no shutdown
exit
exit
```

SDPs on PE-3

```
configure service
  sdp 47 mpls create
    far-end 192.0.2.1
    lsp "LSP-PE-3-PE-1"
    keep-alive
    shutdown
  exit
  no shutdown
exit
sdp 48 mpls create
  far-end 192.0.2.2
  lsp "LSP-PE-3-PE-2"
  keep-alive
  shutdown
exit
  no shutdown
exit
exit
```

The pw-template that is to be used within each VPLS instance must be provisioned on all PEs and must use the keyword **use-provisioned-sdp**. The pw-template looks like:

```
A:PE-1# configure service
      pw-template 3 use-provisioned-sdp create
      exit
exit
```

This configuration must be repeated on both PE-2 and PE-3.

The following output shows the configuration required for a VPLS service using a pseudowire template configured for pre-provisioned RSVP SDPs.

```
# on PE-1
configure
  service
    vpls 4 customer 1 create
      bgp
        route-distinguisher 65536:4
        route-target export target:65536:4 import target:65536:4
        pw-template-binding 3 split-horizon-group "vpls-shg" import-rt "tar-
get:65536:4"
      exit
    exit
  bgp-ad
    vpls-id 65536:4
    vsi-id
      prefix 192.0.2.1
    exit
    no shutdown
  exit
  stp
    shutdown
  exit
  sap 1/1/4:4.0 create
  exit
  no shutdown
exit
```

Similarly, on PE-2 the configuration is shown below:

```
# on PE-2
configure
  service
    vpls 4 customer 1 create
      bgp
        route-distinguisher 65536:4
        route-target export target:65536:4 import target:65536:4
        pw-template-binding 3 split-horizon-group "vpls-shg" import-rt "tar-
get:65536:4"
      exit
    exit
  bgp-ad
    vpls-id 65536:4
    vsi-id
```

VPLS PE Configuration

```
        prefix 192.0.2.2
    exit
    no shutdown
exit
stp
    shutdown
exit
sap 1/1/4:4.0 create
exit
no shutdown
exit
```

On PE-3:

```
# on PE-3
configure
    service
        vpls 4 customer 1 create
            bgp
                route-distinguisher 65536:4
                route-target export target:65536:4 import target:65536:4
                pw-template-binding 3 split-horizon-group "vpls-shg" import-rt "tar-
get:65536:4"
            exit
        exit
        bgp-ad
            vpls-id 65536:4
            vsi-id
                prefix 192.0.2.4
            exit
            no shutdown
        exit
        stp
            shutdown
        exit
        sap 1/1/4:4.0 create
        exit
        no shutdown
    exit
```

Verify that the service is operationally up on PE-1.

```
*A:PE-1# show service id 4 base
=====
Service Basic Information
=====
Service Id       : 4                Vpn Id           : 0
Service Type     : VPLS
Name             : (Not Specified)
Description      : (Not Specified)
Customer Id      : 1                Creation Origin   : manual
Last Status Change: 03/16/2015 13:01:09
Last Mgmt Change  : 03/16/2015 13:02:03
Etree Mode       : Disabled
Admin State      : Up               Oper State        : Up
MTU              : 1514             Def. Mesh VC Id   : 4
```

```

SAP Count          : 1
Snd Flush on Fail  : Disabled
Propagate MacFlush : Disabled
Allow IP Intf Bind : Disabled
Def. Gateway IP    : None
Def. Gateway MAC   : None
Temp Flood Time    : Disabled
Temp Flood Chg Cnt : 0
VSD Domain         : <none>
SPI load-balance   : Disabled
SDP Bind Count     : 2
Host Conn Verify   : Disabled
Per Svc Hashing    : Disabled
Temp Flood         : Inactive

```

Service Access & Destination Points

Identifier	Type	AdmMTU	OprMTU	Adm	Opr
sap:1/1/4:4.0	qinq	1522	1522	Up	Up
sdp:43:4294967293 S(192.0.2.2)	BgpAd	0	1556	Up	Up
sdp:44:4294967292 S(192.0.2.3)	BgpAd	0	1556	Up	Up

=====

*A:PE-1#

Note that the SDP identifiers are the pre-provisioned SDPs, i.e. SDP 43 and 44.

For completeness, verify the service is operationally up on PE-2.

*A:PE-2# show service id 4 base

=====

Service Basic Information

```

=====
Service Id          : 4
Service Type        : VPLS
Name                : (Not Specified)
Description          : (Not Specified)
Customer Id         : 1
Creation Origin      : manual
Last Status Change  : 03/16/2015 13:47:43
Last Mgmt Change    : 03/16/2015 13:48:35
Etree Mode          : Disabled
Admin State         : Up
Oper State           : Up
MTU                  : 1514
Def. Mesh VC Id     : 4
SAP Count           : 1
SDP Bind Count       : 2
Snd Flush on Fail   : Disabled
Host Conn Verify     : Disabled
Propagate MacFlush  : Disabled
Per Svc Hashing     : Disabled
Allow IP Intf Bind  : Disabled
Def. Gateway IP     : None
Def. Gateway MAC    : None
Temp Flood Time     : Disabled
Temp Flood          : Inactive
Temp Flood Chg Cnt  : 0
VSD Domain          : <none>
SPI load-balance    : Disabled

```

Service Access & Destination Points

Identifier	Type	AdmMTU	OprMTU	Adm	Opr
sap:1/1/4:4.0	qinq	1522	1522	Up	Up
sdp:45:4294967293 S(192.0.2.1)	BgpAd	0	1556	Up	Up
sdp:46:4294967292 S(192.0.2.3)	BgpAd	0	1556	Up	Up

VPLS PE Configuration

```
=====
*A:PE-2#
```

Verify service is operational on PE-3.

```
*A:PE-3# show service id 4 base
```

```
=====
Service Basic Information
```

```
=====
Service Id      : 4                      Vpn Id      : 0
Service Type    : VPLS
Name            : (Not Specified)
Description     : (Not Specified)
Customer Id     : 1                      Creation Origin : manual
Last Status Change: 03/16/2015 14:48:20
Last Mgmt Change : 03/16/2015 14:48:39
Etree Mode     : Disabled
Admin State     : Up                    Oper State    : Up
MTU             : 1514                  Def. Mesh VC Id : 4
SAP Count      : 1                     SDP Bind Count : 2
Snd Flush on Fail : Disabled            Host Conn Verify : Disabled
Propagate MacFlush: Disabled            Per Svc Hashing  : Disabled
Allow IP Intf Bind: Disabled
Def. Gateway IP : None
Def. Gateway MAC : None
Temp Flood Time : Disabled              Temp Flood     : Inactive
Temp Flood Chg Cnt: 0
VSD Domain     : <none>
SPI load-balance : Disabled
```

```
-----
Service Access & Destination Points
```

```
-----
Identifier                                     Type      AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/4:4.0                                qinq      1522    1522    Up    Up
sdp:47:4294967293 S(192.0.2.1)               BgpAd      0       1556    Up    Up
sdp:48:4294967292 S(192.0.2.2)               BgpAd      0       1556    Up    Up
=====
```

```
*A:PE-3#
```


Conclusion

BGP-Auto discovery coupled with LDP pseudowire signaling allows the delivery of L2 VPN services to customers where BGP is commonly used. This example shows the configuration of BGP-Auto discovery together with the associated show outputs which can be used for verification and troubleshooting.

Multi-Chassis Endpoint for VPLS Active/Standby Pseudowire

In This Chapter

This section provides information about multi-chassis endpoint for VPLS active/standby pseudowire.

Topics in this section include:

- [Applicability on page 1160](#)
- [Overview on page 1161](#)
- [Configuration on page 1165](#)
- [Conclusion on page 1188](#)

Applicability

The examples covered in this section are applicable to all 7x50 SR series and were tested on Release 13.0.R3.

Multi-chassis endpoint peers must be the same chassis type.

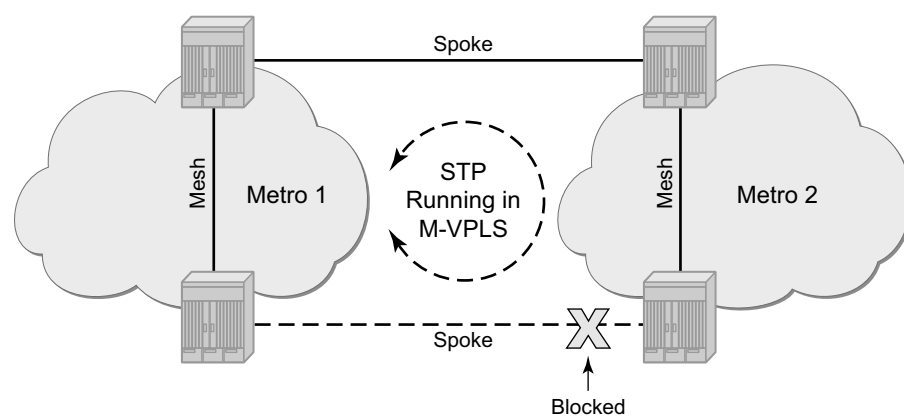
Overview

When implementing a large VPLS, one of the limiting factors is the number of T-LDP sessions required for the full mesh of SDPs. Mesh SDPs are required between all PEs participating in the VPLS with a full mesh of T-LDP sessions.

This solution is not scalable, as the number of sessions grows more rapidly than the number of participating PEs. Several options exist to reduce the number of T-LDP sessions required in a large VPLS.

The first option is hierarchical VPLS (H-VPLS) with spoke SDPs. By using spoke SDPs between two clouds of fully meshed PEs, any-to-any T-LDP sessions for all participating PEs are not required.

However, if spoke SDP redundancy is required, STP must be used to avoid a loop in the VPLS. Management VPLS can be used to reduce the number of STP instances and separate customer and STP traffic ([Figure 172](#)).



OSSG432

Figure 172: H-VPLS with STP

VPLS pseudowire redundancy provides H-VPLS redundant spoke connectivity. The active spoke is in forwarding state, while the standby spoke is in blocking state. Hence, STP is not needed anymore to break the loop, as illustrated in [Figure 173](#).

However, the PE implementing the active and standby spokes represents a single point of failure in the network.

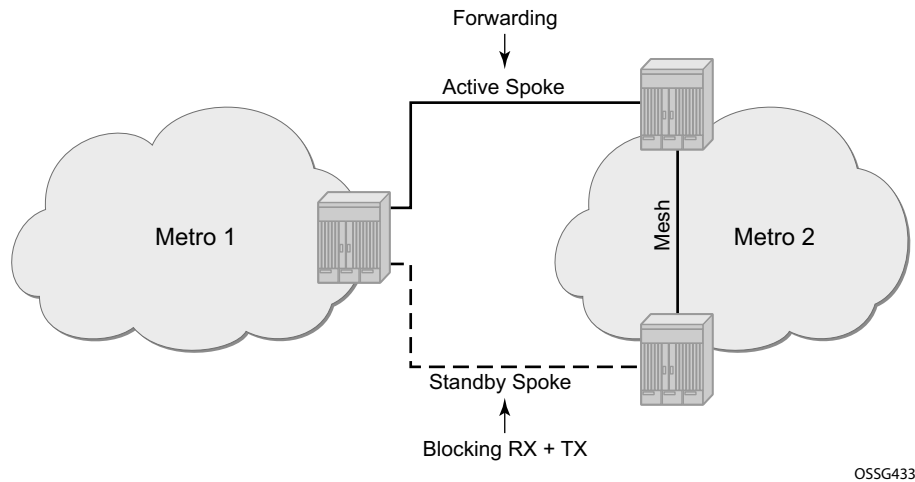


Figure 173: VPLS Pseudowire Redundancy

Multi-chassis endpoint (MC-EP) for VPLS active/standby pseudowire expands on the VPLS pseudowire redundancy and allows the removal of the single point of failure.

There is only one spoke in forwarding state, all standby spokes are in blocking state. Mesh and square resiliency are supported.

Mesh resiliency can protect against simultaneous node failure in the core and in the MC-EP (double failure), but requires more SDPs (and thus more T-LDP sessions). Mesh resiliency is illustrated in [Figure 174](#).

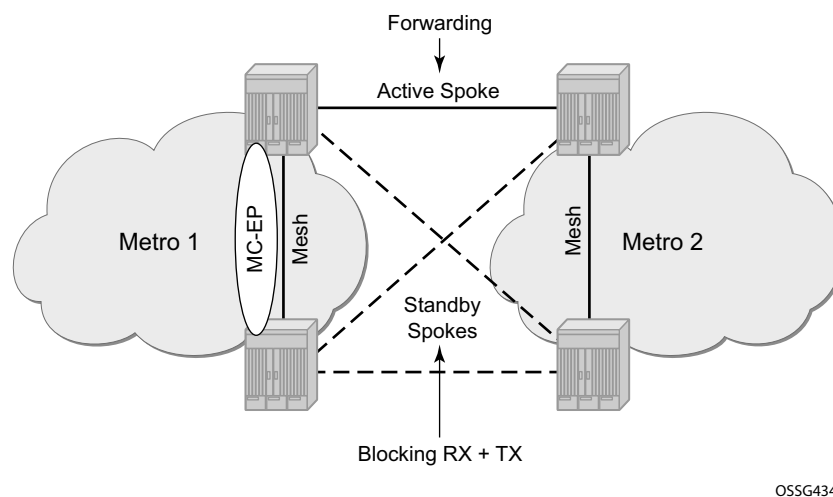


Figure 174: Multi-Chassis Endpoint with Mesh Resiliency

Square resiliency provides single failure node protection, and requires less SDPs (and thus less T-LDP sessions). Square resiliency is illustrated in [Figure 175](#).

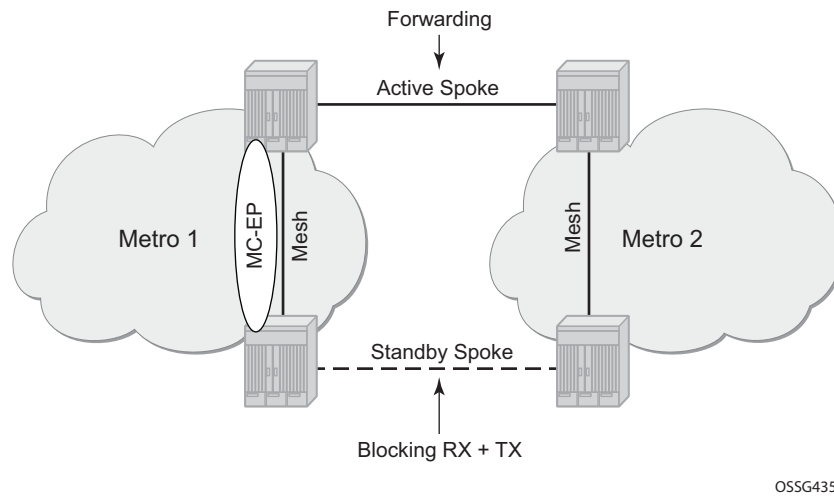


Figure 175: Multi-Chassis Endpoint with Square Resiliency

Network Topology

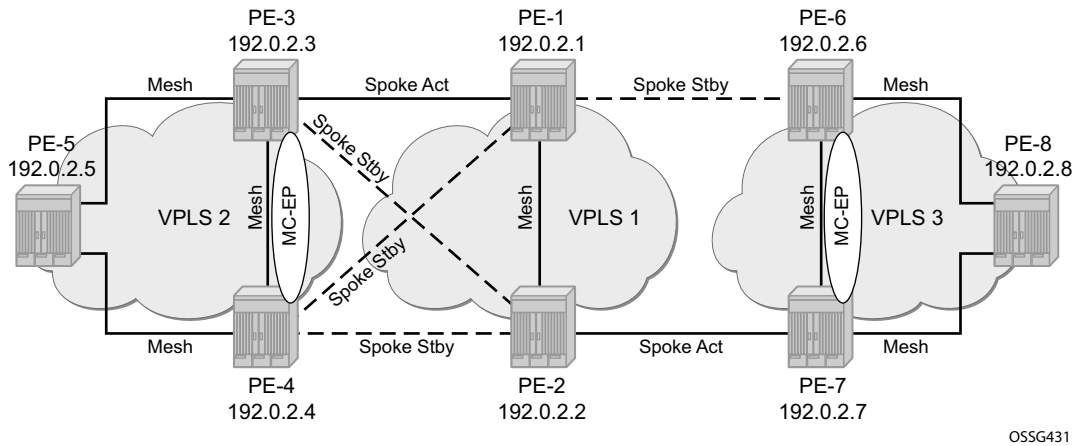


Figure 176: Network Topology

The network topology is displayed in [Figure 176](#).

The setup consists of:

- Two core nodes (PE-1 and PE-2), and three nodes for each metro area (PE-3, PE-4, PE-5 and PE-6, PE-7, PE-8, respectively).
- VPLS 1 is the core VPLS, used to interconnect the two metro areas represented by VPLS 2 and VPLS 3.
- VPLS 2 will be connected to the core VPLS in mesh resiliency.
- VPLS 3 will be connected to the core VPLS in square resiliency.

Note that three separate VPLS identifiers are used for clarity, however, the same identifier could be used for each. For interoperation, only the same VC-ID is required to be used on both ends of the spoke SDPs.

The following configuration tasks should be done first:

- ISIS or OSPF throughout the network.
- RSVP or LDP-signaled LSPs over the paths used for mesh/spoke SDPs.

Configuration

SDP Configuration

On each PE, SDPs are created to match the topology described in [Figure 176](#).

The convention for the SDP naming is: XY where X is the originating node and Y the target node.

An example of the SDP configuration in PE-3 (using LDP):

```
A:PE-3# configure service
      sdp 31 mpls create
          far-end 192.0.2.1
          ldp
          no shutdown
      exit
      sdp 32 mpls create
          far-end 192.0.2.2
          ldp
          no shutdown
      exit
      sdp 34 mpls create
          far-end 192.0.2.4
          ldp
          no shutdown
      exit
      sdp 35 mpls create
          far-end 192.0.2.5
          ldp
          no shutdown
      exit
```

Verification of the SDPs on PE-3:

```
*A:PE-3# show service sdp
=====
Services: Service Destination Points
=====
SdpId  AdmMTU  OprMTU  Far End      Adm  Opr      Del    LSP    Sig
-----
31      0       1556    192.0.2.1    Up   Up       MPLS   L      TLDP
32      0       1556    192.0.2.2    Up   Up       MPLS   L      TLDP
34      0       1556    192.0.2.4    Up   Up       MPLS   L      TLDP
35      0       1556    192.0.2.5    Up   Up       MPLS   L      TLDP
-----
Number of SDPs : 4
-----
Legend: R = RSVP, L = LDP, B = BGP, M = MPLS-TP, n/a = Not Applicable
       I = SR-ISIS, O = SR-OSPF
=====
*A:PE-3#
```

Full Mesh VPLS Configuration

Next, three fully meshed VPLS services are configured.

- VPLS 1 is the core VPLS, on PE-1 and PE-2
- VPLS 2 is the metro 1 VPLS, on PE-3, PE-4 and PE-5
- VPLS 3 is the metro 2 VPLS, on PE-6, PE-7 and PE-8

On PE-1 (similar configuration on PE-2):

```
configure service
  vpls 1 customer 1 create
    description "core VPLS"
    stp
      shutdown
    exit
    mesh-sdp 12:1 create
    exit
    no shutdown
  exit
exit
```

On PE-3 (similar configuration on PE-4 and PE-5):

```
configure service
  description "Metro 1 VPLS"
  vpls 2 customer 1 create
    stp
      shutdown
    exit
    mesh-sdp 34:2 create
    exit
    mesh-sdp 35:2 create
    exit
    no shutdown
  exit
exit
```

On PE-6 (similar configuration on PE-7 and PE-8):

```
configure service
  description "Metro 2 VPLS"
  vpls 3 customer 1 create
    stp
      shutdown
    exit
    mesh-sdp 67:3 create
    exit
    mesh-sdp 68:3 create
    exit
```

```

        no shutdown
    exit
exit

```

Verification of the VPLS:

- The service must be operationally up.
- All mesh SDPs must be up in the VPLS service.

On PE-6 (similar on other nodes):

```

*A:PE-6# show service id 3 base
=====
Service Basic Information
=====
Service Id       : 3                Vpn Id           : 0
Service Type     : VPLS
Name             : (Not Specified)
Description      : (Not Specified)
Customer Id      : 1                Creation Origin  : manual
Last Status Change: 07/01/2015 11:51:53
Last Mgmt Change : 07/01/2015 11:51:42
Etree Mode      : Disabled
Admin State      : Up               Oper State       : Up
MTU              : 1514             Def. Mesh VC Id  : 3
SAP Count        : 0               SDP Bind Count   : 2
Snd Flush on Fail : Disabled        Host Conn Verify : Disabled
Propagate MacFlush: Disabled        Per Svc Hashing  : Disabled
Allow IP Intf Bind: Disabled
Def. Gateway IP   : None
Def. Gateway MAC  : None
Temp Flood Time   : Disabled        Temp Flood       : Inactive
Temp Flood Chg Cnt: 0
VSD Domain        : <none>
SPI load-balance  : Disabled

-----
Service Access & Destination Points
-----
Identifier                                     Type      AdmMTU  OprMTU  Adm  Opr
-----
sdp:67:3 M(192.0.2.7)                         Mesh       0      1556   Up   Up
sdp:68:3 M(192.0.2.8)                         Mesh       0      1556   Up   Up
=====
*A:PE-6#

```

Multi-Chassis Configuration

Multi-chassis will be configured on the MC peers PE-3, PE-4 and PE-6, PE-7. The peer system address is configured, and mc-endpoint will be enabled.

On PE-3 (similar configuration on PE-4, PE-6 and PE-7):

```
configure redundancy
  multi-chassis
    peer 192.0.2.4
    mc-endpoint
      bfd-enable
    exit
  exit
exit
```

Verification of the multi-chassis synchronization:

If the multi-chassis synchronization fails, both nodes will fall back to single-chassis mode. In that case, two spoke SDPs could become active at the same time. It is important to verify the multi-chassis synchronization before enabling the redundant spoke SDPs.

```
*A:PE-3# show redundancy multi-chassis mc-endpoint peer 192.0.2.4
=====
Multi-Chassis MC-Endpoint
=====
Peer Addr       : 192.0.2.4           Peer Name       :
Admin State     : up                 Oper State      : up
Last State chg  : 07/01/2015 12:07:45 Source Addr      : 0.0.0.0
System Id       : 02:09:ff:00:00:00  Sys Priority    : 0
Keep Alive Intvl: 10                 Hold on Nbr Fail : 3
Passive Mode    : disabled           Psv Mode Oper   : No
Boot Timer      : 300                 BFD             : disabled
Last update     : 07/01/2015 11:59:12 MC-EP Count      : 1
=====
*A:PE-3#
```

Mesh Resiliency Configuration

PE-3 and PE-4 will be connected to the core VPLS in mesh resiliency.

- First an endpoint is configured.
- The “no suppress-standby-signaling” is needed to block the standby spoke SDP.
- The multi-chassis endpoint peer is configured. The mc-endpoint ID must match between the two peers.

On PE-3 (similar on PE-4):

```
configure service
  vpls 2
    endpoint "CORE" create
    no suppress-standby-signaling
    mc-endpoint 1
      mc-ep-peer 192.0.2.4
    exit
  exit
```

Two spoke SDPs are configured on each peer of the multi-chassis to the two nodes of the core VPLS (mesh resiliency). Each spoke SDP refers to the endpoint CORE.

The precedence is defined on the spoke SDPs as follows:

- Spoke SDP 31 on PE-3 will be active. It is configured as primary (= precedence 0).
- Spoke SDP 32 on PE-3 will be the first backup. It is configured with precedence 1.
- Spoke SDP 41 on PE-4 will be the second backup. It is configured with precedence 2.
- Spoke SDP 42 on PE-4 will be the third backup. It is configured with precedence 3.

On PE-3 (similar on PE-4):

```
configure service
  vpls 2
    spoke-sdp 31:1 endpoint "CORE" create
      precedence primary
    exit
    spoke-sdp 32:1 endpoint "CORE" create
      precedence 1
    exit
  exit
```

Verification of the spoke SDPs:

On PE-3 and PE-4, the spoke SDPs must be up.

```
*A:PE-3# show service id 2 sdp
=====
Services: Service Destination Points
=====
SdpId          Type Far End addr   Adm   Opr    I.Lbl   E.Lbl
-----
31:1           Spok 192.0.2.1      Up    Up     262135  262131
32:1           Spok 192.0.2.2      Up    Up     262134  262131
34:2           Mesh 192.0.2.4      Up    Up     262133  262135
35:2           Mesh 192.0.2.5      Up    Up     262132  262135
-----
Number of SDPs : 4
-----
=====
*A:PE-3#
```

The endpoints on PE-3 and PE-4 can be verified. One spoke SDP is in Tx-Active mode (31 on PE-1 because it is configured as primary).

```
*A:PE-3# show service id 2 endpoint "CORE" | match "Tx Active"
Tx Active (SDP)           : 31:1
Tx Active Up Time        : 0d 01:16:04
Tx Active Change Count   : 8
Last Tx Active Change    : 07/01/2015 12:09:40
*A:PE-3#
```

There is no active spoke SDP on PE-4.

```
*A:PE-4# show service id 2 endpoint "CORE" | match "Tx Active"
Tx Active                 : none
Tx Active Up Time         : 0d 00:00:00
Tx Active Change Count    : 6
Last Tx Active Change     : 07/01/2015 12:36:41
*A:PE-4#
```

On PE-1 and PE-2, the spoke SDPs are operationally up.

```
*A:PE-1# show service id 1 sdp
=====
Services: Service Destination Points
=====
SdpId          Type Far End addr   Adm   Opr       I.Lbl   E.Lbl
-----
12:1           Mesh 192.0.2.2      Up    Up        262135  262135
13:1           Spok 192.0.2.3      Up    Up        262131  262135
14:1           Spok 192.0.2.4      Up    Up        262130  262133
-----
Number of SDPs : 3
-----
=====
*A:PE-1#
```

However, because pseudowire signaling has been enabled, only one spoke SDP will be active, the others are set in standby.

On PE-1, spoke SDP 13 is active (no pseudowire bit signaled from PE-3).

And the spoke SDP 14 is signaled in standby by PE-4.

```
*A:PE-1# show service id 1 sdp 13:1 detail | match "Peer Pw Bits"
Peer Pw Bits      : None
*A:PE-1# show service id 1 sdp 14:1 detail | match "Peer Pw Bits"
Peer Pw Bits      : pwFwdingStandby
*A:PE-1#
```

On PE-2, both spoke SDPs are signaled in standby.

```
*A:PE-2# configure port 1/1/1 no shutdown
*A:PE-2# show service id 1 sdp 23:1 detail | match "Peer Pw Bits"
Peer Pw Bits      : pwFwdingStandby
*A:PE-2# show service id 1 sdp 24:1 detail | match "Peer Pw Bits"
Peer Pw Bits      : pwFwdingStandby
*A:PE-2#
```

There is one active and three standby spoke SDPs.

Square Resiliency Configuration

PE-6 and PE-7 will be connected to the core VPLS in square resiliency.

- First an endpoint is configured.
- The “no suppress-standby-signaling” is needed to block the standby spoke SDP.
- The multi-chassis endpoint peer is configured. The mc-endpoint ID must match between the two peers.

On PE-7 (similar on PE-6):

```
configure service
  vpls 3
    endpoint "CORE" create
      no suppress-standby-signaling
      mc-endpoint 1
      mc-ep-peer 192.0.2.6
    exit
  exit
exit
```

One spoke SDP is configured on each peer of the multi-chassis to one node of the core VPLS (square resiliency). Each spoke SDP refers to the endpoint CORE.

The precedence will be defined on the spoke SDPs as follows:

- Spoke SDP 72:1 on PE-7 will be active. It is configured as primary (= precedence 0)
- Spoke SDP 61:1 on PE-6 will be the first backup with precedence 1.

On PE-7 (similar on PE-6):

```
configure service
  vpls 3
    spoke-sdp 72:1 endpoint "CORE" create
      precedence primary
    exit
  exit
exit
```


Verification of the spoke SDPs.

On PE-6 and PE-7, the spoke SDPs must be up.

```
*A:PE-7# show service id 3 sdp
=====
Services: Service Destination Points
=====
SdpId          Type Far End addr   Adm   Opr      I.Lbl      E.Lbl
-----
72:1           Spok 192.0.2.2       Up    Up        262133     262132
76:3           Mesh 192.0.2.6       Up    Up        262135     262135
78:3           Mesh 192.0.2.8       Up    Up        262128     262142
-----
Number of SDPs : 3
-----
=====
*A:PE-7#
```

The endpoints on PE-7 and PE-6 can be verified. One spoke SDP is in Tx-Active mode (72 on PE-7 because it is configured as primary).

```
*A:PE-7# show service id 3 endpoint | match "Tx Active"
Tx Active (SDP)           : 72:1
Tx Active Up Time         : 0d 01:17:24
Tx Active Change Count    : 3
Last Tx Active Change     : 07/01/2015 12:40:39
*A:PE-7#
```

There are no active spoke SDP on PE-6.

```
*A:PE-6# show service id 3 endpoint | match "Tx Active"
Tx Active                 : none
Tx Active Up Time         : 0d 00:00:00
Tx Active Change Count    : 6
Last Tx Active Change     : 07/01/2015 12:40:37
*A:PE-6#
```

The output shows that on PE-1, spoke SDP 16 is signaled with peer in standby mode.

```
*A:PE-1# show service id 1 sdp 16:1 detail | match "Peer Pw Bits"
Peer Pw Bits              : pwFwdingStandby
*A:PE-1#
```

Full Mesh VPLS Configuration

On PE-2, the spoke SDP 27 is signaled with peer active (no pseudowire bits).

```
*A:PE-2# show service id 1 sdp 27:1 detail | match "Peer Pw Bits"
Peer Pw Bits          : None
*A:PE-2#
```

There is one active and one standby spoke SDP.

Additional Parameters

Multi-Chassis

```
*A:PE-3# configure redundancy multi-chassis peer 192.0.2.4 mc-endpoint
- mc-endpoint
- no mc-endpoint

[no] bfd-enable      - Configure BFD
[no] boot-timer      - Configure boot timer interval
[no] hold-on-neighb* - Configure hold time applied on neighbor failure
[no] keep-alive-int* - Configure keep alive interval for this MC-Endpoint
[no] passive-mode     - Configure passive-mode
[no] shutdown         - Administratively enable/disable the multi-chassis
                        peer end-point
[no] system-priority - Configure system priority
```

Peer Failure Detection

The default mechanism is based on the keepalive messages exchanged between the peers.

The keep-alive-interval is the interval at which keep-alive messages are sent to the MC peer. It is set in tenths of a second from 5 to 500), with a default value of 5.

Hold-on-neighbor-failure is the number of keep-alive-intervals that the node will wait for a packet from the peer before assuming it has failed. After this interval, the node will revert to single chassis behavior. It can be set from 2 to 25 with a default value of 3.

BFD Session

BFD is another peer failure detection mechanism. It can be used to speed up the convergence in case of peer loss.

```
*A:PE-3# configure redundancy
      multi-chassis
        peer 192.0.2.4
          mc-endpoint
            bfd-enable
          exit
        exit
```

It is using the centralized BFD session. BFD must be enabled on the system interface.

```
*A:PE-3# configure router
      interface "system"
```

Additional Parameters

```
address 192.0.2.3/32
bfd 100 receive 100 multiplier 3
exit
```

Verification of the BFD session:

```
*A:PE-3# show router bfd session
=====
Legend:  wp = Working path   pp = Protecting path
=====
BFD Session
=====
If/Lsp Name/Svc-Id      State      Tx Intvl  Rx Intvl  Multipl
  Rem Addr/Info/SdpId:VcId  Protocols  Tx Pkts   Rx Pkts   Type
    LAG port              LAG ID
-----
system                  Up          100       100       3
  192.0.2.4             mcep       152       84       central
-----
No. of BFD sessions: 1
=====
*A:PE-3#
```

Boot Timer

The **boot-timer** command specifies the time after a reboot that the node will try to establish a connection with the MC peer before assuming a peer failure. In case of failure, the node will revert to single chassis behavior.

System Priority

The system priority influences the selection of the MC master. The lowest priority node will become the master.

In case of equal priorities, the lowest system-id (=chassis MAC address) will become the master.

VPLS Endpoint and Spoke SDP

Ignore Standby Pseudowire Bits

```
*A:PE-1# configure service vpls 1 spoke-sdp 14:1
---snip---
[no] ignore-standby* - Ignore 'standby-bit' received from LDP peer
---snip---
```

The peer pseudowire status bits are ignored and traffic is forwarded over the spoke SDP.

It can speed up convergence for multicast traffic in case of spoke SDP failure.

Traffic sent over the standby spoke SDP will be discarded by the peer.

In this topology, if the **ignore-standby-signaling** command is enabled on PE-1, it sends MC traffic to PE-3 and PE-4 (and to PE-6). If PE-3 fails, PE-4 can start forwarding traffic in the VPLS as soon as it detects PE-3 being down. There is no signaling needed between PE-1 and PE-4.

Block-on-Mesh-Failure

```
*A:PE-3# configure service vpls 2 endpoint "CORE"
---snip---
[no] block-on-mesh-* - Block traffic on mesh-SDP failure
---snip---
```

In case a PE loses all the mesh SDPs of a VPLS, it should block the spoke SDPs to the core VPLS, and inform the MC-EP peer who can activate one of its spoke SDPs.

If block-on-mesh-failure is enabled, the PE will signal all the pseudowires of the endpoint in standby.

In this topology, if PE3 does not have any valid mesh SDP to the VPLS 2 mesh, it will set the spoke SDPs under endpoint CORE in standby.

When block-on-mesh-failure is activated under an endpoint, it is automatically set under the spoke SDPs belonging to this endpoint.

```
*A:PE-3# configure service vpls 2
*A:PE-3>config>service>vpls# info
-----
      stp
      shutdown
    exit
  endpoint "CORE" create
    no suppress-standby-signaling
    mc-endpoint 1
```

Additional Parameters

```
        mc-ep-peer 192.0.2.4
    exit
exit
spoke-sdp 31:1 endpoint "CORE" create
    stp
        shutdown
    exit
    precedence primary
    no shutdown
exit
spoke-sdp 32:1 endpoint "CORE" create
    stp
        shutdown
    exit
    precedence 1
    no shutdown
exit
mesh-sdp 34:2 create
    no shutdown
exit
mesh-sdp 35:2 create
    no shutdown
exit
no shutdown
-----
*A:PE-3>config>service>vpls# endpoint "CORE" block-on-mesh-failure
*A:PE-3>config>service>vpls# info
-----
    stp
        shutdown
    exit
endpoint "CORE" create
    no suppress-standby-signaling
    block-on-mesh-failure
    mc-endpoint 1
        mc-ep-peer 192.0.2.4
    exit
exit
spoke-sdp 31:1 endpoint "CORE" create
    stp
        shutdown
    exit
    block-on-mesh-failure
    precedence primary
    no shutdown
exit
spoke-sdp 32:1 endpoint "CORE" create
    stp
        shutdown
    exit
    block-on-mesh-failure
    precedence 1
    no shutdown
exit
mesh-sdp 34:2 create
    no shutdown
exit
mesh-sdp 35:2 create
    no shutdown
```

```
exit
no shutdown
-----
```

Precedence

```
*A:PE-3# configure service vpls 2 spoke-sdp 31:1
---snip---
[no] precedence      - Configure the spoke-sdp precedence
---snip---
```

The precedence is used to indicate in which order the spoke SDPs should be used. The value is from 0 to 4 (0 being primary), the lowest having higher priority. The default value is 4.

Revert-Time

```
*A:PE-3# configure service vpls 2 endpoint "CORE"
---snip---
[no] revert-time      - Configure the time to wait before reverting to primary spoke-sdp
---snip---
```

If the precedence is equal between the spoke SDPs, there is no revertive behavior. Changing the precedence of a spoke SDP will not trigger a revert. The default is **no revert**.

MAC-Flush Parameters

When a spoke SDP goes from standby to active (due to the active spoke SDP failure), the node will send a **flush-all-but-mine** message.

After a restoration of the spoke SDP, a new **flush-all-but-mine** message will be sent.

A node configured with **propagate MAC flush** will forward the flush messages received on the spoke-SDP to its other mesh/spoke SDPs.

```
*A:PE-1# configure service vpls 1 propagate-mac-flush
```

A node configured with **send flush on failure** will send a **flush-all-from-me** message when one of its SDPs goes down.

```
A:PE-1# configure service vpls 1 send-flush-on-failure
```

Failure Scenarios

For the subsequent failure scenarios, the configuration of the nodes is as described in the [Configuration on page 1165](#).

Core Node Failure

When the Core Node PE-1 fails, the spoke SDPs from PE-3 and PE-4 go down.

Because the spoke SDP 31 between PE-3 and PE-4 was active, the MC master (PE-3 in this case) will select the next best spoke SDP, which will be 32 between PE-3 and PE-2 (precedence 1). See [Figure 177](#).

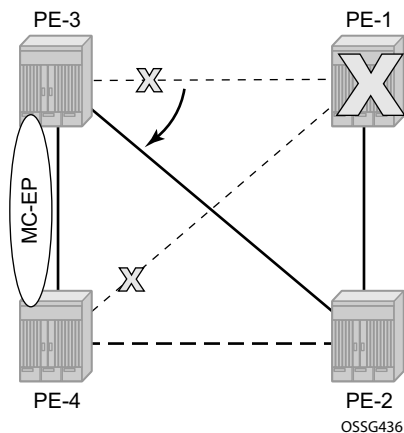


Figure 177: Core Node Failure

```
*A:PE-3# show service id 2 endpoint
=====
Service 2 endpoints
=====
Endpoint name           : CORE
Description              : (Not Specified)
Creation Origin          : manual
Revert time              : 0
Act Hold Delay           : 0
Ignore Standby Signaling : false
Suppress Standby Signaling : false
Block On Mesh Fail       : true
Multi-Chassis Endpoint   : 1
MC Endpoint Peer Addr    : 192.0.2.4
Psv Mode Active          : No
Tx Active (SDP)          : 32:1
Tx Active Up Time        : 0d 00:00:12
Revert Time Count Down   : N/A
```


Multi-Chassis Endpoint for VPLS Active/Standby PW

```
Tx Active Change Count      : 1
Last Tx Active Change       : 07/02/2015 07:17:08
-----
Members
-----
Spoke-sdp: 31:1 Prec:0      Oper Status: Down
Spoke-sdp: 32:1 Prec:1      Oper Status: Up
=====
=====
*A:PE-3#

*A:PE-4# show service id 2 endpoint
=====
Service 2 endpoints
=====
Endpoint name               : CORE
Description                  : (Not Specified)
Creation Origin              : manual
Revert time                  : 0
Act Hold Delay               : 0
Ignore Standby Signaling    : false
Suppress Standby Signaling   : false
Block On Mesh Fail          : false
Multi-Chassis Endpoint      : 1
MC Endpoint Peer Addr       : 192.0.2.3
Psv Mode Active              : No
Tx Active                    : none
Tx Active Up Time            : 0d 00:00:00
Revert Time Count Down      : N/A
Tx Active Change Count      : 3
Last Tx Active Change       : 07/02/2015 07:17:08
-----
Members
-----
Spoke-sdp: 41:1 Prec:2      Oper Status: Down
Spoke-sdp: 42:1 Prec:3      Oper Status: Up
=====
=====
*A:PE-4#
```

Multi-Chassis Node Failure

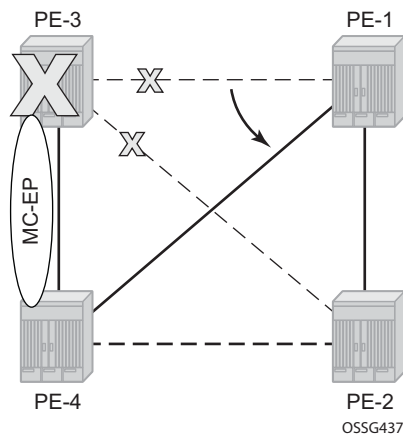


Figure 178: Multi-Chassis Node Failure

When the multi-chassis node PE-3 fails, both spoke SDPs from PE-3 go down.

PE-4 reverts to single chassis mode and selects the best spoke SDP, which will be 41 between PE-4 and PE-1 (precedence 2). See [Figure 178](#).

```
*A:PE-4# show redundancy multi-chassis mc-endpoint peer 192.0.2.3
=====
Multi-Chassis MC-Endpoint
=====
Peer Addr      : 192.0.2.3      Peer Name      :
Admin State    : up            Oper State      : down
Last State chg : 07/02/2015 07:27:22 Source Addr     : 0.0.0.0
System Id      : 02:0c:ff:00:00:00 Sys Priority      : 0
Keep Alive Intvl: 10           Hold on Nbr Fail  : 3
Passive Mode    : disabled      Psv Mode Oper   : No
Boot Timer     : 300            BFD             : enabled
Last update    : 07/02/2015 05:57:45 MC-EP Count      : 1
=====
*A:PE-4#
*A:PE-4# show service id 2 endpoint
=====
Service 2 endpoints
=====
Endpoint name   : CORE
Description     : (Not Specified)
Creation Origin : manual
Revert time     : 0
Act Hold Delay  : 0
Ignore Standby Signaling : false
Suppress Standby Signaling : false
Block On Mesh Fail : false
Multi-Chassis Endpoint : 1
MC Endpoint Peer Addr : 192.0.2.3
Psv Mode Active : No
```

Multi-Chassis Endpoint for VPLS Active/Standby PW

```
Tx Active (SDP)           : 41:1
Tx Active Up Time         : 0d 00:02:40
Revert Time Count Down    : N/A
Tx Active Change Count    : 4
Last Tx Active Change     : 07/02/2015 07:27:22
```

----- Members

```
-----
Spoke-sdp: 41:1 Prec:2           Oper Status: Up
Spoke-sdp: 42:1 Prec:3           Oper Status: Up
```

```
=====
*A:PE-4#
```

Multi-Chassis Communication Failure

If the multi-chassis communication is interrupted, both nodes will revert to single chassis mode.

To simulate a communication failure between the two nodes, define a static route on PE-3 that will black-hole the system address of PE-4.

```
*A:PE-3# configure router static-route 192.0.2.4/32 black-hole
```

Verify that the MC synchronization is down.

```
*A:PE-4# show redundancy multi-chassis mc-endpoint peer 192.0.2.3
=====
Multi-Chassis MC-Endpoint
=====
Peer Addr      : 192.0.2.3          Peer Name      :
Admin State    : up                Oper State     : down
Last State chg : 07/02/2015 08:26:32 Source Addr     : 0.0.0.0
System Id      : 02:0c:ff:00:00:00 Sys Priority     : 0
Keep Alive Intvl: 10              Hold on Nbr Fail : 3
Passive Mode    : disabled         Psv Mode Oper  : No
Boot Timer     : 300              BFD            : enabled
Last update    : 07/02/2015 05:57:45 MC-EP Count     : 1
=====
*A:PE-4#
```

The spoke SDPs are active on PE-3 and on PE-4.

```
*A:PE-3# show service id 2 endpoint | match "Tx Active"
Tx Active (SDP)      : 31:1
Tx Active Up Time    : 0d 00:05:58
Tx Active Change Count : 5
Last Tx Active Change : 07/02/2015 08:26:20

*A:PE-4# show service id 2 endpoint | match "Tx Active"
Tx Active (SDP)      : 41:1
Tx Active Up Time    : 0d 00:04:56
Tx Active Change Count : 6
Last Tx Active Change : 07/02/2015 08:26:32
```

This can potentially cause a loop in the system. The [Passive Mode on page 1185](#) explains how to avoid this loop.

Passive Mode

As in [Multi-Chassis Communication Failure on page 1184](#), if there is a failure in the multi-chassis communication, both nodes will assume that the peer is down and will revert to single-chassis mode. This can create loops because two spoke SDPs can become active.

One solution is to synchronize the two core nodes, and configure them in passive mode. See [Figure 179](#).

In passive mode, both peers will stay dormant as long as one active spoke SDP is signaled from the remote end. If more than one spoke SDP becomes active, the MC-EP algorithm will select the best SDP. All other spoke SDPs are blocked locally (in Rx and Tx directions). There is no signaling sent to the remote PEs.

If one peer is configured in passive mode, the other peer will be forced to passive mode as well.

The **no suppress-standby-signaling** and **no ignore-standby-signaling** commands are required.

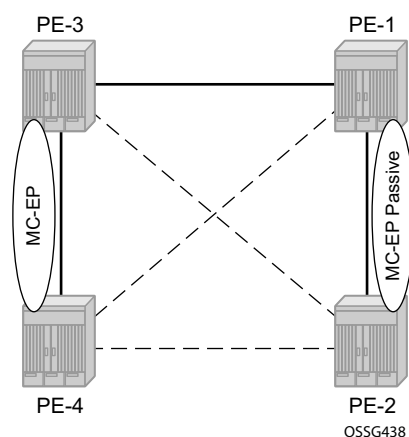


Figure 179: Multi-Chassis Passive Mode

The following output shows the multi-chassis configuration on PE-1 (similar on PE-2).

```
*A:PE-1# configure redundancy
      multi-chassis
        peer 192.0.2.2 create
          mc-endpoint
            no shutdown
            passive-mode
          exit
        no shutdown
      exit
exit
```

The following output shows the VPLS spoke SDPs configuration on PE-1 (similar on PE-2)

```
*A:PE-1# configure service
      vpls 1
        endpoint "METRO1" create
          no suppress-standby-signaling
          mc-endpoint 1
            mc-ep-peer 192.0.2.2
          exit
        exit
      spoke-sdp 13:1 endpoint "METRO1" create
        no shutdown
      exit
      spoke-sdp 14:1 endpoint "METRO1" create
        no shutdown
      exit
    no shutdown
exit
```

To simulate a communication failure between the two nodes, a static route is defined on PE-3 that will black-hole the system address of PE-4.

```
*A:PE-3# configure router static-route 192.0.2.4/32 black-hole
```

The spoke SDPs are active on PE-3 and on PE-4.

```
*A:PE-3# show service id 2 endpoint | match "Tx Active"
Tx Active (SDP)           : 31:1
Tx Active Up Time         : 0d 00:26:45
Tx Active Change Count    : 5
Last Tx Active Change     : 07/02/2015 08:26:20
```

```
*A:PE-4# show service id 2 endpoint | match "Tx Active"
Tx Active (SDP)           : 41:1
Tx Active Up Time         : 0d 00:27:05
Tx Active Change Count    : 6
Last Tx Active Change     : 07/02/2015 08:26:32
```

PE-1 and PE-2 have blocked one spoke SDP which avoids a loop in the VPLS.

```
*A:PE-1# show service id 1 endpoint "METRO1" | match "Tx Active"
Tx Active (SDP)           : 13:1
Tx Active Up Time         : 0d 00:29:46
Tx Active Change Count    : 12
Last Tx Active Change     : 07/02/2015 08:26:32
```

```
*A:PE-2# show service id 1 endpoint "METRO1" | match "Tx Active"
Tx Active                 : none
Tx Active Up Time         : 0d 00:00:00
Tx Active Change Count    : 6
Last Tx Active Change     : 07/02/2015 07:27:12
```

The passive nodes do not set the pseudowire status bits; hence, the nodes PE-3 and PE-4 are not aware that one spoke SDP is blocked.

Conclusion

Multi-chassis endpoint for VPLS active/standby pseudowire allows the building of hierarchical VPLS without single point of failure, and without requiring STP to avoid loops.

Care must be taken to avoid loops. The multi-chassis peer communication is important and should be possible on different interfaces.

Passive mode can be a solution to avoid loops in case of multi-chassis communication failure.

Multi-Segment Pseudowire Routing

In This Chapter

This chapter describes advanced multi-segment pseudowire routing configurations.

Topics in this section include:

- [Applicability on page 1190](#)
- [Summary on page 1191](#)
- [Overview on page 1192](#)
- [Configuration on page 1195](#)
- [Conclusion on page 1242](#)

Applicability

This chapter is applicable to all of the 7x50 series and was tested on release 13.0.R3. There are no specific pre-requisites for this configuration.

Summary

Starting with SR OS 9.0 R3, the SR/ESS portfolio supports the use of Multi-Segment Pseudowire (MS-PW) routing for Epipe services. MS-PW routing is described in draft-ietf-pwe3-dynamic-ms-pw, also known as Dynamic Placement of MS-PW and it is an extension of the procedures proposed in RFC 6073 (static MS-PW) to enable multi-segment pseudowires to be dynamically placed. Ultimately MS-PW Routing provides the capability of setting up MS-PWs without provisioning the S-PE (Switching PEs).

This configuration note will go through the configuration process required to setup MS-PW routing and will provide two configuration examples typically deployed in Service Providers: MS-PW within the same Autonomous System (AS) and MS-PW across two different AS. Different configuration options are tested and described in each example.

Overview

From a data plane perspective, MS-PW Routing does not introduce any changes with respect to the existing MS-PW architecture. However from the control plane perspective, MS-PW Routing brings a new information model and set of procedures to set up a MS-PW. These are the building blocks defined by the MS-PW Routing feature:

- A new information model is introduced for dynamic MS-PW based on the FEC129, Attachment Individual Identifier (AII) Type 2. Note that static MS-PW uses FEC128 while VPLS with BGP-AD uses FEC129, but with AII Type 1 instead.
- FEC129 is suitable for applications where the local PE with a SAII (Source Attachment Individual Identifier) must automatically learn the remote TAI (Target Attachment Individual Identifier), normally through BGP, before launching the LDP mapping message for the pseudowire setup. The following figure shows the FEC129 structure:

G.Pwid (0x81)	C	Pw Type	Pw Info Length
AGI Type	Length	Value	
AGI Value (Cont.)			
All Type	Length	Value	
SAII Value (Cont.)			
All Type	Length	Value	
TAII Value (Cont.)			

ACG0004A

Figure 180: FEC129 Structure

- The Attachment Group Identifier (AGI) is not used in dynamic MS-PW signaling. In VPLS, it typically carries the instance identifier. It is zero in dynamic MS-PWs.
- The SAII and TAI (or pseudowire end-point identifiers) are encoded in FEC129 and can have two different formats: AII Type 1 or AII Type 2.
- AII Type 1 is composed of a fixed 32-bit value unique on the local PE. This AII Type is used by VPLS when BGP-AD is needed.
- AII Type 2 is composed of `GID:prefix:AC-ID` (Global-ID:prefix:Attachment-Circuit-ID) and allows for summarization; by this enhancing scalability in large networks. The GID is normally derived from the AS number, the prefix from the node system address and the AC-ID is the local pseudowire end-point identifier. The combination of the three identifiers gives us a globally unique 96-bit AII value. In general, the same global ID and prefix are assigned for all ACs belonging to the same Terminating PE (T-PE). This is not a strict requirement though.

All Type=2	Length	Global ID
Global ID (Cont.)	Prefix	
Prefix (Cont.)	AC ID	
AC ID (Cont.)		

ACG0004B

Figure 181: All Type 2 Format

- A MS-PW routing table must be built in all the T-PEs and S-PEs through one of the following two mechanisms:
 - Multi-Protocol BGP (MP-BGP), using a new NLRI and a new SAFI (pseudowire routing SAFI=6, with AFI=25 L2VPN). The FEC129 All Type 2 global values are mapped in the pseudowire routing NLRI and advertised by BGP. In the tested release, SR OS only supports an NLRI comprising a Length, RD, Global ID and 32-bit Prefix, that is, the AC ID is not included in the advertised NLRI. The AC ID is not included as indicated in the draft-ietf-pwe3-dynamic-ms-pw since “the source T-PE knows by provisioning the AC ID on the terminating T-PE to use in signaling. Hence, there is no need to advertise a “fully qualified” 96 bit address on a per pseudowire Attachment Circuit basis. Only the T-PE Global ID, Prefix, and prefix length needs to be advertised as part of well known BGP procedures”. This also minimizes the amount of routing information that is advertised in BGP to only what is necessary to reach the far-end T-PE.

Length	
Route Distinguisher (8 bytes)	
	Global ID
Global ID	Prefix
Prefix	AC ID
AC ID	

ACG0004C

Figure 182: Pseudowire Routing NLRI (the AC ID is always zero)

- Static routes, configurable via CLI

- Once the MS-PW routing table is populated, Targeted LDP (TLDP) will make use of it to signal the MS-PW all the way from the originating T-PE to the terminating T-PE as well as in the reverse direction. The following methods will be used:
 - At the originating T-PE¹, a longest-match lookup will be performed in the pseudowire routing table for the configured TAIL. Based on the lookup outcome, a label mapping message will be sent to the Next Signaling Hop (NSH).
 - At the intermediate S-PEs and terminating T-PE, a longest-match lookup between the TAIL Type 2 included in the TLDP signaling message and entries installed in the pseudowire routing table will be performed.
 - Alternatively to the pseudowire routing table lookup, TLDP can also use explicit routing, as per section 7.4.2 of draft-ietf-pwe3-dyn-ms-pw. If that is the case, a “path” must be configured at the T-PEs. The originating T-PE will include an ERO (Explicit Route Object) in the TLDP label mapping, containing all the S-PE hops specified in the configured path. Each S-PE along the path will remove its own entry from the ERO and will forward the label mapping message to the next hop.

The SR OS, starting from R9.0R3, supports the information model and all the methods described above:

- Dynamic placement through MP-BGP, with the pseudowire routing NLRI
- Static routes
- Explicit paths

In addition to the above, the following features are supported on dynamic MS-PW:

- Auto-configuration of spoke SDPs at T-PE (if enabled on a T-PE, there is no need for configuring the TAIL of the remote T-PE. Refer to [Active/Passive Signaling and Auto-Configuration on page 1208](#). The auto-configuration is typically used in hub-and-spoke scenarios. The TAIL would only be configured on the spoke T-PE while the TAIL would be automatically provisioned on the hub T-PE if the auto-config parameter is added.
- OAM using Virtual Circuit Connectivity Verification vccv-ping and vccv-trace
- Pseudowire redundancy
- Control word
- Hash label (if the chassis dependency requirements are met)
- Standby-signaling-master and standby-signaling-slave commands
- Filters

1. The “originating T-PE” will be the T-PE initiating the MS-PW signaling. Refer to the [Active/Passive Signaling and Auto-Configuration](#) section for further information.

Configuration

The following flow-chart shows the configuration process to be followed when setting up MS-PW routing. Base IGP and MPLS configuration is assumed to be in place before these configuration tasks can be carried out.

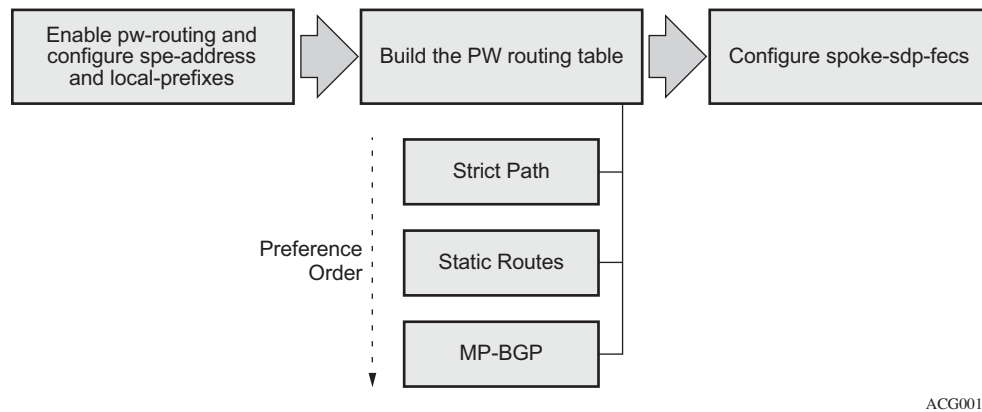


Figure 183: Configuration Flow Chart

The following subsections review these three steps, including all the options in detail.

- [Pseudowire Routing Enablement on page 1196](#)
- [Building the Pseudowire Routing Table on page 1199](#)
- [Spoke-SDP-FEC Timers on page 1211](#)

Pseudowire Routing Enablement

The first step in the configuration is to enable pw-routing and configure the required pw-routing basic parameters: the spe-address (in S-PEs and T-PEs) and the local-prefix/prefixes (only required in T-PEs). A new pw-routing context has been added from 9.0 R3 onward. The following CLI examples show the configuration of the spe-address and local-prefixes.

```
*A:PE-1# configure service
    pw-routing
        spe-address 65536:192.0.2.1
        local-prefix 65536:192.0.2.11 create
            advertise-bgp route-distinguisher 65536:11 community 65535:11
        exit
        local-prefix 65536:192.0.2.12 create
            advertise-bgp route-distinguisher 65536:12 community 65535:12
        exit
    exit

*A:PE-3# configure service
    pw-routing
        spe-address 65536:192.0.2.3
        local-prefix 65536:192.0.2.3 create
            advertise-bgp route-distinguisher 65536:3
        exit
    exit
```

In order to enable support for MS-PW routing on a 7x50 node, a single, globally unique, S-PE ID (known as the spe-address) is first configured under **config>service>pw-routing** on each 7x50 to be used as a T-PE or S-PE. The S-PE address has the format global-id:prefix. It is not possible to configure any local prefixes used for pseudowire routing or to configure spoke SDPs using dynamic MS-PWs at a T-PE unless an S-PE address has already been configured. The S-PE address is used as the address of a node when populating the switching point TLV in the LDP label mapping message and the pseudowire status notification sent for faults at an S-PE. The following CLI output shows the spe-address configuration format:

```
A:PE-1# configure service pw-routing spe-address
- no spe-address
- spe-address <global-id:prefix>

<global-id:prefix>   : <global-id>:{<prefix>|<ipaddress>}
global-id - [1..4294967295]
prefix    - [1..4294967295]
ipaddress - a.b.c.d
```

Where:

- <global-id> is normally the 2 or 4-byte ASN identifying the network (although nothing prevents the operator from configuring any value here)

- <prefix> is normally the node's system address (although any value in ip address or decimal format can be used)

If an S-PE is capable of Dynamic MS-PW signaling, but is not assigned with an S-PE address, then on receiving a Dynamic MS-PW label mapping message the S-PE will return a Label Release with the "LDP_RESOURCES_UNAVAILABLE" (0x38) status code. Note that the S-PE address cannot be changed unless the dynamic MS-PW configuration is completely removed; therefore it is recommended to configure the spe-address carefully and keep it for the life of the services.

The second basic pw-routing context parameter is the local-prefix:

```
A:PE-1# configure service pw-routing local-prefix
- local-prefix <local-prefix> [create]
- no local-prefix <local-prefix>

<local-prefix>          : <global-id>:<ip-addr>|<raw-prefix>
                        ip-addr      - a.b.c.d
                        raw-prefix    - [1..4294967295]
                        global-id     - [1..4294967295]

[no] advertise-bgp      - Configure BGP advertisement
```

One or more local (Layer 2) prefixes (up to a maximum of 16), which are formatted in the style of <global-id>:<ipv4-address>, are supported. A local prefix identifies a T-PE in the pseudowire routing domain. When using explicit paths or static-routes, the definition of the local-prefix (or local-prefixes) without any further attribute is enough. However, when BGP is used, the advertise-bgp parameter along with a route-distinguisher (RD) value and an optional BGP community is required.

```
*A:PE-1# configure service pw-routing local-prefix 65536:192.0.2.11 advertise-bgp
- advertise-bgp route-distinguisher <rd> [community <community>]
- no advertise-bgp route-distinguisher <rd>

<rd>                  : <ip-addr:comm-val>|<2byte-asnumber:ext-comm-val>|<4byte-asnum-
ber:comm-val>
                        ip-addr      - a.b.c.d
                        comm-val      - [0..65535]
                        2byte-asnumber - [1..65535]
                        ext-comm-val   - [0..4294967295]
                        4byte-asnumber - [1..4294967295]

<community>          : <asnumber:comm-val>
                        asnumber      - [1..65535]
                        comm-val      - [0..65535]
```

Up to four unique RDs (and communities) can be configured per each local-prefix. Different RDs for the same prefix allow the operator to advertise the same prefix coming from up to four different next-signaling hops (NSH). Route-Reflectors would reflect the four routes in that case, whereas only one would be reflected should the same RD be used.

Pseudowire Routing Enablement

```
*A:PE-1>config>service>pw-routing>local-prefix# info
-----
      advertise-bgp route-distinguisher 400:20
      advertise-bgp route-distinguisher 500:3
      advertise-bgp route-distinguisher 600:300
      advertise-bgp route-distinguisher 65536:11 community 65535:11

*A:PE-1>config>service>pw-routing>local-prefix# advertise-bgp route-distinguisher 700:100
MINOR: SVCMGR #6072 Maximum number of RD's has been reached
```

For each local prefix, BGP then advertises each global ID/prefix tuple and unique RD and community (if configured) using the MS-PW NLRI, based on the aggregated FEC129 AII Type 2 and the Layer 2 VPN/PW routing AFI/SAFI 25/6, to each BGP neighbor, subject to local BGP policies.

Building the Pseudowire Routing Table

Once the spe-address and the local-prefix(es) have been configured and before configuring the Epipe service itself on the T-PE nodes, we need to populate the pseudowire routing table in all the participating T-PE and S-PE nodes, so that TLDP knows what the Next Signaling Hop (NSH) is and sends LDP Label Mapping messages.

The pseudowire routing table will be populated with local prefixes, static-routes and BGP routes, where the static-routes have preference over the BGP-learned routes. The pseudowire routing table can be overridden by the explicit paths, should the operator want to configure them. Therefore, when TLDP signals an LDP Label Mapping for a given TAIL, it will:

- First check if there is an explicit path configured for that spoke-sdp-fec.
- Otherwise it will look up the TAIL prefix into the pseudowire routing table, where static routes take precedence over BGP routes.

An aggregation scheme, similar to that used for classless IPv4 addresses, can be employed in the pseudowire routing table, where a longest match is used to find a route. Except for the default pseudowire route, which is encoded with a 0 mask, masks included in the pw-routing table are:

- /64 for regular prefixes, including a global-id and prefix (as previously mentioned, note that the AC-ID is not included in the BGP NLRI).
- /96 for local prefixes, including the AC-ID, as well as global-id and prefix.

Each S-PE and T-PE must have a pseudowire routing table that contains a reference to the TLDP session to use to signal to a set of next hop S-PEs to reach a given T-PE (or the T-PE if that is the next hop). For VLLs, this table contains aggregated AII Type 2 FECs and may be populated with routes that are learned through MP-BGP or that are statically configured.

Explicit Paths

A set of default explicit routes to a remote T-PE prefix may be configured on a T-PE under **config>services>pw-routing** using the path name command. Explicit paths are used to populate the explicit route TLV used by MS-PW TLDP signaling. Only strict (fully qualified) explicit paths are supported. Note that it is possible to configure explicit paths independently of the configuration of BGP or static routing.

The following CLI excerpt shows an explicit path example for a MS-PW following the PE-1-PE-3-PE-5-PE-2 path (see the diagram in [Figure 184](#)). The IP addresses are the system addresses of all the S-PE and T-PE along the path (except for PE-1).

```
*A:PE-1# configure service pw-routing
      path "path-1" create
        hop 1 192.0.2.3
        hop 2 192.0.2.5
        hop 3 192.0.2.2
        no shutdown
      exit
```

Static Routes

In addition to support for BGP routing, static MS-PW routes may also be configured using the **config>services>pw-routing>static-route** command. Each static route comprises of the target T-PE Global-ID and prefix, and the IP address of the TLDP session to the next hop S-PE or T-PE that should be used:

```
A:PE-1# configure service pw-routing static-route
- no static-route <route-name>
- static-route <route-name>

<route-name>      : <global-id>:<prefix>:<next-hop-ip_addr>
global-id         - 0..4294967295
prefix            - a.b.c.d|0..4294967295
ip_addr           - a.b.c.d
```

If a static route **<global-id>:<prefix>** is set to 0, then this represents the default route.

```
*A:PE-1# configure service pw-routing
---snipped---
static-route 0:0.0.0.0:192.0.2.3
static-route 0:0.0.0.0:192.0.2.4
```

Note that, even though you can configure several default-routes, only one default route is added to the pseudowire routing table. The following command shows the pseudowire routing table content

where only one default route (out of the two previously configured ones) is added. The default route added to the pseudowire routing table is the first valid route added to the configuration.

```
A:PE-1# show service pw-routing route-table all-routes
=====
Service PW L2 Routing Information
=====
```

AII-Type2/Prefix-Len Route-Distinguisher	Next-Hop Community	Owner Best	Age
0:0.0.0.0:0/0	192.0.2.3	static	19h11m57s
0:0	0:0	yes	
...			

If a static route exists to a given T-PE, then this is used in preference to any BGP route that may exist.

BGP Routes

As already mentioned, the dynamic advertisement of the pseudowire routes is enabled for each prefix and RD using the **advertise-bgp** command in the **config>services>pw-routing>local-prefix** context. Note that a BGP export policy is also required in order to export MS-PW routes in MP-BGP. This can be done using a default policy matching all the MS-PW routes, such as the following:

```
*A:PE-1# configure router
      policy-options
      begin
      policy-statement "export_ms-pw"
      entry 10
      from
      family ms-pw
      exit
      action accept
      exit
      exit
      exit
      exit
      commit
exit

*A:PE-1# configure router
      bgp
      enable-peer-tracking
      rapid-withdrawal
      group "region"
      family ms-pw
      type internal
      export "export_ms-pw"
      neighbor 192.0.2.3
      exit
      neighbor 192.0.2.4
      exit
      exit
      no shutdown
exit
```

MS-PW routes advertised/received can be debugged and shown on the log sessions (**debug router bgp update**). Note that a new address family and NLRI are used to distribute the MS-PW prefixes:

```
28 2015/06/03 07:42:59.69 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 51
  Flag: 0x90 Type: 14 Len: 26 Multiprotocol Reachable NLRI:
    Address Family MSPW
    NextHop len 4 NextHop 192.0.2.1
    [MSPW] rd: 65536:12, global-id 65536, prefix 192.0.2.12,  ac-id 0, preflen 128
  Flag: 0x40 Type: 1 Len: 1 Origin: 2
  Flag: 0x40 Type: 2 Len: 0 AS Path:
```

```

Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
Flag: 0xc0 Type: 8 Len: 4 Community:
65535:12

```

MS-PW BGP routes can also be displayed in the pseudowire routing table along with the static-routes and the local-prefixes.

```

*A:PE-1# show service pw-routing route-table
=====
Service PW L2 Routing Information
=====
AII-Type2/Prefix-Len      Next-Hop      Owner  Age
Route-Distinguisher      Community      Best
-----
65536:192.0.2.11:0/64    192.0.2.1     local  17h19m42s
0:0                      0:0           yes
65536:192.0.2.11:0/64    192.0.2.1     local  17h19m42s
65536:11                 65535:11      yes
65536:192.0.2.12:0/64    192.0.2.1     local  17h19m42s
0:0                      0:0           yes
65536:192.0.2.12:0/64    192.0.2.1     local  17h19m42s
65536:12                 65535:12      yes
65536:192.0.2.13:0/64    192.0.2.1     local  00h49m29s
0:0                      0:0           yes
65536:192.0.2.14:0/64    192.0.2.1     local  00h49m29s
0:0                      0:0           yes
65536:192.0.2.21:0/64    192.0.2.3     bgp     00h05m46s
65536:21                 65535:11      yes
65536:192.0.2.22:0/64    192.0.2.4     bgp     00h05m42s
65536:22                 65535:12      yes
65536:192.0.2.23:0/64    192.0.2.3     static  00h49m29s
0:0                      0:0           yes
65536:192.0.2.24:0/64    192.0.2.4     static  00h49m29s
0:0                      0:0           yes
-----
Entries found: 10
=====

```

It is important to note that if there are two (or more) equal cost BGP MS-PW routes with identical <global-ID:prefix> and different RDs in the RIB they are both tagged as best/used and both will be added to the pseudowire routing table, however only the one with a higher RD will be shown as “Best” and as a result of that, only that one will be used by TLDP for the NSH.

The pw-routing context at PE-2 contains the following advertise-bgp entries for local-prefix 65536:192.0.2.2:

```

*A:PE-2# configure service pw-routing
      local-prefix 65536:192.0.2.2 create
        advertise-bgp route-distinguisher 65536:21
        advertise-bgp route-distinguisher 65536:22
      exit

```

BGP Routes

The following CLI output shows an example of two equal cost MS-PW routes. The route 65536:192.0.2.2 with RD 65536:21 and RD 65536:22 are tagged as best/used (u*>):

```
*A:PE-1# show router bgp routes ms-pw aii-type2 65536:192.0.2.2:0
=====
BGP Router ID:192.0.2.1      AS:65536      Local AS:65536
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP MSPW Routes
=====
Flag  Network      RD
     Nexthop      AII-Type2/Preflen
     As-Path
-----
u*>?  65536:192.0.2.2      65536:21
      192.0.2.3          65536:192.0.2.2:0/64
      No As-Path
*?    65536:192.0.2.2      65536:21
      192.0.2.4          65536:192.0.2.2:0/64
      No As-Path
u*>?  65536:192.0.2.2      65536:22
      192.0.2.4          65536:192.0.2.2:0/64
      No As-Path
*?    65536:192.0.2.2      65536:22
      192.0.2.3          65536:192.0.2.2:0/64
      No As-Path
-----
Routes : 4
=====
```

However, only the one with RD 65536:22 (higher RD) is added as “Best” to the pseudowire routing table and TLDP will use 192.0.2.4 as the NSH:

```
*A:PE-1# show service pw-routing route-table all-routes
=====
Service PW L2 Routing Information
=====
AII-Type2/Prefix-Len      Next-Hop      Owner  Age
Route-Distinguisher      Community     Best
-----
65536:192.0.2.2:0/64      192.0.2.3     bgp    00h12m22s
65536:21                  65535:11      no
65536:192.0.2.2:0/64      192.0.2.4     bgp    00h12m17s
65536:22                  65535:12      yes
---snipped---
```

How does the 7x50 SR/ESS TLDP process select the NSH (Next-Signaling Hop) for two identical <global-ID;prefix/RD> tuples?

In case the originating T-PE or any intermediate S-PE receives two (or more) equal cost MS-PW routes with the same RD but from different Next-Hops, all the MS-PW routes will be added to the MS-PW routing table. The following output shows two MS-PW routes with the same <global-ID:prefix/RD> but different NH. Both are added to the MS-PW routing table as “Best”.

```
*A:PE-1# show service pw-routing route-table all-routes
=====
Service PW L2 Routing Information
=====
AII-Type2/Prefix-Len      Next-Hop      Owner  Age
Route-Distinguisher      Community     Best
-----
---snipped---
65536:192.0.2.2:0/64      192.0.2.3     bgp    01d15h21m
65536:21                  65535:21      yes
65536:192.0.2.2:0/64      192.0.2.4     bgp    01d15h21m
65536:21                  65535:21      yes
---snipped---
```

If that is the case, TLDP will pick up the NSH out of an ECMP hashing algorithm applied to the <global-ID:prefix:AC-ID> for the SAI and the TAI of the pseudowires pointing at the same prefix. The output of that hashing algorithm will determine what the NSH will be for a given spoke-sdp-fec.

When path diversity for an active and a standby pseudowire (hot standby pseudowire redundancy) is desired and the two pseudowires of the same Epipe end-point are pointing at the same remote <global-ID:prefix> coming from two different NHs, the operator has to make sure TLDP chooses a different NSH for the standby pseudowire. Only in that case hot standby pseudowire redundancy can be achieved. As a rule of thumb, if the SAI/TAI of the active and standby pseudowires are separated by 16 or more AC-ID values, TLDP will select a different NSH for both pseudowires.

For example:

- Given the following SAI/TAI AC-ID values for the active/standby pseudowires on the originating T-PE, TLDP will select the same NSH:
 - Active pseudowire: saii-type2 — 65536:192.0.2.1:1, taii-type2 — 65536:192.0.2.2:1
 - Standby pseudowire: saii-type2 — 65536:192.0.2.1:2, taii-type2 — 65536:192.0.2.2:2
- However, the following SAI/TAI AC-ID values for the active/standby pseudowires on the originating T-PE will allow the ECMP hashing algorithm to make TLDP select different NSHs for the active and the standby pseudowires:
 - Active pseudowire: saii-type2 — 65536:192.0.2.1:1, taii-type2 — 65536:192.0.2.2:1
 - Standby pseudowire: saii-type2 — 65536:192.0.2.1:16, taii-type2 — 65536:192.0.2.2:16

Other AC-ID values greater than 16 (for the standby pseudowire) would also have achieved next hop diversity.

Configuring Dynamic Pseudowires on the T-PEs

Before any LDP signaling can take place, note that T-LDP sessions must be explicitly configured on T-PEs and S-PEs.

One or more spoke-SDPs may be configured for distributed Epipe VLL services. Dynamic MS-PWs use FEC129 (also known as the Generalized ID FEC) with Attachment Individual Identifier (AII) Type 2 to identify the pseudowire, as opposed to FEC128 (also known as the PW ID FEC) used for traditional single segment pseudowires and for pseudowire switching. FEC129 spoke-SDPs are configured under the spoke-sdp-fec command in the CLI. Note that spoke-sdp-fecs (or FEC129 spoke-SDPs) are by default fec-type 129 and aii-type 2 (those are the only values supported for spoke-sdp-fecs in the release 10.0R4). Spoke-sdp-fecs can be part of an endpoint and even an ICB (Inter-Chassis Backup) pseudowire.

```
*A:PE-1# configure service epipe 2 spoke-sdp-fec
- no spoke-sdp-fec <spoke-sdp-fec-id>
- spoke-sdp-fec <spoke-sdp-fec-id> [fec <fec-type>] [aii-type <aii-type>] [create]
- spoke-sdp-fec <spoke-sdp-fec-id> no-endpoint
- spoke-sdp-fec <spoke-sdp-fec-id> [fec <fec-type>] [aii-type <aii-type>] [create] end-
point <name> [icb]

<spoke-sdp-fec-id>      : [1..4294967295]
<fec-type>              : [129..130]
<aii-type>              : [1..2]
<name>                  : [32 chars max]
<icb>                   : keyword - configure spoke-sdp as inter-chassis backup
```

FEC129 AII Type 2 uses a SAII and a TAII to identify the ends of a pseudowire at the T-PE. The SAII identifies the local end, while the TAII identifies the remote end. The SAII and TAII are each structured as follows:

- Global-ID: this is a 4 byte identifier that uniquely identifies an operator or the local network. Normally this matches the ASN
- Prefix: a 4-byte prefix, which should correspond to one of the local prefixes assigned under pw-routing
- AC-ID: a 4-byte identifier for this end of the pseudowire. This should be locally unique within the scope of the global-id:prefix

In terms of the SDP tunnel being used by each spoke-sdp-fec, pw-routing chooses the MS-PW path in terms of the sequence of S-PEs to use to reach a given T-PE. It does not select the SDP to use on each hop, which is instead determined at signaling time. When a label mapping is sent for a given pseudowire segment, an LDP SDP will be used to reach the next-hop S-PE/T-PE if such an SDP exists. If not, and an RFC 3107 labeled BGP SDP is available, then that will be used². Otherwise, the label mapping will fail and a label release will be sent.

The following CLI output shows one example of two spoke-sdp-fecs belonging to an endpoint:

```
*A:PE-1# configure service
      pw-template 1 create
        controlword
      exit
    epipe 2 customer 1 create
      description "ms-pw epipe with bgp - using 2 prefixes"
      endpoint "CORE" create
        description "end-point for epipe A/S PW redundancy"
        revert-time 10
        standby-signaling-master
      exit
    sap 1/1/4:2 create
    exit
    spoke-sdp-fec 21 fec 129 aii-type 2 create endpoint CORE
      precedence primary
      pw-template-bind 1
      saii-type2 65536:192.0.2.11:1
      taii-type2 65536:192.0.2.21:1
      no shutdown
    exit
    spoke-sdp-fec 22 fec 129 aii-type 2 create endpoint CORE
      pw-template-bind 1
      saii-type2 65536:192.0.2.12:1
      taii-type2 65536:192.0.2.22:1
      no shutdown
    exit
  no shutdown
exit
```

These are all of the options available under the spoke-sdp-fec context:

```
*A:PE-1# configure service epipe 1 spoke-sdp-fec
- no spoke-sdp-fec <spoke-sdp-fec-id>
- spoke-sdp-fec <spoke-sdp-fec-id> [fec <fec-type>] [aii-type <aii-type>] [create]
- spoke-sdp-fec <spoke-sdp-fec-id> no-endpoint
- spoke-sdp-fec <spoke-sdp-fec-id> [fec <fec-type>] [aii-type <aii-type>] [create] end-
point <name> [icb]

<spoke-sdp-fec-id>      : [1..4294967295]
<fec-type>              : [129..130]
```

2. Note that RSVP SDPs might be picked at the T-PE through the use of pw-template <policy-id> [use-provisioned-sdp], however there is no way to select an RSVP SDP on an S-PE.

```
<aii-type>          : [1..2]
<name>              : [32 chars max]
<icb>               : keyword - configure spoke-sdp as inter-chassis backup

[no] auto-config     - Configure auto-configuration
[no] path            - Configure path-name
[no] precedence      - Configure precedence
[no] pw-template-bi* - Configure Pseudo-Wire template-binding policy
[no] retry-count     - Configure retry count
[no] retry-timer     - Configure retry timer
[no] saii-type2      - Configure Source Attachment Individual Identifier (SAII)
[no] shutdown        - Administratively enable/disable the spoke SDP FEC binding
[no] signaling       - Configure Spoke-SDP FEC signaling
[no] standby-signal* - Enable PW standby-signaling slave
[no] taii-type2      - Configure Target Attachment Individual Identifier (TAII)
```

Active/Passive Signaling and Auto-Configuration

When an MS-PW is signaled, each T-PE might independently initiate signaling of the MS-PW. This could result in a different path being used in each direction of the pseudowire. To avoid this situation one of the T-PEs will start the pseudowire signaling (active role), while the other T-PE waits to receive the LDP label mapping message before sending the LDP label mapping message for the reverse direction of the pseudowire (passive role).

Enable debugging for LDP messages on PE-2:

```
*A:PE-2# debug router ldp peer 192.0.2.5 packet init detail
*A:PE-2# debug router ldp peer 192.0.2.5 packet label detail
*A:PE-2# debug router ldp peer 192.0.2.6 packet init detail
*A:PE-2# debug router ldp peer 192.0.2.6 packet label detail
```

By default, the T-PE with SAII>TAII will have the active role and will send the label mapping first. When spoke-sdp-fec 21 is first disabled, and then enabled, PE-2 sends a label mapping to PE-5 first (message 77 in output below). Afterwards, it receives a label mapping packet from PE-5 (message 78).

```
*A:PE-2# configure service epipe 2 spoke-sdp-fec 21 shutdown
*A:PE-2# configure service epipe 2 spoke-sdp-fec 21 no shutdown
```

```
*A:PE-2# show log log-id 3
=====
Event Log 3
=====
78 2015/06/03 12:07:01.09 UTC MINOR: DEBUG #2001 Base LDP
"LDP: LDP
Recv Label Mapping packet (msgId 11220) from 192.0.2.5:0
Protocol version = 1
Label 262131 advertised for the following FECs
Service FEC GENPWE3: ENET(5)
```

```

AGI = type: 1, len: 8, val: 00:00
SAII = T: 2, L: 12, Global-id: 65536, Prefix: 192.0.2.11, AcId: 1
TAII = T: 2, L: 12, Global-id: 65536, Prefix: 192.0.2.21, AcId: 1
Group ID = 0 cBit = 1
Interface parameter Mtu = 1500
Interface parameter VCCV = 0x106
PW status bits = 0x18
Switching hop: System = 192.0.2.3, Remote System = 192.0.2.1
previous segment fec AGI = type: 1, len: 8, val: 00:00
SAII = T: 2, L: 12, Global-id: 65536, Prefix: 192.0.2.11, AcId: 1
TAII = T: 2, L: 12, Global-id: 65536, Prefix: 192.0.2.21, AcId: 1
S-PE = T: 2, L: 12, Global-id: 65536, Prefix: 192.0.2.3, AcId: 0
Switching hop: System = 192.0.2.5, Remote System = 192.0.2.3
previous segment fec AGI = type: 1, len: 8, val: 00:00
SAII = T: 2, L: 12, Global-id: 65536, Prefix: 192.0.2.11, AcId: 1
TAII = T: 2, L: 12, Global-id: 65536, Prefix: 192.0.2.21, AcId: 1
S-PE = T: 2, L: 12, Global-id: 65536, Prefix: 192.0.2.5, AcId: 0
"

77 2015/06/03 12:07:01.09 UTC MINOR: DEBUG #2001 Base LDP
"LDP: LDP
Send Label Mapping packet (msgId 8687) to 192.0.2.5:0
Protocol version = 1
Label 262132 advertised for the following FECs
Service FEC GENPWE3: ENET(5)
AGI = type: 1, len: 8, val: 00:00
SAII = T: 2, L: 12, Global-id: 65536, Prefix: 192.0.2.21, AcId: 1
TAII = T: 2, L: 12, Global-id: 65536, Prefix: 192.0.2.11, AcId: 1
Group ID = 0 cBit = 1
Interface parameter Mtu = 1500
Interface parameter VCCV = 0x306
PW status bits = 0x0
Recv Label Mapping packet (msgId 30) from 192.0.2.3:0

```

For the other T-PE, it is the other way round. PE-1 receives a label mapping packet first before it sends a label mapping packet back.

This default behavior can be modified by the signaling command. When set to master, the T-PE will send a label mapping message regardless of the SAII and TAII. By default the parameter is set to auto (which means the T-PE will trigger label mapping if SAII>TAII).

```

*A:PE-1# configure service epipe 1 spoke-sdp-fec 21 signaling
    - signaling <signaling>

    <signaling>                : auto|master

*A:PE-1# configure service epipe 2 spoke-sdp-fec 21
    shutdown
    signaling master
    no shutdown
    exit

```

The MS-PW routing implementation on the 7x50 supports single-sided auto-provisioning. This allows it to have “hub” T-PEs where the TAII is not required to be configured and as such

simplifies the provisioning. In this case, the spoke T-PE PWs would be configured with specific SAII and TAII as well as signaling master, whereas the hub T-PE PWs would be configured with only the SAII and the auto-config parameter. When the auto-config attribute is set for a spoke-sdp-fec, the T-PE always passively waits for the label mapping to be received before issuing a label mapping message (since it does not know the TAII beforehand). This is a CLI example for a hub T-PE spoke-sdp-fec:

```
*A:PE-2# configure service epipe 2
      spoke-sdp-fec 21 fec 129 aii-type 2 create endpoint CORE
          auto-config
          precedence primary
          pw-template-bind 1
          saii-type2 65536:192.0.2.21:1
          no shutdown
      exit
```

Spoke-SDP-FEC Timers

MS-PW routing provides a few timers that can be configured at the global pw-routing level or at each specific spoke-sdp-fec level:

```
*A:PE-1# configure service pw-routing
    boot-timer 20
    retry-timer 40
    retry-count 50

*A:PE-1# configure service epipe 2 spoke-sdp-fec 21
    retry-timer 10
    retry-count 10
```

Where:

- Boot-timer (the default is 10 seconds with values 0 — 600 seconds allowed): Configures a hold-off timer for MS-PW routing advertisements and signaling that is used at boot time. This timer helps to make sure all the network infrastructure is up and running before setting up the PWs.
- Retry-timer (the default is 30 seconds with values 10 — 480 seconds allowed): The exponential back-off timer that determines the interval between consecutive retries to re-establish a spoke-SDP. The configured value gives the initial retry time. The attempt fails if a label withdrawal is received. If configured at global and spoke-sdp-fec level, the latter overrides the value set by the global settings.
- Retry-count (the default 30 with values 10 — 10000): Specifies the number of attempts the system should make to re-establish the spoke-SDP after it has failed. After each successful attempt, the counter is reset to zero. When the specified number is reached, no more attempts are made and the spoke-sdp is put into the shutdown state. Use the **no shutdown** command to bring up the path after the retry limit is exceeded. It is present at the pw-routing level as well as the spoke-SDP level. If configured at global and spoke-sdp-fec level, the latter overrides the value set by the global settings.
- The usual endpoint level timers are also available for MS-PW routing:
 - Revert-time <time-value|infinite> (default is 0, values 0-600 sec): configures the time to wait before reverting to the primary spoke-sdp-fec.
 - Active-hold-delay (the default is 0, values 0 — 60 deci-seconds): It specifies that the node will delay sending the T-LDP status bits for VLL endpoint when the MC-LAG transitions the LAG subgroup which hosts the SAP from active to standby (MC-Ring or MC-APS are supported too) or when any object in the endpoint, i.e. SAP, ICB, or regular spoke SDP, transitions from up to down operational state. The active-hold-delay range starts from 1 (in units of deci-sec) via CLI, and the only way to get the default value of zero is to use the **no active-hold-delay** command

Standby Signaling

Just as with a regular endpoint with regular spoke-sdps, there can also be standby-signaling-master and standby-signaling-slave parameters for spoke-sdp-fecs.

The standby-signaling-master command is configured under the end-point context and makes sure that standby signaling (TLDLP pseudowire status bits 0x20) is sent for the selected standby pseudowire.

```
*A:PE-1# configure service epipe 2 endpoint "CORE" standby-signaling-master
```

It is not allowed to add a SAP associated to an endpoint configured as standby-signaling-master to an Epipe.

```
*A:PE-1>config>service>epipe# sap 1/1/4:2 endpoint "CORE" create
MINOR: SVCMGR #6025 The endpoint has standby-signaling-master configured
```

Standby-signaling-master cannot be set if SAPs have been configured at the end-point (for MC-LAG/Ring/APS or ICB).

```
*A:PE-1>config>service>epipe>endpoint# standby-signaling-master
MINOR: SVCMGR #3805 The command is not allowed in an endpoint with sap
```

The standby-signaling-slave can be configured at endpoint or spoke-sdp-fec level (if the spoke-sdp-fec is not part of an endpoint) but never on both at the same time:

```
*A:PE1>config>service>epipe>endpoint# info
-----
standby-signaling-slave

*A:PE1>config>service>epipe>spoke-sdp-fec# standby-signaling-slave
MINOR: SVCMGR #2031 Sdp-bind is in an explicit endpoint

*A:PE1>config>service>epipe# info
-----
sap 1/1/3:3 create
exit
spoke-sdp-fec 11 fec 129 aii-type 2 create
standby-signaling-slave
```

When this parameter is configured, the node will block the transmit forwarding direction of a spoke SDP based on the pseudowire standby bit received from a TLDLP peer.

Spoke-SDP-FEC Templates and Filters

PW-templates are the way to configure the control word for this type of pseudowire as well as ingress/egress filters (ipv4/mac/ipv6). It is important to note that filters are only supported on the T-PEs, since there is no provisioning of a pw-template (or Epipe at all) on the S-PEs.

```
*A:PE-1# configure service
      pw-template 1 create
        controlword
        egress
          filter ip 1
        exit
      exit

*A:PE-1# configure service
      epipe 2 customer 1 create
---snip---
      spoke-sdp-fec 22 fec 129 aii-type 2 create endpoint CORE
        standby-signaling-slave
        pw-template-bind 1
        saii-type2 65536:192.0.2.12:1
        taii-type2 65536:192.0.2.22:1
        no shutdown
      exit
```

Note that pw-template changes (just like for VPLS with BGP-AD or BGP-VPLS) are not automatically propagated. A tools perform command is provided to evaluate and distribute the changes at the service level to one or all the services that use that template (if the service ID is omitted, then all the services will be updated).

```
*A:PE-1# tools perform service id 2 eval-pw-template 1 allow-service-impact
```

Intra-AS MS-PW Routing

This section provides a configuration example for an intra-AS scenario. The following network setup will be used for this section.

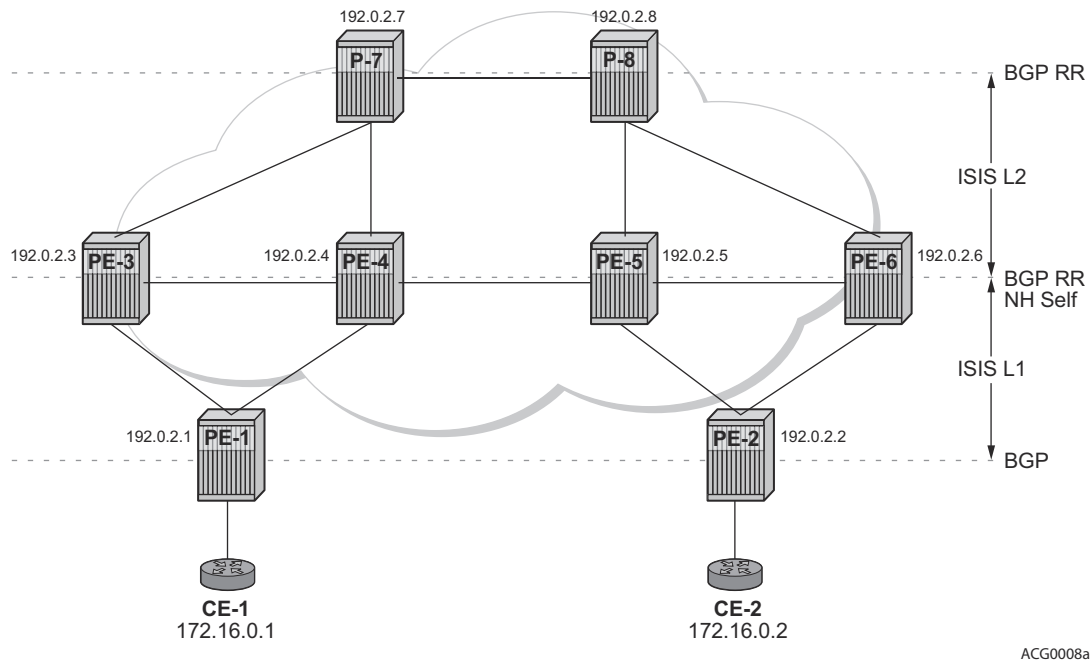


Figure 184: Intra-AS MS-PW Network Topology

Multiple MS-PW routing Epipes are to be configured between PE-1 and PE-2, with PE-3, PE-4, PE-5 and PE-6 being S-PE routers. P-7 and P-8 are pure P routers from a data plane perspective.

All the PEs are pre-configured with ISIS as the IGP, as shown in the figure: PE-1 and PE-2 are level-1 routers, P-7 and P-8 are level-2 only routers and the rest of the routers are level-1/level-2. Link level LDP is also pre-configured on all the network interfaces and targeted LDP is configured between PE-1 and PE-3/PE-4, between PE-2 and PE-5/PE-6 and among PE-3, PE-4, PE-5 and PE-6. There is no targeted LDP sessions configured on P-7 and P-8.

As outlined in [Figure 183](#), the configuration is a three-step process where the pw-routing context is configured first, then the required configuration so that routing tables get populated accordingly and finally the services themselves.

MS-PW using BGP Routing

In this sub-section, Epipe 2 will be configured between PE-1 and PE-2, where TLDP will use the BGP routes populated in the MS-PW routing table to signal the MS-PW.

The first step is the provisioning of the pw-routing context on all the T-PEs and S-PEs. The spe-address will be configured on all the T-PEs and S-PEs (that is, all the routers except for P-7 and P-8) using the ASN as the global-id and the system address as the prefix. On PE-1 and PE-2 (only) the prefixes used for setting up Epipe 2 are configured. Note that two prefixes are configured per T-PE so that pseudowire redundancy with path diversity for the standby pseudowire can be carried out. The spe-address and local-prefixes for the T-PEs are shown below. Note that the advertise-bgp parameter is required since we are using BGP here.

```
*A:PE-1# configure service pw-routing
      spe-address 65536:192.0.2.1
      local-prefix 65536:192.0.2.11 create
          advertise-bgp route-distinguisher 65536:11 community 65535:11
      exit
      local-prefix 65536:192.0.2.12 create
          advertise-bgp route-distinguisher 65536:12 community 65535:12
      exit

*A:PE-2# configure service pw-routing

      spe-address 65536:192.0.2.2
      local-prefix 65536:192.0.2.21 create
          advertise-bgp route-distinguisher 65536:21 community 65535:11
      exit
      local-prefix 65536:192.0.2.22 create
          advertise-bgp route-distinguisher 65536:22 community 65535:12
      exit
```

The second step is the configuration of BGP.

As depicted in [Figure 184](#), BGP is enabled in all the routers. Note that the middle routers (PE-3, PE-4 and PE-5, PE-6) are BGP route-reflectors for PE-1 and PE-2 and they reflect MS-PW routes while changing the next-hop to their own system address. This is required so that TLDP knows where to send the label mapping message for a particular prefix. P-7 and P-8 are regular RRs reflecting routes among all the S-PEs. The BGP configuration of PE-1, PE-3, PE-4 and a P-7 is shown below. Similar commands are configured on the other PEs depending on their T-PE, S-PE or RR function.

The T-PEs have dual-homed BGP sessions to the S-PEs. Example for PE-1:

```
*A:PE-1# configure router
  policy-options
    begin
    policy-statement "export_ms-pw"
      entry 10
        from
          family ms-pw
        exit
        action accept
        exit
      exit
    exit
  commit
exit
```

```
*A:PE-1# configure router
  bgp
    enable-peer-tracking
    rapid-withdrawal
    group "region"
      family ms-pw
      type internal
      export "export_ms-pw"
      neighbor 192.0.2.3
      exit
      neighbor 192.0.2.4
      exit
    exit
  no shutdown
exit
exit
```

The S-PEs are reflecting routes and also changing the NH and Local Preference based on the communities accordingly, so that pseudowire diversity can be ensured.

```
*A:PE-3# configure router
  policy-options
    begin
    community "65535:11" members "65535:11"
    community "65535:12" members "65535:12"
    policy-statement "export_ms-pw_ABR-to-core"
      entry 10
        from
          protocol bgp
          community "65535:11"
          family ms-pw
        exit
        action accept
          local-preference 150
          next-hop-self
        exit
      exit
    entry 20
      from
        protocol bgp
```

```

        community "65535:12"
        family ms-pw
    exit
    action accept
        local-preference 100
        next-hop-self
    exit
exit
exit
policy-statement "export_ms-pw_ABR-to-region"
    entry 10
        from
            protocol bgp
            community "65535:11"
            family ms-pw
        exit
        action accept
            local-preference 150
            next-hop-self
        exit
    exit
exit
    entry 20
        from
            protocol bgp
            community "65535:12"
            family ms-pw
        exit
        action accept
            local-preference 100
            next-hop-self
        exit
    exit
exit
commit
exit

```

```

*A:PE-3# configure router bgp
    rapid-withdrawal
    group "core"
        family ms-pw
        type internal
        export "export_ms-pw_ABR-to-core"
        neighbor 192.0.2.7
        exit
        neighbor 192.0.2.8
        exit
    exit
    group "region"
        family ms-pw
        type internal
        cluster 3.3.3.3
        export "export_ms-pw_ABR-to-region"
        enable-peer-tracking
        neighbor 192.0.2.1
        exit
    exit
no shutdown

```

The second S-PE to which PE-1 is connected has the following BGP configuration:

```
*A:PE-4# configure router
  policy-options
    begin
      community "65535:11" members "65535:11"
      community "65535:12" members "65535:12"
      policy-statement "export_ms-pw_ABR-to-core"
        entry 10
          from
            protocol bgp
            community "65535:12"
            family ms-pw
          exit
          action accept
            local-preference 150
            next-hop-self
          exit
        exit
      entry 20
        from
          protocol bgp
          community "65535:11"
          family ms-pw
        exit
        action accept
          local-preference 100
          next-hop-self
        exit
      exit
    exit
  policy-statement "export_ms-pw_ABR-to-region"
    entry 10
      from
        protocol bgp
        community "65535:12"
        family ms-pw
      exit
      action accept
        local-preference 150
        next-hop-self
      exit
    exit
  entry 20
    from
      protocol bgp
      community "65535:11"
      family ms-pw
    exit
    action accept
      local-preference 100
      next-hop-self
    exit
  exit
exit
commit
exit
```

```

*A:PE-4# configure router
    bgp
        rapid-withdrawal
        group "core"
            family ms-pw
            type internal
            export "export_ms-pw_ABR-to-core"
            neighbor 192.0.2.7
            exit
            neighbor 192.0.2.8
            exit
        exit
        group "region"
            family ms-pw
            type internal
            cluster 4.4.4.4
            export "export_ms-pw_ABR-to-region"
            enable-peer-tracking
            neighbor 192.0.2.1
            exit
        exit
    no shutdown
exit
exit

```

Finally this is the BGP configuration for P-7 and P-8. These are pure RRs.

```

*A:P-7# configure router
    bgp
        enable-peer-tracking
        rapid-withdrawal
        group "core"
            family ms-pw
            type internal
            cluster 1.1.1.1
            neighbor 192.0.2.3
            exit
            neighbor 192.0.2.4
            exit
            neighbor 192.0.2.5
            exit
            neighbor 192.0.2.6
            exit
        exit
    no shutdown
exit

```

After BGP is properly configured and the BGP update exchange takes place, the RIBs are properly populated and the required prefixes uploaded into the MS-PW routing table. An example for PE-1's RIB and pseudowire routing table is provided below.

```

*A:PE-1# show router bgp routes ms-pw
=====
BGP Router ID:192.0.2.1          AS:65536          Local AS:65536
=====
Legend -

```

MS-PW using BGP Routing

```
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
                l - leaked
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP MSPW Routes
=====
Flag  Network                RD
      Nexthop              AII-Type2/Preflen
      As-Path
-----
u*>?  65536:192.0.2.21        65536:21
      192.0.2.3             65536:192.0.2.21:0/64
      No As-Path
*?    65536:192.0.2.21        65536:21
      192.0.2.4             65536:192.0.2.21:0/64
      No As-Path
u*>?  65536:192.0.2.22        65536:22
      192.0.2.4             65536:192.0.2.22:0/64
      No As-Path
*?    65536:192.0.2.22        65536:22
      192.0.2.3             65536:192.0.2.22:0/64
      No As-Path
---snip---
-----

*A:PE-1# show service pw-routing route-table
=====
Service PW L2 Routing Information
=====
AII-Type2/Prefix-Len      Next-Hop      Owner  Age
Route-Distinguisher      Community     Best
-----
65536:192.0.2.11:0/64    192.0.2.1     local  01h03m37s
65536:11                  65535:11      yes
65536:192.0.2.12:0/64    192.0.2.1     local  01h03m37s
65536:12                  65535:12      yes
---snip---
65536:192.0.2.21:0/64    192.0.2.3     bgp    00h16m42s
65536:21                  65535:11      yes
65536:192.0.2.22:0/64    192.0.2.4     bgp    00h17m31s
65536:22                  65535:12      yes
-----
```

It is important to note that the two prefixes advertised by PE-2 are properly learned by PE-1 through two different next hops. Now, use each one with a different pseudowire and make sure that the active and standby pseudowires follow different paths in the network.

Once the routes are installed in the MS-PW routing table, configure the services on PE-1 and PE-2.

```
*A:PE-1# configure service
      pw-template 1 create
      controlword
      exit
      epipe 2 customer 1 create
      description "ms-pw epipe with bgp - using 2 prefixes"
```



```

endpoint "CORE" create
    description "end-point for epipe A/S PW redundancy"
    revert-time 10
exit
sap 1/1/4:2 create
exit
spoke-sdp-fec 21 fec 129 aii-type 2 create endpoint CORE
    precedence primary
    pw-template-bind 1
    sai-type2 65536:192.0.2.11:1
    taii-type2 65536:192.0.2.21:1
    no shutdown
exit
spoke-sdp-fec 22 fec 129 aii-type 2 create endpoint CORE
    pw-template-bind 1
    sai-type2 65536:192.0.2.12:1
    taii-type2 65536:192.0.2.22:1
    no shutdown
exit
no shutdown
exit

*A:PE-2# configure service
pw-template 1 create
    controlword
exit
epipe 2 customer 1 create
    description "ms-pw epipe with bgp - using 2 prefixes"
    endpoint "CORE" create
        description "end-point for epipe A/S PW redundancy"
        revert-time 10
    exit
    sap 1/1/4:2 create
    exit
    spoke-sdp-fec 21 fec 129 aii-type 2 create endpoint CORE
        precedence primary
        pw-template-bind 1
        sai-type2 65536:192.0.2.21:1
        taii-type2 65536:192.0.2.11:1
        no shutdown
    exit
    spoke-sdp-fec 22 fec 129 aii-type 2 create endpoint CORE
        pw-template-bind 1
        sai-type2 65536:192.0.2.22:1
        taii-type2 65536:192.0.2.12:1
        no shutdown
    exit
    no shutdown
exit

```

The following command can be executed to check that the service and spoke-sdp-fecs are up:

```
*A:PE-1# show service id 2 base
=====
Service Basic Information
=====
Service Id      : 2                      Vpn Id      : 0
Service Type    : Epipe
Name            : (Not Specified)
Description     : ms-pw epipe with bgp - using 2 prefixes
Customer Id     : 1                      Creation Origin : manual
Last Status Change: 06/02/2015 12:27:05
Last Mgmt Change : 06/02/2015 12:27:05
Test Service    : No
Admin State     : Up                      Oper State    : Up
MTU             : 1514
Vc Switching    : False
SAP Count       : 1                      SDP Bind Count : 2
Per Svc Hashing : Disabled
Force QTag Fwd  : Disabled

-----
Service Access & Destination Points
-----
Identifier                               Type      AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/4:2                             q-tag     1518    1518    Up   Up
sdp:17405:4294967292 SB(192.0.2.4)      MS-PW     0        1556    Up   Up
sdp:17406:4294967293 SB(192.0.2.3)      MS-PW     0        1556    Up   Up
=====
*A:PE-1#
```

Note that the sdp-binding identifiers and sdp identifiers are automatically generated by the system.

Use vccv-trace to check that the spoke-sdp-fecs for the active and standby pseudowires follow different and disjoint paths:

```
*A:PE-1# oam vccv-trace spoke-sdp-fec 21

VCCV-TRACE with 120 bytes of MPLS payload
1 192.0.2.3 rtt=1.75ms rc=8(DSRtrMatchLabel)
2 192.0.2.5 rtt=5.38ms rc=8(DSRtrMatchLabel)
3 192.0.2.2 rtt=8.34ms rc=3(EgressRtr)

*A:PE-1# oam vccv-trace spoke-sdp-fec 22

VCCV-TRACE with 120 bytes of MPLS payload
1 192.0.2.4 rtt=1.80ms rc=8(DSRtrMatchLabel)
2 192.0.2.6 rtt=5.48ms rc=8(DSRtrMatchLabel)
3 192.0.2.2 rtt=7.83ms rc=3(EgressRtr)
```

MS-PW using Static Routing

In this sub-section, Epipe 3 will be configured between PE-1 and PE-2, where TLDP will use static-routes in the MS-PW routing table to signal the MS-PW.

On PE-1 and PE-2 (only) we will configure the prefixes used for setting up Epipe 3. Those could be the same as used for Epipe 2, however we will use different ones in this example. Note that the **no advertise-bgp** parameter is required now. The static routes for each remote prefix are also configured. Since we will also have pseudowire redundancy for Epipe 3, two prefixes with static-routes pointing at different next-hops will be used:

```
*A:PE-1# configure service pw-routing

    spe-address 65536:192.0.2.1
    local-prefix 65536:192.0.2.13 create
    exit
    local-prefix 65536:192.0.2.14 create
    exit
    static-route 65536:192.0.2.23:192.0.2.3
    static-route 65536:192.0.2.24:192.0.2.4

*A:PE-2# configure service pw-routing

    spe-address 65536:192.0.2.2
    local-prefix 65536:192.0.2.23 create
    exit
    local-prefix 65536:192.0.2.24 create
    exit
    static-route 65536:192.0.2.13:192.0.2.5
    static-route 65536:192.0.2.14:192.0.2.6
```

It is important to note that static-routes are also required at all S-PEs along the path (keeping the path diversity for the prefixes as well) and for both directions:

```
*A:PE-3# configure service pw-routing
    spe-address 65536:192.0.2.3
    static-route 65536:192.0.2.13:192.0.2.1
    static-route 65536:192.0.2.23:192.0.2.5

*A:PE-4# configure service pw-routing

    spe-address 65536:192.0.2.4
    static-route 65536:192.0.2.14:192.0.2.1
    static-route 65536:192.0.2.24:192.0.2.6
```

Finally, once the MS-PW routing tables are properly populated, the services can be configured and brought up:

```
*A:PE-1# configure service
pw-template 1 create
  controlword
exit
epipe 3 customer 1 create
  description "ms-pw epipe with static routes"
  endpoint "CORE" create
    description "end-point for epipe A/S PW redundancy"
    revert-time 10
    standby-signaling-master
  exit
  sap 1/1/4:3 create
  exit
  spoke-sdp-fec 31 fec 129 aii-type 2 create endpoint CORE
    precedence primary
    pw-template-bind 1
    saii-type2 65536:192.0.2.13:31
    taii-type2 65536:192.0.2.23:31
    no shutdown
  exit
  spoke-sdp-fec 32 fec 129 aii-type 2 create endpoint CORE
    pw-template-bind 1
    saii-type2 65536:192.0.2.14:32
    taii-type2 65536:192.0.2.24:32
    no shutdown
  exit
  no shutdown
exit

*A:PE-2# configure service
pw-template 1 create
  controlword
exit
epipe 3 customer 1 create
  description "ms-pw epipe with static routes"
  endpoint "CORE" create
    description "end-point for epipe A/S PW redundancy"
    revert-time 10
    standby-signaling-master
  exit
  sap 1/1/4:3 create
  exit
  spoke-sdp-fec 31 fec 129 aii-type 2 create endpoint CORE
    precedence primary
    pw-template-bind 1
    saii-type2 65536:192.0.2.23:31
    taii-type2 65536:192.0.2.13:31
    no shutdown
  exit
  spoke-sdp-fec 32 fec 129 aii-type 2 create endpoint CORE
    pw-template-bind 1
    saii-type2 65536:192.0.2.24:32
    taii-type2 65536:192.0.2.14:32
    no shutdown
```

```

exit
no shutdown
exit

```

Check the status and path of the spoke-sdp-fecs with the proper show commands and oam vccv-trace/ping commands (see previous sub-section).

MS-PW using Explicit Paths

In this sub-section, Epipe 4 will be configured between PE-1 and PE-2, where TLDP will use explicit paths to signal the MS-PW, overriding the information given by the MS-PW routing table. Although this mode requires the specific configuration of the hops, one by one, the configuration is only done on the T-PEs, as opposed to the static-routes where all the S-PEs must be configured with static routes (a mixed of static-routes and BGP routes can coexist). The local-prefixes shown for Epipe 3 will be re-used here for Epipe 4.

Now path-1 and path-2 will be configured hop by hop, using diverse paths. Note that all the S-PE nodes as well as the terminating T-PE must be included in the path.

```

*A:PE-1# configure service
    pw-routing
        spe-address 65536:192.0.2.1
        local-prefix 65536:192.0.2.13 create
        exit
        local-prefix 65536:192.0.2.14 create
        exit
        path "path-1" create
            hop 1 192.0.2.3
            hop 2 192.0.2.5
            hop 3 192.0.2.2
            no shutdown
        exit
        path "path-2" create
            hop 1 192.0.2.4
            hop 2 192.0.2.6
            hop 3 192.0.2.2
            no shutdown
        exit
    exit

```

```

*A:PE-2# configure service
    pw-routing
        spe-address 65536:192.0.2.2
        local-prefix 65536:192.0.2.23 create
        exit
        local-prefix 65536:192.0.2.24 create
        exit
        path "path-1" create
            hop 1 192.0.2.5

```

MS-PW using Explicit Paths

```
        hop 2 192.0.2.3
        hop 3 192.0.2.1
        no shutdown
    exit
    path "path-2" create
        hop 1 192.0.2.6
        hop 2 192.0.2.4
        hop 3 192.0.2.1
        no shutdown
    exit
exit
```

And now, those paths must be specified when configuring the Epipe:

```
*A:PE-1# configure service
    epipe 4 customer 1 create
        description "ms-pw epipe with explicit paths"
        endpoint "CORE" create
            description "end-point for epipe A/S PW redundancy"
            revert-time 10
            standby-signaling-master
        exit
        sap 1/1/4:4 create
        exit
        spoke-sdp-fec 41 fec 129 aii-type 2 create endpoint CORE
            precedence primary
            saii-type2 65536:192.0.2.13:41
            taii-type2 65536:192.0.2.23:41
            path "path-1"
            no shutdown
        exit
        spoke-sdp-fec 42 fec 129 aii-type 2 create endpoint CORE
            saii-type2 65536:192.0.2.14:42
            taii-type2 65536:192.0.2.24:42
            path "path-2"
            no shutdown
        exit
        no shutdown
    exit
```

```
*A:PE-2# configure service
    epipe 4 customer 1 create
        description "ms-pw epipe with explicit paths"
        endpoint "CORE" create
            description "end-point for epipe A/S PW redundancy"
            revert-time 10
        exit
        sap 1/1/4:4 create
        exit
        spoke-sdp-fec 41 fec 129 aii-type 2 create endpoint CORE
            precedence primary
            saii-type2 65536:192.0.2.23:41
            taii-type2 65536:192.0.2.13:41
            path "path-1"
            no shutdown
        exit
```

```
spoke-sdp-fec 42 fec 129 aii-type 2 create endpoint CORE
    sai-type2 65536:192.0.2.24:42
    tai-type2 65536:192.0.2.14:42
    path "path-2"
    no shutdown
exit
no shutdown
exit
```

Now, check the status and path of the spoke-sdp-fecs with the proper **show** commands and **oam vccv-trace/ping** commands (see previous sub-section).

Inter-AS MS-PW Routing

This section provides a configuration example for an inter-AS scenario, using BGP tunnels between ASBRs and BGP as the MS-PW routing mechanism. The following network setup will be used in this section.

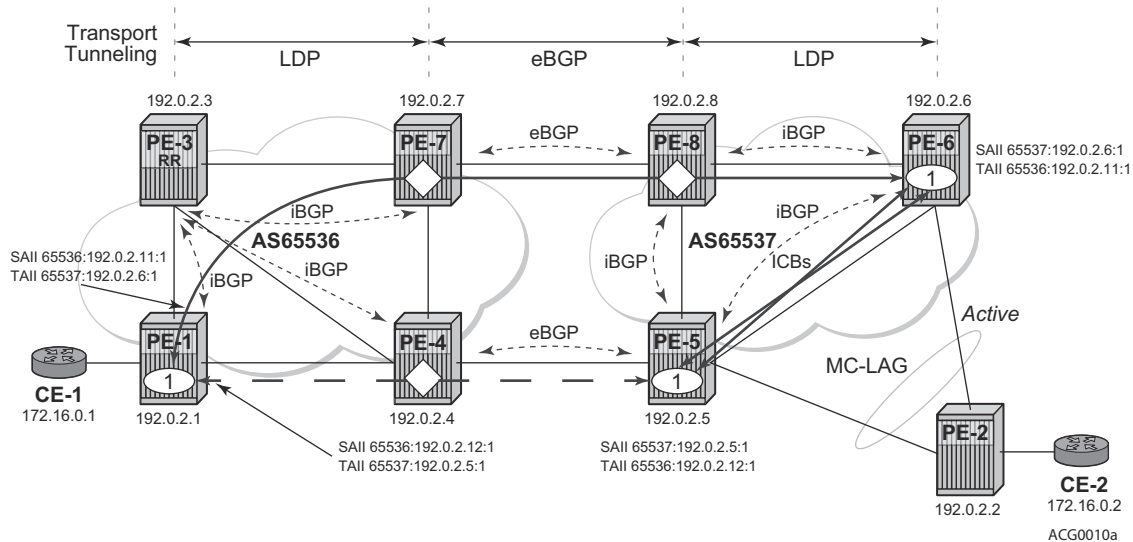


Figure 185: Inter-AS MS-PW Network Topology

In this example, only one Epipe is configured (Epipe 1, using MS-PW BGP routing). The T-PEs are PE-1, PE-5 and PE-6. PE-7, PE-8 and PE-4 are S-PEs.

A/S pseudowire redundancy together with MC-LAG at one end will be used, as depicted in the diagram. ICB (Inter-chassis Backup) spoke SDPs between PE-5 and PE-6 are required in order to forward the in-flight packets while MC-LAG and A/S pseudowire are converging, in case of network failures. Those ICBs will also be signaled following the MS-PW routing procedures.

The network setup in the figure is pre-configured with the following settings:

- There are two AS (65536 and 65537) which are connected by two ASBR pairs (PE-7/PE-4 and PE-8/PE-5) running eBGP between them. These eBGP sessions will be used to exchange ipv4-labels (to setup the transport BGP-LBL tunnel, according to the RFC 3107) and MS-PW NLRIs.
- Within AS65536, PE-3 is used as an RR to reflect the MS-PW routes. In AS65537 there is a full mesh of iBGP sessions to distribute the MS-PW routes.
- ISIS is used within each AS.

- LDP is used as a transport MPLS signaling protocol within each AS and a BGP tunnel will be used between the ASBRs (note that MS-PW routing supports LDP or BGP tunnels as transport).
- A redundant MC-LAG access to PE-6 and PE-5 is configured.

The next section will go through the configuration required to set up a redundant Epipe between CE-1 and CE-2, by combining A/S pseudowire in the network and MC-LAG at the access.

MS-PW using BGP Routing

Epipe 1 will be configured at the end of this section, including the active and redundant pseudowires from PE-1 to PE-5/PE-6, as well as the required ICBs and SAPs at the access.

As discussed, the first step is the provisioning of the pw-routing context. Again, the spe-address must be provisioned in all T-PEs and S-PEs whereas prefixes are mandatory only on the T-PEs involved in the service. The following shows the prefixes configured on PE-1, PE-5 and PE-6. Note that two prefixes are needed in PE-1 in order to make sure that active and standby pseudowires follow disjoint paths.

```
*A:PE-1# configure service pw-routing
      spe-address 65536:192.0.2.1
      local-prefix 65536:192.0.2.11 create
      advertise-bgp route-distinguisher 65536:11 community 65535:11
      exit
      local-prefix 65536:192.0.2.12 create
      advertise-bgp route-distinguisher 65536:12 community 65535:12
      exit

*A:PE-5# configure service pw-routing
      spe-address 65537:192.0.2.5
      local-prefix 65537:192.0.2.5 create
      advertise-bgp route-distinguisher 65537:5 community 65535:5
      exit

*A:PE-6# configure service pw-routing
      spe-address 65537:192.0.2.6
      local-prefix 65537:192.0.2.6 create
      advertise-bgp route-distinguisher 65537:6 community 65535:6
      exit
```

Once the spe-addresses and prefixes have been provisioned, BGP must be configured accordingly. An example of the configuration at PE-1 and PE-6 is shown below. Note that a simple BGP export-policy is used to export all the local MS-PW prefixes.

```
*A:PE-1# configure router
      policy-options
        begin
          policy-statement "export_ms-pw"
            entry 10
              from
                family ms-pw
              exit
              action accept
              exit
            exit
          exit
        commit
      exit
```

```
*A:PE-1# configure router
      bgp
        min-route-advertisement 1
        rapid-withdrawal
        group "intra-AS"
          family ms-pw
          type internal
          export "export_ms-pw"
          neighbor 192.0.2.3
          exit
        exit
        no shutdown
      exit
    exit
```

```
*A:PE-6# configure router
      policy-options
        begin
          policy-statement "export_ms-pw"
            entry 10
              from
                family ms-pw
              exit
              action accept
              exit
            exit
          exit
        commit
      exit
```

```
*A:PE-1# configure router
      bgp
        min-route-advertisement 1
        enable-peer-tracking
        rapid-withdrawal
        group "intra-AS"
          family ms-pw
          type internal
```

```

        export "export_ms-pw"
        neighbor 192.0.2.5
        exit
        neighbor 192.0.2.8
        exit
    exit
    no shutdown
exit
exit

```

At the ASBR, the BGP policies are a little more complex since the following tasks must be accomplished:

- ASBR IPv4 system addresses must be exported to the peer ASBR to establish the RFC 3107 BGP tunnel between ASBRs.
- BGP export policies must be used so that MS-PW NLRI exchange can be controlled and attributes like MED (towards the remote AS) and/or local-preference (towards the local AS) can be modified.
- Finally, BGP import policies must also be used to modify the MS-PW route NH (next-hops) since the TLDP next signaling hop must match a peer TLDP system address.

As an example of ASBR BGP configuration, PE-4 and PE-7 are shown below.

Note that the prefixes 65536:192.0.2.11 and 65537:192.0.2.6 must be preferred in the PE-7/PE-8 pair whereas the prefixes 65536:192.0.2.12 and 65537:192.0.2.5 must be preferred in the PE-4/PE-5 pair, so that the pseudowires are established as depicted in [Figure 185](#). The preference can be propagated by using the BGP MED (use the local preference (LP) within the AS (LP is not relevant to eBGP)). The following CLI excerpt shows an example of how to modify MED and LP, as well as changing the NH with an import policy.

```

*A:PE-4# configure router
    policy-options
        begin
        prefix-list "system"
            prefix 192.0.2.4/32 exact
        exit
        community "65535:5" members "65535:5"
        community "65535:6" members "65535:6"
        community "65535:11" members "65535:11"
        community "65535:12" members "65535:12"
        policy-statement "ASBR to ASBR"
            entry 10
                from
                    protocol bgp
                    community "65535:12"
                    family ms-pw
                exit
            action accept
                origin igp
                metric set 50
            exit
        exit
    exit

```

MS-PW using BGP Routing

```
entry 20
  from
    protocol bgp
    community "65535:11"
    family ms-pw
  exit
  action accept
    origin igp
    metric set 100
  exit
exit
exit
policy-statement "ASBR to region"
  entry 10
    from
      protocol bgp
      community "65535:5"
      family ms-pw
    exit
    action accept
      origin igp
      local-preference 150
      next-hop-self
    exit
  exit
entry 20
  from
    protocol bgp
    community "65535:6"
    family ms-pw
  exit
  action accept
    origin igp
    next-hop-self
  exit
exit
exit
policy-statement "export_ipv4_system"
  entry 10
    from
      prefix-list "system"
    exit
    action accept
      origin igp
    exit
  exit
exit
policy-statement "import ms-pw NH change"
  entry 10
    from
      protocol bgp
      family ms-pw
    exit
    action accept
      next-hop 192.0.2.5
    exit
  exit
exit
exit
commit
```

```

exit

*A:PE-4# configure router      bgp
    min-route-advertisement 1
    enable-peer-tracking
    rapid-withdrawal
    group "inter-AS"
        family ipv4 ms-pw
        type external
        import "import ms-pw NH change"
        export "export_ipv4_system" "ASBR to ASBR"
        local-as 65536
        peer-as 65537
        neighbor 192.168.45.2
            advertise-label ipv4
        exit
    exit
    group "intra-AS"
        family ms-pw
        type internal
        export "ASBR to region"
        neighbor 192.0.2.3
        exit
    exit
    no shutdown
exit
exit

*A:PE-7# configure router
    policy-options
        begin
        prefix-list "system"
            prefix 192.0.2.7/32 exact
        exit
        community "65535:5" members "65535:5"
        community "65535:6" members "65535:6"
        community "65535:11" members "65535:11"
        community "65535:12" members "65535:12"
        policy-statement "ASBR to ASBR"
            entry 10
                from
                    protocol bgp
                    community "65535:11"
                    family ms-pw
                exit
                action accept
                origin igp
                metric set 50
            exit
        exit
        entry 20
            from
                protocol bgp
                community "65535:12"
                family ms-pw
            exit
            action accept
            origin igp
            metric set 100

```

MS-PW using BGP Routing

```
        exit
    exit
exit
policy-statement "ASBR to region"
    entry 10
        from
            protocol bgp
            community "65535:6"
            family ms-pw
        exit
        action accept
            origin igp
            local-preference 150
            next-hop-self
        exit
    exit
exit
entry 20
    from
        protocol bgp
        community "65535:5"
        family ms-pw
    exit
    action accept
        origin igp
        next-hop-self
    exit
exit
exit
policy-statement "export_ipv4_system"
    entry 10
        from
            prefix-list "system"
        exit
        action accept
            origin igp
        exit
    exit
exit
policy-statement "import ms-pw NH change"
    entry 10
        from
            protocol bgp
            family ms-pw
        exit
        action accept
            next-hop 192.0.2.8
        exit
    exit
exit
commit
exit
```

```
*A:PE-7# configure router      bgp
min-route-advertisement 1
enable-peer-tracking
rapid-withdrawal
transport-tunnel mpls
group "inter-AS"
```

```

        family ipv4 ms-pw
        type external
        import "import ms-pw NH change"
        export "export_ipv4_system" "ASBR to ASBR"
        local-as 65536
        peer-as 65537
        neighbor 192.168.78.2
            advertise-label ipv4
        exit
    exit
    group "intra-AS"
        family ms-pw
        type internal
        export "ASBR to region"
        neighbor 192.0.2.3
        exit
    exit
    no shutdown
exit
exit

```

PE-5 and PE-8 have similar configurations to the one shown above. Note that PE-5 is a T-PE as well as an ASBR, therefore a local MS-PW prefix must be exported as opposed to only remote prefixes (that is, some export entries for the local MS-PW routes will not contain **protocol bgp** in the matching criteria).

After BGP is properly configured and the updates get exchanged, the RIBs are populated and the prefixes uploaded onto the MS-PW routing table as shown below for PE-1 and PE-6.

```

*A:PE-1# show router bgp routes ms-pw
=====
BGP Router ID:192.0.2.1      AS:65536      Local AS:65536
=====
BGP MSPW Routes
=====
Flag  Network      RD
      Nexthop    AII-Type2/Preflen
      As-Path
-----
---snipped---
u*>i  65537:192.0.2.5      65537:5
      192.0.2.4        65537:192.0.2.5:0/64
      65537
u*>i  65537:192.0.2.6      65537:6
      192.0.2.7        65537:192.0.2.6:0/64
      65537
-----
Routes : 4

*A:PE-1# show service pw-routing route-table
=====
Service PW L2 Routing Information
=====
AII-Type2/Prefix-Len      Next-Hop      Owner  Age

```

MS-PW using BGP Routing

Route-Distinguisher	Community	Best
65536:192.0.2.11:0/64	192.0.2.1	local 30d22h25m
0:0	0:0	yes
65536:192.0.2.11:0/64	192.0.2.1	local 30d22h24m
65536:11	65535:11	yes
65536:192.0.2.12:0/64	192.0.2.1	local 30d22h19m
0:0	0:0	yes
65536:192.0.2.12:0/64	192.0.2.1	local 30d22h19m
65536:12	65535:12	yes
65537:192.0.2.5:0/64	192.0.2.4	bgp 02h43m25s
65537:5	65535:5	yes
65537:192.0.2.6:0/64	192.0.2.7	bgp 02h45m49s
65537:6	65535:6	yes

Entries found: 6

*A:PE-1#

*A:PE-6# show router bgp routes ms-pw

```

=====
BGP Router ID:192.0.2.6      AS:65537      Local AS:65537
=====
BGP MSPW Routes
=====
Flag   Network      RD
      Nexthop    AII-Type2/Preflen
      As-Path
-----
u*>i  65536:192.0.2.11  65536:11
      192.0.2.8     65536:192.0.2.11:0/64
      65536
u*>i  65536:192.0.2.12  65536:12
      192.0.2.5     65536:192.0.2.12:0/64
      65536
u*>i  65537:192.0.2.5   65537:5
      192.0.2.5     65537:192.0.2.5:0/64
      No As-Path
-----

```

Routes : 3

*A:PE-6#

*A:PE-6# show service pw-routing route-table

```

=====
Service PW L2 Routing Information
=====
AII-Type2/Prefix-Len      Next-Hop      Owner  Age
Route-Distinguisher      Community     Best
-----
65536:192.0.2.11:0/64     192.0.2.8     bgp    02h52m08s
65536:11                  65535:11      yes
65536:192.0.2.12:0/64     192.0.2.5     bgp    02h38m51s
65536:12                  65535:12      yes
65537:192.0.2.5:0/64      192.0.2.5     bgp    02h38m51s
65537:5                   65535:5       yes

```



```

65537:192.0.2.6:0/64          192.0.2.6      local  28d02h16m
0:0                          0:0           yes
65537:192.0.2.6:0/64          192.0.2.6      local  28d02h16m
65537:6                      65535:6       yes
-----
Entries found: 5
=====
*A:PE-6#

```

As can be seen in the show commands, the two PE-1 prefixes are learned on PE-5 and PE-6 through different and disjoint paths, and the PE-5 and PE-6 prefixes are learned by PE-1 through two different and disjoint paths.

The last step is the service configuration on the three T-PEs, as shown below. Note that TLDP sessions must have been previously and explicitly configured between the T-PEs and S-PEs (i.e., between PE-1 and PE-4/7, between PE-4 and PE-5, PE-7 and PE-8 and between PE-6 and PE-5/8).

```

*A:PE-1# configure router ldp
      targeted-session
        peer 192.0.2.4
        exit
        peer 192.0.2.7
        exit
      exit

*A:PE-1# configure service
      pw-template 1 create
        controlword
      exit
      epipe 1 customer 1 create
        description "ms-pw epipe with bgp, inter-AS, MC-LAG redundancy"
        endpoint "CORE" create
          description "end-point for epipe A/S PW redundancy"
        exit
        sap 1/1/4:1 create
        exit
        spoke-sdp-fec 11 fec 129 aii-type 2 create endpoint CORE
          precedence primary
          pw-template-bind 1
          saii-type2 65536:192.0.2.11:1
          taii-type2 65537:192.0.2.6:1
          no shutdown
        exit
        spoke-sdp-fec 12 fec 129 aii-type 2 create endpoint CORE
          pw-template-bind 1
          saii-type2 65536:192.0.2.12:1
          taii-type2 65537:192.0.2.5:1
          no shutdown
        exit
        no shutdown
      exit

*A:PE-5# configure service

```

MS-PW using BGP Routing

```
pw-template 1 create
  controlword
exit
epipe 1 customer 1 create
  description "ms-pw epipe with bgp, inter-AS, MC-LAG redundancy"
  endpoint "CORE" create
    description "end-point for epipe A/S PW redundancy"
  exit
  endpoint "ACCESS" create
  exit
  sap lag-1:1 endpoint "ACCESS" create
  exit
  spoke-sdp-fec 11 fec 129 aii-type 2 create endpoint CORE
    pw-template-bind 1
    saii-type2 65537:192.0.2.5:1
    taii-type2 65536:192.0.2.12:1
    no shutdown
  exit
  spoke-sdp-fec 12 fec 129 aii-type 2 create endpoint CORE icb
    pw-template-bind 1
    saii-type2 65537:192.0.2.5:2
    taii-type2 65537:192.0.2.6:2
    no shutdown
  exit
  spoke-sdp-fec 13 fec 129 aii-type 2 create endpoint ACCESS icb
    pw-template-bind 1
    saii-type2 65537:192.0.2.5:3
    taii-type2 65537:192.0.2.6:3
    no shutdown
  exit
  no shutdown
exit

*A:PE-6# configure service
pw-template 1 create
  controlword
exit
epipe 1 customer 1 create
  description "ms-pw epipe with bgp, inter-AS, MC-LAG redundancy"
  endpoint "CORE" create
    description "end-point for epipe A/S PW redundancy"
  exit
  endpoint "ACCESS" create
  exit
  sap lag-1:1 endpoint "ACCESS" create
  exit
  spoke-sdp-fec 11 fec 129 aii-type 2 create endpoint CORE
    pw-template-bind 1
    saii-type2 65537:192.0.2.6:1
    taii-type2 65536:192.0.2.11:1
    no shutdown
  exit
  spoke-sdp-fec 12 fec 129 aii-type 2 create endpoint CORE icb
    pw-template-bind 1
    saii-type2 65537:192.0.2.6:3
    taii-type2 65537:192.0.2.5:3
    no shutdown
  exit
```

```

spoke-sdp-fec 13 fec 129 aii-type 2 create endpoint ACCESS icb
pw-template-bind 1
saii-type2 65537:192.0.2.6:2
taii-type2 65537:192.0.2.5:2
no shutdown
exit
no shutdown
exit

```

The following show commands can be executed to check the status of the Epipe 1 and the pseudowire status signaling received:

```

*A:PE-1# show service id 1 base
=====
Service Basic Information
=====
Service Id      : 1                Vpn Id          : 0
Service Type    : Epipe
Name            : (Not Specified)
Description     : ms-pw epipe with bgp, inter-AS, MC-LAG redundancy
Customer Id     : 1                Creation Origin  : manual
Last Status Change: 06/01/2015 09:52:53
Last Mgmt Change : 06/01/2015 09:56:07
Test Service    : No
Admin State     : Up                Oper State      : Up
MTU             : 1514
Vc Switching    : False
SAP Count       : 1                SDP Bind Count  : 2
Per Svc Hashing : Disabled
Force QTag Fwd  : Disabled
-----
Service Access & Destination Points
-----
Identifier      Type      AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/4:1     q-tag    1518    1518    Up   Up
sdp:17406:4294967290 SB(192.0.2.7) MS-PW    0      1978    Up   Up
sdp:17407:4294967291 SB(192.0.2.4) MS-PW    0      1978    Up   Up
=====

*A:PE-1# show service id 1 endpoint
=====
Service 1 endpoints
=====
Endpoint name    : CORE
Description      : end-point for epipe A/S PW redundancy
Creation Origin  : manual
Revert time      : 0
Act Hold Delay   : 0
Standby Signaling Master : false
Standby Signaling Slave  : false
Tx Active (SDP-FEC)    : 11
Tx Active Up Time      : 0d 01:04:59
Revert Time Count Down : N/A
Tx Active Change Count : 12
Last Tx Active Change  : 06/01/2015 09:56:07
-----

```

Members

```

-----
Sdp-fec: 11 Prec:0                               Oper Status: Up
Sdp-fec: 12 Prec:4                               Oper Status: Up
=====

```

Note that PE-5 will have the MC-LAG standby interface and as such the SAP will be operationally down and will drive the standby signaling to the remote T-PEs:

```
*A:PE-5# show service id 1 base
```

Service Basic Information

```

=====
Service Id      : 1                      Vpn Id      : 0
Service Type    : Epipe
Name            : (Not Specified)
Description     : ms-pw epipe with bgp, inter-AS, MC-LAG redundancy
Customer Id     : 1                      Creation Origin : manual
Last Status Change: 06/01/2015 09:52:58
Last Mgmt Change  : 06/01/2015 09:52:58
Test Service    : No
Admin State     : Up                      Oper State    : Up
MTU             : 1514
Vc Switching    : False
SAP Count       : 1                      SDP Bind Count : 3
Per Svc Hashing : Disabled
Force QTag Fwd  : Disabled

```

Service Access & Destination Points

```

-----
Identifier                                     Type      AdmMTU  OprMTU  Adm  Opr
-----
sap:lag-1:1                                   q-tag     1518    1518    Up   Down
sdp:17402:4294967280 SB(192.0.2.6)            MS-PW     0       1978    Up   Up
sdp:17402:4294967281 SB(192.0.2.6)            MS-PW     0       1978    Up   Up
sdp:17403:4294967282 SB(192.0.2.4)            MS-PW     0       1978    Up   Up
=====

```

```
*A:PE-5# show service id 1 all | match Flag
```

```

Flags      : None
Flags      : None
Flags      : None
Flags      : PortOperDown StandByForMcProtocol

```

The following commands are useful on the S-PEs in order to find the PWs automatically created as well as the SDPs automatically used for those PWs.

```
*A:PE-7# show service sdp-using
```

SDP Using

```

=====
SvcId      SdpId      Type      Far End      Opr      I.Label  E.Label
State
-----
2147483647 17406:4294967294 MS-PW     192.0.2.1    Up       131070   131069

```

```

2147483647 17407:4294967295    MS-PW  192.0.2.8                Up      131071  131065
-----
Number of SDPs : 2
-----
=====
*A:PE-7#

```

As it can be seen above, two PWs (type MS-PW) have been automatically created over two also automatically created SDPs: 17406 and 17407. SDP 17406 is built over an LDP tunnel whereas SDP 17407 runs over a BGP tunnel.

```

*A:PE-7# show router tunnel-table
=====
IPv4 Tunnel Table (Router: Base)
=====
Destination      Owner      Encap TunnelId  Pref    Nexthop      Metric
-----
192.0.2.1/32     sdp        MPLS  17406      5       192.0.2.1     0
192.0.2.1/32     ldp        MPLS  65547      9       192.168.37.1  20
192.0.2.3/32     ldp        MPLS  65545      9       192.168.37.1  10
192.0.2.4/32     ldp        MPLS  65546      9       192.168.47.1  10
192.0.2.8/32     sdp        MPLS  17407      5       192.0.2.8     0
192.0.2.8/32     bgp        MPLS  262146     12      192.168.78.2  1000
-----
Flags: B = BGP backup route available
      E = inactive best-external BGP route
=====
*A:PE-7#

*A:PE-7# show service sdp 17406 detail | match "Active LSP"
Mixed LSP Mode      : Enabled                Active LSP Type      : LDP

*A:PE-7# show service sdp 17407 detail | match "Active LSP"
Mixed LSP Mode      : Enabled                Active LSP Type      : BGP

```

In addition to all of the recommended show commands, **vccv-ping** and **vccv-trace** are two extremely useful commands in this environment. **vccv-trace** can even help to trace the traffic going through the ICBs under failure situations.

Conclusion

Service Providers are always seeking highly scalable VLL services that can be deployed with the lowest operational cost. The SR OS supports MS-PW routing according to the draft-ietf-pwe3-dynamic-ms-pw. MS-PW routing allows the Service Provider to deploy ELINE services without having to provision services in the core of the network. In other words, MS-PW enables end-point provisioning in highly scalable seamless MPLS networks, through the use of BGP. Alternatively, static MS-PW routes or explicit paths can also be used.

The examples used in this section illustrate the configuration of MS-PW routing in intra-AS and inter-AS scenarios. Show and OAM commands have also been suggested so that the operator can verify and troubleshoot the MS-PW routing paths and procedures.

In This Chapter

This section provides information about Provider Backbone Bridging (PBB) — Ethernet Virtual Leased Line in an MPLS-based network which is applicable to all of the 7750 SR and 7450 ESS routers.

Topics in this section include:

- [Applicability on page 1244](#)
- [Overview on page 1245](#)
- [Configuration on page 1247](#)
- [Conclusion on page 1267](#)

Applicability

This section is applicable to all 7750 SR and 7450 ESS series and was tested on release 13.0.R1.
There are no specific prerequisites required.

Overview

The draft-ietf-l2vpn-pbb-vpls-pe-model-00, *Extensions to VPLS PE model for Provider Backbone Bridging*, describes the PBB-VPLS model supported by SR OS. This model expands the VPLS PE model to support PBB as defined by the IEEE 802.1ah.

The PBB model is organized around a B-component (backbone instance) and an I-component (customer instance). In Alcatel-Lucent's implementation of the PBB model, the use of an Epipe as I-component is allowed for point-to-point services. Multiple I-VPLS and Epipe services can be all mapped to the same B-VPLS (backbone VPLS instance).

The use of Epipe scales the E-Line services as no MAC switching, learning or replication is required in order to deliver the point-to-point service. All packets ingressing the customer SAP are PBB-encapsulated and unicasted through the B-VPLS tunnel using the backbone destination MAC of the remote PBB PE. All the packets ingressing the B-VPLS destined for the Epipe are PBB de-encapsulated and forwarded to the customer SAP.

Some use cases for PBB-Epipe are:

- Get a more efficient and scalable solution for point-to-point services:
 - Up to 8K VPLS services per box are supported (including I-VPLS or B-VPLS) and using I-VPLS for point-to-point services takes VPLS resources as well as unnecessary customer MAC learning. A better solution is to connect a PBB-Epipe to a B-VPLS instance, where there is no customer MAC switching/learning.
- Take advantage of the pseudowire aggregation in the M:1 model:
 - Many Epipe services may use only a single service and set of pseudowires over the backbone.
- Have a uniform provisioning model for both point-to-point (Epipe) and multipoint (VPLS) services.
 - Using the PBB-Epipe, the core MPLS/pseudowire infrastructure does not need to be modified: the new Epipe inherits the existing pseudowire and MPLS structure already configured on the B-VPLS and there is no need for configuring new tunnels or pseudowire switching instances at the core.

Knowledge of the PBB-VPLS architecture and functionality on the service router family is assumed throughout this section. For additional information, refer to the relevant Alcatel-Lucent user documentation.

The following network setup will be used throughout the rest of the chapter.

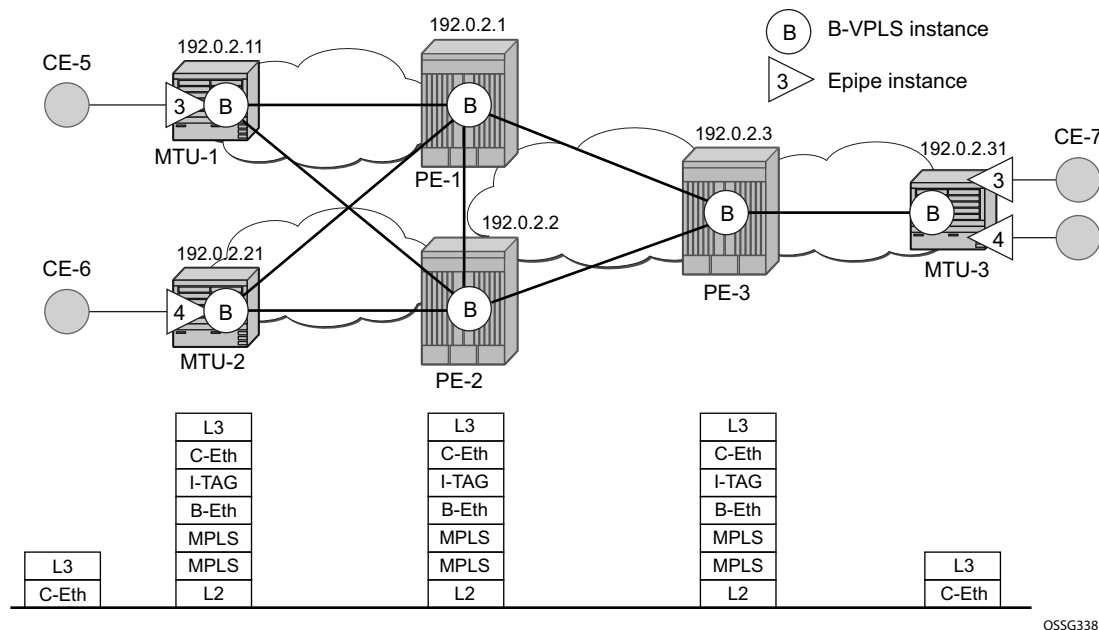


Figure 186: Network Topology

The setup consists of a three 7x50 SR/ESS (PE-1, PE-2 and PE-3) core and three Multi-Tenant Unit (MTU) nodes connected to the core. A backbone VPLS instance (B-VPLS 101) will be defined in all the six nodes, whereas two Epipe services will be defined as illustrated in [Figure 186](#) (Epipe 3 in nodes MTU-1 and MTU-3, Epipe 4 in nodes MTU-2 and MTU-3). Those Epipe services will be multiplexed into the common B-VPLS 101, using the I-Service ID (ISID) field within the I-TAG as the demultiplexer field required at the egress MTU to differentiate each specific customer. Note that I-VPLS and Epipe services can be mapped to the same B-VPLS.

The B-VPLS domain constitutes a H-VPLS network itself, with spoke SDPs from the MTUs to the core PE layer. Active/standby (A/S) spoke SDPs can be used from the MTUs to the PEs (like in the MTU-1 and MTU-2 cases) or single non-redundant spoke SDPs (like MTU-3).

The protocol stack being used along the path between the CEs is represented in [Figure 186](#).

Configuration

This section describes all the relevant PBB-Epipe configuration tasks for the setup shown in [Figure 186](#). Note that the appropriate B-VPLS and associated IP/MPLS configuration is out of the scope of this document. In this particular example the following protocols will be configured beforehand in the core:

- ISIS-TE as IGP with all the interfaces being level-2. Alternatively OSPF could have been used.
- RSVP-TE as the MPLS protocol to signal the transport tunnels.
- LSPs between core PEs will be fast re-route protected (facility bypass tunnels) whereas LSP tunnels between MTUs and PEs will not be protected.
- The protection between MTU-1, MTU-2 and PE-1, PE-2 will be based on the A/S pseudowire protection configured in the B-VPLS.
- BGP is configured for auto-discovery, BGP-AD (Layer 2 VPN family), since FEC 129 will be used to establish the pseudowires between PEs in the core (FEC 128 between MTU and PE nodes).

Once the IP/MPLS infrastructure is up and running, the service configuration tasks described in the following sections can be implemented.

PBB Epipe Service Configuration

In this particular example, the Epipes 3 and 4 are using the B-VPLS 101 in the core. The same B-VPLS which is multiplexing the Epipe services into a common service provider infrastructure can also be used to connect the I-VPLS instances existing in the network for multipoint services.

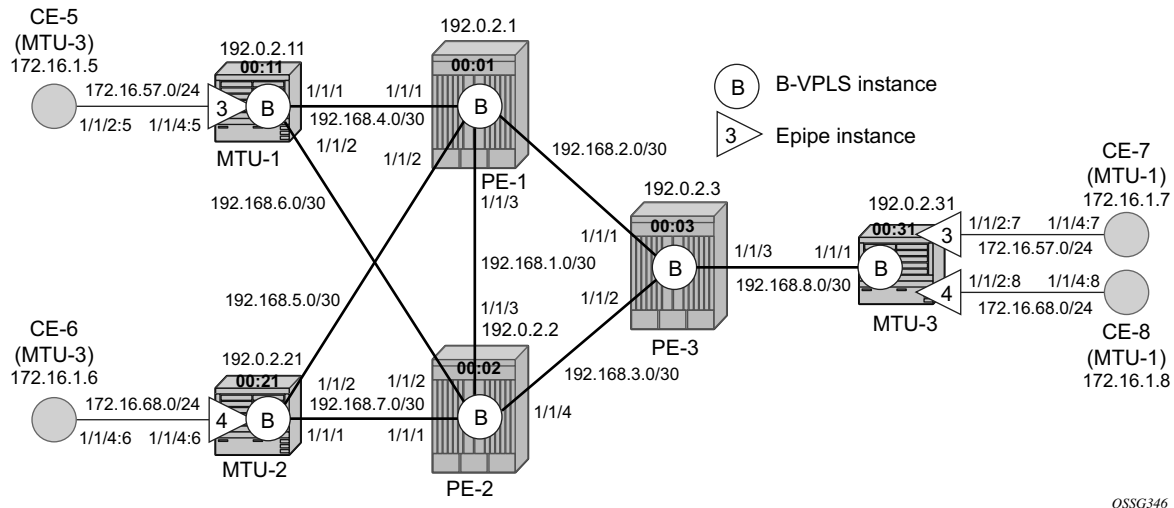


Figure 187: Setup Detailed View

OSSG346

B-VPLS and PBB Configuration

First, configure the B-VPLS instance that will carry the PBB traffic. There is no specific requirement on the B-VPLS to support Epipes. The following shows the B-VPLS configuration on MTU-1 and PE-1.

```
# for MTU-1
configure
service
  vpls 101 customer 1 b-vpls create
    service-mtu 2000
    pbb
      source-bmac 00:11:11:11:11:11
    exit
    stp
      shutdown
    exit
    endpoint "core" create
      no suppress-standby-signaling
    exit
    spoke-sdp 111:101 endpoint "core" create
      stp
        shutdown
      exit
      precedence primary
      no shutdown
    exit
    spoke-sdp 112:101 endpoint "core" create
      stp
        shutdown
      exit
      no shutdown
    exit
    no shutdown
  exit

# for PE-1
configure
service
  pw-template 1 use-provisioned-sdp create
    split-horizon-group "CORE"
  exit
  exit
  vpls 101 customer 1 b-vpls create
    service-mtu 2000
    pbb
      source-bmac 00:01:01:01:01:01
    exit
    bgp
      route-target export target:65000:101 import target:65000:101
      pw-template-binding 1
    exit
  exit
  bgp-ad
    vpls-id 65000:101
```

PBB Epipe Service Configuration

```
        no shutdown
    exit
    stp
        shutdown
    exit
    spoke-sdp 111:101 create
        no shutdown
    exit
    spoke-sdp 121:101 create
        no shutdown
    exit
    no shutdown
exit
```

The relevant B-VPLS commands are in **bold**.

Note that the keyword **b-vpls** is given at creation time and therefore it cannot be added to an existing regular VPLS instance. Besides the **b-vpls** keyword, the B-VPLS is a regular VPLS instance in terms of configuration, with the following exceptions:

- The B-VPLS service MTU must be at least 18 bytes greater than the Epipe MTU of the multiplexed instances. In this example, the I-VPLS instances will have the default service MTU (1514 bytes) hence any MTU equal or greater than 1532 bytes must be configured. In this particular example, a MTU of 2000 bytes is configured in the B-VPLS instance throughout the network.
- The source B-MAC is the MAC that will be used as a source when the PBB traffic is originated from that node. Note that you can configure a source B-MAC per B-VPLS instance (if there are more than one B-VPLS) or a common source B-MAC that will be shared by all the B-VPLS instances in the node. The way to configure a common B-MAC is shown below:

```
configure
  service
    pbb
      source-bmac 00:11:11:11:11:11
```

The following considerations will be taken into account when configuring the B-VPLS:

- B-VPLS SAPs:
 - Ethernet DOT1Q and NULL encapsulations are supported.
 - Default SAP types are blocked in the CLI for the B-VPLS SAP.

- B-VPLS SDPs:
 - For MPLS, both mesh and spoke SDPs with split horizon groups are supported.
 - Similar to regular pseudowire, the outgoing PBB frame on an SDP (for example, Bpseudowire) contains a BVID Qtag only if the pseudowire type is Ethernet VLAN (vc-type=vlan). If the pseudowire type is Ethernet (vc-type=ether), the BVID qtag is stripped before the frame goes out.
 - BGP-AD is supported in the B-VPLS, therefore, spoke SDPs in the B-VPLS can be signalled using FEC 128 or FEC 129. In this example, BGP-AD and FEC 129 are used. A split-horizon group has been configured to emulate the behavior of mesh SDPs in the core.
- While Multiple MAC Registration Protocol (MMRP) is useful to optimize the flooding in the B-VPLS domain and build a flooding tree on a per I-VPLS basis, it does not have any effect for Epipes since the destination B-MAC used for Epipes is always the destination B-MAC configured in the Epipe and never the group B-MAC corresponding to the ISID.
- If a local Epipe instance is associated with the B-VPLS, local frames originated or terminated on local Epipe(s) are PBB encapsulated or de-encapsulated using the PBB Etype provisioned under the related port or SDP component.

By default, the PBB Etype is 0x88e7 (which is the standard one defined in the 802.1ah, indicating that there is an I-TAG in the payload) but this PBB Etype can be changed if required due to interoperability reasons. This is the way to change it at port and/or SDP level:

```
A:MTU-1# configure port 1/1/1 ethernet pbb-etype
- pbb-etype <0x0600..0xffff>
- no pbb-etype

<0x0600..0xffff>      : [1536..65535] - accepts in decimal or hex

A:MTU-1# configure service sdp 111 pbb-etype
- no pbb-etype [<0x0600..0xffff>]
- pbb-etype <0x0600..0xffff>

<0x0600..0xffff>      : [1536..65535] - accepts in decimal or hex
```

The following commands are useful to check the actual PBB etype.

```
A:MTU-1# show service sdp 111 detail | match PBB
Bw BookingFactor      : 100                PBB Etype                : 0x88e7
A:MTU-1#

A:MTU-1# show port 1/1/1 | match PBB
PBB Ethertype       : 0x88e7
A:MTU-1#
```

PBB Epipe Service Configuration

Before the next step, the Epipe configuration, the operator can optionally configure MAC names under the PBB context. MAC names will simplify the Epipe provisioning later on and in case of any change on the remote node MAC address, only one configuration modification is required as opposed as one change per affected Epipe (potentially thousands of Epipes which are terminated onto the same remote node). The MAC names are configured under the service PBB CLI context:

```
*A:MTU-1# configure service pbb mac-name
- mac-name <name> <ieee-address>
- no mac-name <name>

<name>                : 32 char max
<ieee-address>        : xx:xx:xx:xx:xx:xx or xx-xx-xx-xx-xx-xx

*A:MTU-1>config>service# info
-----
pbb
  source-bmac 00:11:11:11:11:11
  mac-name "MTU-1" 00:11:11:11:11:11
  mac-name "MTU-2" 00:21:21:21:21:21
  mac-name "MTU-3" 00:31:31:31:31:31
exit

---snipped---

*A:MTU-1>config>service#
```


Epipe Configuration

Once the common B-VPLS is configured, the next step is the provisioning of the customer Epipe instances. For PBB-Epipes, the I-component or Epipe is composed of an I-SAP and a PBB tunnel endpoint which points to the backbone destination MAC address (B-DA).

The following outputs show the relevant CLI configuration for the two Epipe instances represented in [Figure 187 on page 1248](#). The Epipe instances are configured on the MTU devices, whereas the core PEs are kept as customer-unaware nodes.

The following shows the relevant Epipe commands on MTU-3.

```
configure
service
  pbb
    source-bmac 00:31:31:31:31:31
    mac-name "MTU-1" 00:11:11:11:11:11
    mac-name "MTU-2" 00:21:21:21:21:21
    mac-name "MTU-3" 00:31:31:31:31:31
  exit
  epipe 3 customer 1 create
    description "pbb epipe number 3"
    pbb
      tunnel 101 backbone-dest-mac "MTU-1" isid 3
    exit
    sap 1/1/2:7 create
    exit
    no shutdown
  exit
  epipe 4 customer 1 create
    description "pbb epipe number 4"
    pbb
      tunnel 101 backbone-dest-mac "MTU-2" isid 4
    exit
    sap 1/1/2:8 create
    exit
    no shutdown
  exit
```

It is not required to configure a node with its own MAC address, so the line defining the mac-name MTU-3 can be omitted.

The following shows the relevant configuration on MTU-1 and MTU-2.

```
# for MTU-1
configure
  service
    epipe 3 customer 1 create
      description "pbb epipe number 3"
      pbb
        tunnel 101 backbone-dest-mac "MTU-3" isid 3
      exit
    sap 1/1/4:5 create
    exit
  no shutdown
exit

# for MTU-2
configure
  service
    epipe 4 customer 1 create
      description "pbb epipe number 4"
      pbb
        tunnel 101 backbone-dest-mac "MTU-3" isid 4
      exit
    sap 1/1/4:6 create
    exit
  no shutdown
exit
```

All Ethernet SAPs supported by a regular Epipe are also supported in the PBB Epipe. Note that spoke SDPs are not supported in PBB-Epipes, for example, no spoke SDP is allowed when PBB tunnels are configured on the Epipe.

The PBB tunnel links the SAP configured to the B-VPLS 101 existing in the core. The following parameters are accepted in the PBB tunnel configuration:

```
A:MTU-2>config>service>epipe>pbb# tunnel
- no tunnel
- tunnel <service-id> backbone-dest-mac <mac-name> isid <ISID>
- tunnel <service-id> backbone-dest-mac <ieee-address> isid <ISID>

<service-id>      : [1..2148007978] |<svc-name:64 char max>
<mac-name>        : 32 char max
<ieee-address>    : xx:xx:xx:xx:xx:xx or xx-xx-xx-xx-xx-xx
<ISID>            : [0..16777215]
```

Where:

- The service-id matches the B-VPLS ID.
- The **backbone-dest-mac** can be given by a MAC name (as in this configuration example) or the MAC itself. It is recommended to use MAC names, as explained in the previous section.
- The ISID must be specified.

Flood Avoidance in PBB-Epipes

As already discussed in the previous section, when provisioning a PBB Epipe, the remote **backbone-dest-mac** must be explicitly configured on the PBB tunnel so that the ingress PBB node can build the 802.1ah encapsulation.

If the configured remote backbone-destination-mac is not known in the local FDB, the Epipe customer frames will be 802.1ah encapsulated and flooded into the B-VPLS until the MAC is learned. As previously discussed, MMRP does not help to minimize the flooding because the PBB Epipes always use the configured **backbone-destination-mac** for flooding traffic as opposed to the group B-MAC derived from the ISID.

Flooding could be indefinitely prolonged in the following cases:

- Configuration mistake of the **backbone-destination-mac**. The service will not work but the operator will not detect the mistake since the customer traffic is not dropped at the source node. Every single frame is turned into an unknown unicast PBB frame and hence flooded into the B-VPLS domain.
- Change the **backbone-smac** in the remote PE B-VPLS instance.
- There is only unidirectional traffic in the Epipe service. In this case, the backbone-dest-mac will never be learned in the local SFIB and the frames will always be flooded into the B-VPLS domain.
- The remote node owning the **backbone-destination-mac** simply goes down.

In any of those cases, the operator can easily check whether the PBB Epipe is flooding into the B-VPLS domain, just by looking at the flood flag in the following command output:

```
*A:MTU-1# show service id 3 base
=====
Service Basic Information
=====
Service Id      : 3                Vpn Id          : 0
Service Type    : Epipe
Name            : (Not Specified)
Description     : pbb epipe number 3
Customer Id     : 1                Creation Origin  : manual
Last Status Change: 02/27/2015 08:22:59
Last Mgmt Change  : 02/27/2015 08:22:59
Test Service    : No
Admin State     : Up              Oper State      : Up
MTU             : 1514
Vc Switching    : False
SAP Count       : 1              SDP Bind Count  : 0
Per Svc Hashing : Disabled
Force QTag Fwd  : Disabled

-----
Service Access & Destination Points
-----
```

Flood Avoidance in PBB-Epipes

```
Identifier                               Type      AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/4:5                             q-tag     1518    1518    Up   Up

-----
PBB Tunnel Point
-----
B-vpls      Backbone-dest-MAC  Isid      AdmMTU  OperState  Flood  Oper-dest-MAC
-----
101          MTU-3                  3          2000    Up          Yes    00:31:31:31:31:31
-----
Last Status Change: 02/27/2015 08:22:59
Last Mgmt Change   : 02/27/2015 08:22:59
=====
*A:MTU-1#
```

In this particular example, the PBB Epipe 3 is flooding into the B-VPLS 101, as the flood flag indicates. The operator can also confirm that the operational destination B-MAC for the pbb-tunnel, MTU-3, has not been learned in the B-VPLS FDB:

```
A:MTU-1# show service id 101 fdb pbb
=====
Forwarding Database, b-Vpls Service 101
=====
MAC                Source-Identifier    iVplsMACs  Epipes    Type/Age
-----
No Matching Entries
=====
A:MTU-1#
```

Flooding Cases 1 and 2 — Wrong backbone-dest-mac

Flooding cases 1 and 2 should be fixed after detecting the flooding (see previous commands) and checking the FDBs and PBB tunnel configurations.

Flooding Case 3 — Unidirectional Traffic: Virtual MEP and CCM Configuration

For flooding case 3 (unidirectional traffic), Alcatel-Lucent recommends the use of ETH-CFM (802.1ag/Y.1731 Connectivity Fault Management) virtual Maintenance End Points (MEPs). By defining a virtual MEP per node terminating a PBB-Epipe, configuring the MEP mac-address to be the source-bmac value and activating continuity check messages (ccm) we achieve a twofold effect:

- The **pbb-tunnel backbone-destination-mac** will always be learned at the local FDB, as long as the remote virtual MEP is active and sending **cc** messages. As a result, there will not be flooding even if we have unidirectional traffic.
- An automatic proactive OAM mechanism exists to detect failures on remote nodes, which ultimately cause unnecessary flooding in the B-VPLS domain.

In the following network example, the virtual MEPs in B-VPLS 101: MEP11, MEP21 and MEP31 are configured.

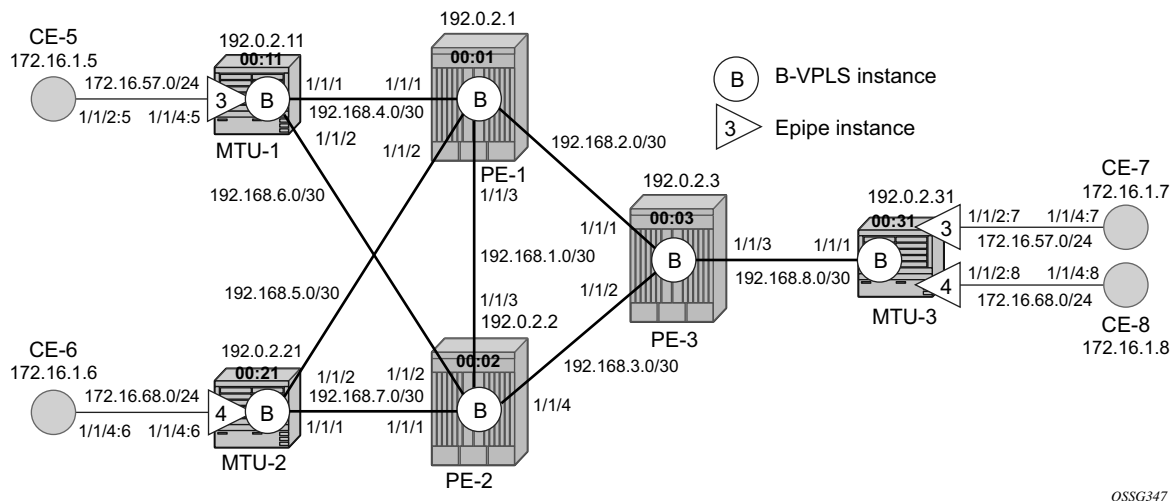


Figure 188: Virtual MEPs for Flooding Avoidance

Flood Avoidance in PBB-Epipes

The following configuration example uses MTU-1. First, the general ETH-CFM configuration is made:

```
configure
  eth-cfm
    domain 1 format none level 3
      association 1 format icc-based name "B-VPLS-000101"
        bridge-identifier 101
      exit
      remote-mepid 21
      remote-mepid 31
    exit
  exit
exit
```

Then the actual virtual MEP configuration is made:

```
configure
  service
    vpls 101
      eth-cfm
        mep 11 domain 1 association 1
          ccm-enable
          mac-address 00:11:11:11:11:11
          no shutdown
        exit
      exit
    exit
exit
```

Note that the MAC address configured for the MEP11 matches the MAC address configured as the **source-bmac** on MTU-1, which is the **backbone-destination-mac** configured on the Epipe 3 pbb-tunnel on MTU-3:

```
# for MTU-1
configure
  service
    pbb
      source-bmac 00:11:11:11:11:11
      mac-name "MTU-1" 00:11:11:11:11:11
      mac-name "MTU-2" 00:21:21:21:21:21
      mac-name "MTU-3" 00:31:31:31:31:31
    exit
```

```
# for MTU-3
configure
  service
    pbb
      source-bmac 00:31:31:31:31:31
      mac-name "MTU-1" 00:11:11:11:11:11
      mac-name "MTU-2" 00:21:21:21:21:21
      mac-name "MTU-3" 00:31:31:31:31:31
    exit
  epipe 3 customer 1 create
    description "pbb epipe number 3"
```

```

pbb
    tunnel 101 backbone-dest-mac "MTU-1" isid 3
exit
sap 1/1/2:7 create
exit
no shutdown
exit

```

Once MEP11 has been configured, check that MTU-3 is receiving **cc** messages from MEP11 with the following command:

```

*A:MTU-3# show eth-cfm mep 31 domain 1 association 1 all-remote-mepids
=====
Eth-CFM Remote-Mep Table
=====
R-mepId AD Rx CC RxRdi Port-Tlv If-Tlv Peer Mac Addr      CCM status since
-----
11      True False Absent Absent 00:11:11:11:11:11 02/27/2015 08:26:07
21      True False Absent Absent 00:21:21:21:21:21 02/27/2015 08:26:07
=====
Entries marked with a 'T' under the 'AD' column have been auto-discovered.
*A:MTU-3#

```

As a result of the **cc** messages coming from MEP11, the MTU-1 MAC is permanently learned in the B-VPLS 101 FDB on node MTU-3, and no flooding exists:

```

*A:MTU-3# show service id 101 fdb pbb
=====
Forwarding Database, b-Vpls Service 101
=====
MAC                Source-Identifier      iVplsMACs  Epipes    Type/Age
-----
00:11:11:11:11:11  sdp:33:101            0           1         L/0
00:21:21:21:21:21  sdp:33:101            0           1         L/0
4a:c4:ff:00:00:00  sdp:33:101            0           0         L/0
=====
*A:MTU-3#
*A:MTU-3# show service id 3 base
=====
Service Basic Information
=====
Service Id          : 3                Vpn Id            : 0
Service Type        : Epipe
Name                 : (Not Specified)
Description          : pbb epipe number 3
Customer Id         : 1                Creation Origin    : manual
Last Status Change  : 02/27/2015 08:23:34
Last Mgmt Change    : 02/27/2015 08:23:34
Test Service        : No
Admin State         : Up                Oper State         : Up
MTU                  : 1514
Vc Switching        : False
SAP Count           : 1                SDP Bind Count     : 0
Per Svc Hashing     : Disabled
Force QTag Fwd      : Disabled
=====

```

Flood Avoidance in PBB-Epipes

Service Access & Destination Points

Identifier	Type	AdmMTU	OprMTU	Adm	Opr
sap:1/1/2:7	q-tag	1518	1518	Up	Up

PBB Tunnel Point

B-vpls	Backbone-dest-MAC	Isid	AdmMTU	OperState	Flood	Oper-dest-MAC
101	MTU-1	3	2000	Up	No	00:11:11:11:11:11

Last Status Change: 02/27/2015 08:23:34

Last Mgmt Change : 02/27/2015 08:23:34

*A:MTU-3#

Flooding Case 4 — Remote Node Failure

If the node owner of the **backbone-dest-mac** fails or gets isolated, the node where the PBB Epipe is initiated will not detect the failure; that is, if MTU-1 fails, the Epipe 3 remote end will also fail but MTU-3 will not detect the failure and as a result of that, MTU-3 will flood the traffic to the network (flooding will occur after MTU-1 MAC is removed from the B-VPLS FDBs, due to either the B-VPLS flushing mechanisms or aging).

In order to avoid/reduce flooding in this case, the following mechanisms are recommended:

- Provision virtual MEPs in the B-VPLS instances terminating PBB Epipes, as already explained. This will guarantee there is no unknown B-MAC unicast being flooded under normal operation.
- CCM timers should be provisioned based on how long the service provider is willing to accept flooding.

```
*A:MTU-3# configure eth-cfm domain 1 association 1 ccm-interval
- ccm-interval <interval>
- no ccm-interval

<interval>                : {10ms|100ms|1|10|60|600} - default 10 seconds
```

- It is possible to also provision discard-unknown in the B-VPLS on the MTUs, i.e. MTU-1, MTU-2 and MTU-3, so that flooded traffic due to the destination MAC being unknown in the B-VPLS is discarded immediately at the MTU. Note that it is important to configure this in conjunction with the CC messages from the virtual MEPs to ensure that the remote B-MACs are learned in both directions. If for any reason the remote B-MACs are not in the MTU B-VPLS, no traffic will be forwarded at all on the PBB-Epipe.
- As soon as the MTU node recovers, it will start sending CC messages and the backbone-mac will be learned on the backbone and MTU nodes again.

```
*A:PE-1# configure service vpls 101 discard-unknown
*A:PE-2# configure service vpls 101 discard-unknown
*A:PE-3# configure service vpls 101 discard-unknown
*A:MTU-1# configure service vpls 101 discard-unknown
*A:MTU-2# configure service vpls 101 discard-unknown
*A:MTU-3# configure service vpls 101 discard-unknown
```

With the recommended configuration in place, in case MTU-1 fails, the **backbone-dest-mac** configured on the pbb-tunnel for Epipe 3 on MTU-3 will be removed from the B-VPLS 101 on all the nodes (either by MAC flush mechanisms on the B-VPLS or by aging). From that point on, traffic originated from CE-7 will be discarded at MTU-3 and won't be flooded further.

As soon as MTU-1 comes back up, MEP11 will start sending CCM and as such the MTU-1 MAC will be learned throughout the B-VPLS 101 domain and in particular in PE-1, PE-3 and MTU-3

(note that CCM PDUs use a multicast address). From the moment MTU-1 MAC is known on the backbone nodes and MTU-3, the traffic won't be discarded any more, but forwarded to MTU-1.

PBB-Epipe Show Commands

The following commands can help to check the PBB Epipe configuration and their related parameters.

For the B-VPLS service:

```
A:MTU-1# show service id 101 base
=====
Service Basic Information
=====
Service Id       : 101                Vpn Id           : 0
Service Type     : b-VPLS
Name             : (Not Specified)
Description      : (Not Specified)
Customer Id      : 1                  Creation Origin   : manual
Last Status Change: 02/27/2015 08:20:25
Last Mgmt Change : 02/27/2015 08:26:23
Etree Mode       : Disabled
Admin State      : Up                 Oper State        : Up
MTU              : 2000               Def. Mesh VC Id   : 101
SAP Count        : 0                 SDP Bind Count    : 2
Snd Flush on Fail : Disabled          Host Conn Verify  : Disabled
Propagate MacFlush: Disabled          Per Svc Hashing   : Disabled
Allow IP Intf Bind: Disabled
Temp Flood Time  : Disabled           Temp Flood        : Inactive
Temp Flood Chg Cnt: 0
VSD Domain       : <none>
SPI load-balance : Disabled
Oper Backbone Src : 00:11:11:11:11:11
Use SAP B-MAC    : Disabled
i-Vpls Count     : 0
Epipe Count      : 1
=====
Service Access & Destination Points
=====
Identifier                               Type      AdmMTU  OprMTU  Adm  Opr
-----
sdp:111:101 S(192.0.2.1)                 Spok      8000    8000    Up   Up
sdp:112:101 S(192.0.2.2)                 Spok      8000    8000    Up   Up
=====
*A:MTU-1#
```

For the Epipe service:

```
*A:MTU-1# show service id 3 base
=====
Service Basic Information
=====
Service Id       : 3                Vpn Id           : 0
Service Type     : Epipe
```

```

Name           : (Not Specified)
Description     : pbb epipe number 3
Customer Id    : 1                      Creation Origin   : manual
Last Status Change: 02/27/2015 08:22:59
Last Mgmt Change  : 02/27/2015 08:22:59
Test Service    : No
Admin State     : Up                    Oper State      : Up
MTU             : 1514
Vc Switching    : False
SAP Count       : 1                    SDP Bind Count   : 0
Per Svc Hashing : Disabled
Force QTag Fwd  : Disabled

```

----- Service Access & Destination Points

Identifier	Type	AdmMTU	OprMTU	Adm	Opr
sap:1/1/4:5	q-tag	1518	1518	Up	Up

----- PBB Tunnel Point

B-vpls	Backbone-dest-MAC	Isid	AdmMTU	OperState	Flood	Oper-dest-MAC
101	MTU-3	3	2000	Up	No	00:31:31:31:31:31

Last Status Change: 02/27/2015 08:22:59

Last Mgmt Change : 02/27/2015 08:22:59

=====

*A:MTU-1#

The following command shows all the Epipe instances multiplexed into a particular B-VPLS and its status.

```
* A:MTU-1# show service id 101 epipe
```

```
=====
```

Related Epipe services for b-Vpls service 101

Epipe SvcId	Oper ISID	Admin	Oper
3	3	Up	Up

Number of Entries : 1

=====

*A:MTU-1#

To check the virtual MEP information, show the local virtual MEPs configured on the node:

```
* A:MTU-1# show eth-cfm cfm-stack-table all-virtuals
```

```
=====
```

CFM Stack Table Defect Legend:

R = Rdi, M = MacStatus, C = RemoteCCM, E = ErrorCCM, X = XconCCM

A = AisRx, L = CSF LOS Rx, F = CSF AIS/FDI rx, r = CSF RDI rx

=====

CFM Virtual Stack Table

=====

PBB-Epipe Show Commands

```

Service                Lvl Dir Md-index  Ma-index  MepId  Mac-address  Defect
-----
101                    3  U      1          1    11  00:11:11:11:11:11  -----
=====
*A:MTU-1#

```

The following command shows all the information related to the remote MEPs configured in the association, for example, the remote virtual MEPs configured in MTU-2 and MTU-3:

```

*A:MTU-1# show eth-cfm mep 11 domain 1 association 1 all-remote-mepids
=====
Eth-CFM Remote-Mep Table
=====
R-mepId AD Rx CC RxRdi Port-Tlv If-Tlv Peer Mac Addr      CCM status since
-----
21      True False Absent Absent 00:21:21:21:21:21 02/27/2015 08:24:03
31      True False Absent Absent 00:31:31:31:31:31 02/27/2015 08:24:57
=====
Entries marked with a 'T' under the 'AD' column have been auto-discovered.
*A:MTU-1#

```

The following command shows the detail information and status of the local virtual MEP configured in MTU-1:

```

*A:MTU-1# show eth-cfm mep 11 domain 1 association 1
=====
Eth-Cfm MEP Configuration Information
=====
Md-index      : 1                      Direction      : Up
Ma-index      : 1                      Admin          : Enabled
MepId         : 11                     CCM-Enable    : Enabled
SvcId         : 101
Description    : (Not Specified)
FngAlarmTime  : 0                      FngResetTime   : 0
FngState      : fngReset                ControlMep     : False
LowestDefectPri : macRemErrXcon          HighestDefect   : none
Defect Flags   : None
Mac Address    : 00:11:11:11:11:11      Collect LMM Stats : disabled
CcmLtmPriority : 7                      CcmPaddingSize  : 0 octets
CcmTx         : 19                     CcmSequenceErr  : 0
CcmIgnoreTLVs : (Not Specified)
Fault Propagation: disabled              FacilityFault    : n/a
MA-CcmInterval : 10                     MA-CcmHoldTime   : 0ms
MA-Primary-Vid : Disabled
Eth-1Dm Threshold: 3(sec)                MD-Level        : 3
Eth-Ais       : Disabled
Eth-Ais Tx defCCM: allDef
Eth-Tst       : Disabled
Eth-CSF       : Disabled

Redundancy:
  MC-LAG State : n/a

CcmLastFailure Frame:
  None

```

```
XconCcmFailure Frame:
    None
```

```
=====
*A:MTU-1#
```

When there is a failure on a remote Epipe node, as discussed, the source node keeps sending traffic. The 802.1ag/Y.1731 virtual MEP configured can help to detect and troubleshoot the problem. For instance, when a failure happens in MTU-3 (node goes down or the B-VPLS instance is shut down), the virtual MEP show commands will show the following information:

```
*A:MTU-1# show eth-cfm mep 11 domain 1 association 1
=====
Eth-Cfm MEP Configuration Information
=====
Md-index          : 1                      Direction       : Up
Ma-index          : 1                      Admin           : Enabled
MepId             : 11                     CCM-Enable      : Enabled
SvcId             : 101
Description        : (Not Specified)
FngAlarmTime      : 0                      FngResetTime    : 0
FngState           : fngDefectReported      ControlMep      : False
LowestDefectPri    : macRemErrXcon          HighestDefect    : defRemoteCCM
Defect Flags       : bDefRDICCM bDefRemoteCCM
Mac Address        : 00:11:11:11:11:11      Collect LMM Stats : disabled
CcmLtmPriority      : 7                      CcmPaddingSize   : 0 octets
CcmTx              : 39                     CcmSequenceErr   : 0
CcmIgnoreTLVs      : (Not Specified)
Fault Propagation  : disabled                FacilityFault    : n/a
MA-CcmInterval     : 10                     MA-CcmHoldTime   : 0ms
MA-Primary-Vid     : Disabled
Eth-1Dm Threshold : 3(sec)                  MD-Level         : 3
Eth-Ais            : Disabled
Eth-Ais Tx defCCM  : allDef
Eth-Tst            : Disabled
Eth-CSF            : Disabled

Redundancy:
    MC-LAG State : n/a

CcmLastFailure Frame:
    None

XconCcmFailure Frame:
    None
=====
*A:MTU-1#
```

The bDefRemoteCCMdefect flag clearly shows that there is a remote MEP in the association which has stopped sending CCMs. In order to find out which node is affected, see the following output:

```
*A:MTU-1# show eth-cfm mep 11 domain 1 association 1 all-remote-mepids
=====
Eth-CFM Remote-Mep Table
=====
```

PBB-Epipe Show Commands

```
R-mepId AD Rx CC RxRdi Port-Tlv If-Tlv Peer Mac Addr      CCM status since
-----
21          True  True  Absent   Absent 00:21:21:21:21:21 02/27/2015 08:24:03
31          False False Absent   Absent 00:00:00:00:00:00 02/27/2015 08:27:53
=====
Entries marked with a 'T' under the 'AD' column have been auto-discovered.
*A:MTU-1#
```

CCMs are no longer received from virtual MEP 31 (the one defined in MTU-3) since 02/27/2015 08:27:53. This conveys which node has failed and when it failed.

Conclusion

Point-to-Point Ethernet services can use the same operational model followed by PBB VPLS for multipoint services. In other words, Epipes can be linked to the same B-VPLS domain being used by I-VPLS instances and use the existing H-VPLS network infrastructure in the core. The use of PBB Epipes reduces dramatically the number of services and pseudowires in the core and therefore allows the service provider to scale the number of ELINE services in the network.

The example used in this document shows the configuration of the PBB Epipes as well as all the related features which are required for this environment. Show commands have also been suggested so that the operator can verify and troubleshoot the service.

In This Chapter

This section provides information about Provider Backbone Bridging (PBB) in an MPLS-based network.

Topics in this section include:

- [Applicability on page 1270](#)
- [Overview on page 1272](#)
- [Configuration on page 1273](#)
- [Conclusion on page 1310](#)

Applicability

This chapter is applicable to the 7x50 series and was tested on Release 13.0.R1. Network-mode D (described in the Access Dual-Homing and MAC-notification section) is required.¹

1.Although it can be used in an MPLS-based PBB network as explained in this document, the MAC notification feature for dual-homed access is normally used in native PBB networks.

Summary

The draft-ietf-l2vpn-pbb-vpls-pe-model-00, *Extensions to LDP Signaling for PBB-VPLS*, describes the PBB-VPLS model supported by SR OS. This model expands the VPLS PE model to support PBB as defined by the IEEE 802.1ah.

PBB-VPLS combines the best of the PBB and VPLS technologies to deliver the most scalable multi-point Layer 2 VPN in the market. PBB-VPLS inherits all the benefits derived from MPLS (for example, sub-50ms FRR protection, traffic engineering, no need for MSTP in the backbone) while greatly increasing the scalability of the network by providing MAC hiding, service multiplexing and pseudowire aggregation.

The SR OS PBB-VPLS implementation also includes support for:

- MMRP (Multiple MAC Registration Protocol, application within IEEE 802.1ak) for flood containment in the backbone instances, as specified in Section 6 of the draft-ietf-l2vpn-pbb-vpls-pe-model.
- Extensions to LDP signaling for PBB-VPLS, according to draft-balus-l2vpn-pbb-ldp-ext-00. These extensions will avoid network blackhole issues, as described in the Section 3 of the mentioned draft.

The objective of this section is to provide the required guidelines to configure and troubleshoot a PBB-VPLS network.

Knowledge of the VPLS and H-VPLS (RFC 4762, *Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling*) architecture and functionality is assumed throughout this document. The most relevant concepts will be briefly explained throughout the document, taking the network setup shown in the next section as an example. For further information, refer to the relevant Alcatel-Lucent documentation.

Overview

The following network setup will be used throughout the rest of the chapter.

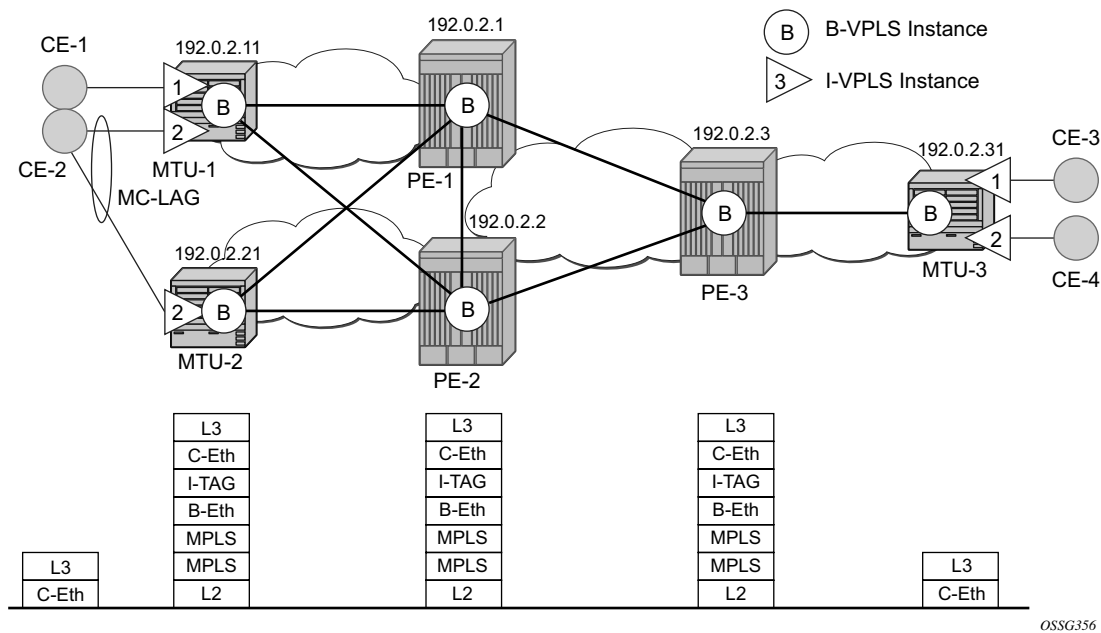


Figure 189: Network Topology

The setup consists of three core nodes (PE-1, PE-2 and PE-3) and three MTU (Multi-Tenant Unit) nodes connected to the core. A backbone VPLS instance (B-VPLS 100) will be defined in all the six nodes, whereas a few customer I-VPLS instances will be defined on the three MTU nodes.

Those I-VPLS instances will be multiplexed into the common B-VPLS, using the ISID field within the I-TAG as the demultiplexer field at the egress MTU to differentiate each specific customer.

The B-VPLS domain constitutes an H-VPLS network itself, with spoke SDPs from the MTUs to the core PE layer. Active/standby spoke SDPs can be used from the MTUs to the PEs (for example, in the MTU-1 and MTU-2 cases) or single non-redundant spoke SDPs (for example, MTU-3). CE-2 is dual-connected to the service provider network through MC-LAG.

The protocol stack being used along the path between the CEs is represented in [Figure 189](#).

Configuration

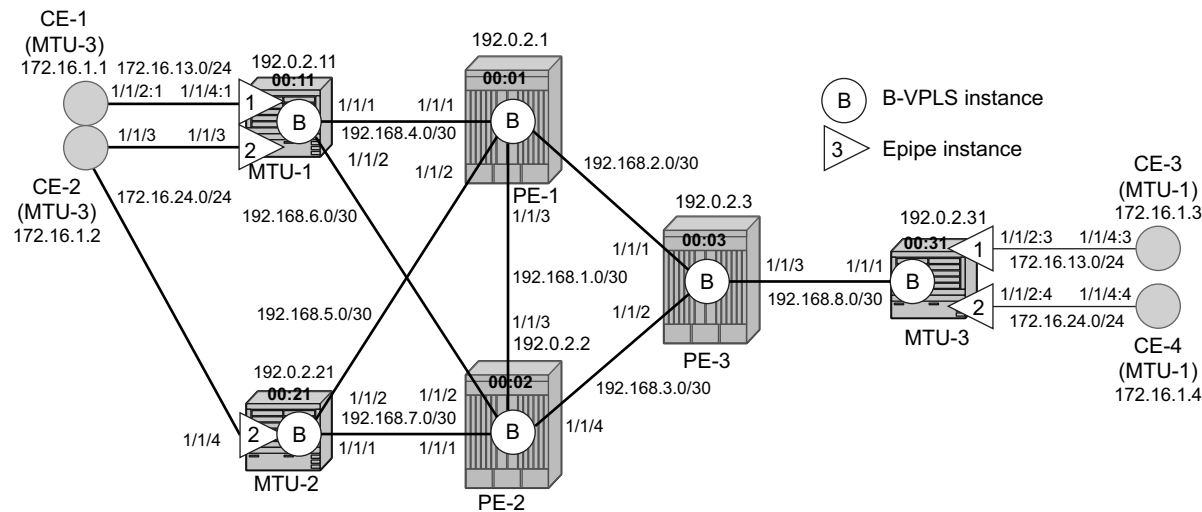
This section describes all the relevant PBB-VPLS configuration tasks for the setup shown in [Figure 189](#). Note that the appropriate associated IP/MPLS configuration is out of the scope of this example. In this particular example, the following protocols will be configured beforehand:

- ISIS-TE as IGP with all the interfaces being Level-2 (OSPF-TE could have been used instead).
- RSVP-TE as the MPLS protocol to signal the transport tunnels (LDP could have been used instead, but without fast re-route).
- LSPs between core PEs will be fast re-route protected (facility bypass tunnels) whereas LSP tunnels between MTUs and PEs will not be protected.
- The protection between MTU-1, MTU-2 and PE-1, PE-2 will be based on the A/S pseudowire protection configured in the B-VPLS.
- BGP is configured for auto-discovery (Layer 2-VPN family), since FEC 129 will be used for the pseudowires between PEs in the core.

Once the IP/MPLS infrastructure is up and running, the service configuration tasks described in the following sections can be implemented.

PBB-VPLS M:1 Service Configuration

This section explains the process to configure PBB-VPLS services in a M:1 fashion, **M** being the number of customer I-VPLS services multiplexed into the same B-VPLS instance (instance 100). An alternative configuration is 1:1, where each customer I-VPLS has its own B-VPLS. MTU-1 and PE-1 will be picked to show the relevant CLI configuration commands. Note that the bold digits separated by colons **00:xx** are abbreviations for the backbone MAC addresses.



OSSG358

Figure 190: MTU-1 and PE-1 Nodes as Configuration Examples

B-VPLS Configuration

The first step is to configure the B-VPLS instance that will carry the PBB traffic. The following shows the B-VPLS configuration.

Configuration examples:

```
# for MTU-1
configure
service
    vpls 100 customer 1 b-vpls create
        service-mtu 2000
        pbb
            source-bmac 00:11:11:11:11:11
        exit
        stp
            shutdown
        exit
        endpoint "core" create
            no suppress-standby-signaling
        exit
        spoke-sdp 111:100 endpoint "core" create
            stp
                shutdown
            exit
            precedence primary
            no shutdown
        exit
        spoke-sdp 112:100 endpoint "core" create
            stp
                shutdown
            exit
            no shutdown
        exit
        no shutdown
    exit

# for PE-1
configure
service
    pw-template 1 use-provisioned-sdp create
        split-horizon-group "CORE"
    exit
exit
vpls 100 customer 1 b-vpls create
    service-mtu 2000
    pbb
        source-bmac 00:01:01:01:01:01
    exit
    bgp
        route-target export target:65000:100 import target:65000:100
        pw-template-binding 1
    exit
exit
bgp-ad
    vpls-id 65000:100
    no shutdown
```

PBB-VPLS M:1 Service Configuration

```
exit
stp
    shutdown
exit
spoke-sdp 111:100 create
    no shutdown
exit
spoke-sdp 121:100 create
    no shutdown
exit
no shutdown
exit
```

The relevant B-VPLS commands are in **bold**.

Note that the keyword **b-vpls** is given at creation time and therefore it cannot be added to a regular existing VPLS instance. Besides the **b-vpls** keyword, the B-VPLS is a regular VPLS instance in terms of configuration, with the following exceptions:

- The B-VPLS service MTU must be at least 18 bytes greater than the I-VPLS MTU of the multiplexed instances. In this example, the I-VPLS instances will have the default service MTU (1500 bytes); hence, any MTU equal to or greater than 1518 bytes must be configured. In this particular example, a MTU of 2000 bytes is configured in the B-VPLS instance throughout the network.
- The source B-MAC is the MAC that will be sourced when the PBB traffic is originated from that node. Note that you can configure a source B-MAC per B-VPLS instance (if there are more than one B-VPLS) or a common source B-MAC that will be shared by all the B-VPLS instances in the box. If no specific source B-MAC is provisioned, the system MAC address is used as the source B-MAC. Note that when using the Access Multi-Homing feature for Native PBB, the source-bmac must be a configured one and never the chassis mac address. The way to configure a common B-MAC for all the B-VPLS instances on MTU-1 is shown below:

```
configure
    service
        pbb
            source-bmac 00:11:11:11:11:11
```

The following considerations will be taken into account when configuring the B-VPLS:

- B-VPLS SAPs:
 - Ethernet dot1q and null encapsulations are supported
 - Default SAP (:*) types are blocked in the CLI for the B-VPLS SAP

- B-VPLS SDPs:
 - For MPLS, both mesh and spoke SDPs with split horizon groups are supported.
 - Similar to regular pseudowires, the outgoing PBB frame on an SDP (for example, B-pseudowire) contains a BVID qtag only if the pseudowire type is Ethernet VLAN. If the pseudowire type is **Ethernet**, the BVID qtag is stripped before the frame goes out.
 - BGP-AD is supported in the B-VPLS; therefore, spoke SDPs in the B-VPLS can be signaled using FEC 128 or FEC 129. In this example, BGP-AD and FEC 129 are used. A split-horizon group has been configured to emulate the behavior of mesh-SDPs in the core.
- If a local I-VPLS instance is associated with the B-VPLS, “local frames” originated/terminated on local I-VPLS(s) are PBB encapsulated/de-encapsulated using the PBB etype provisioned under the related port or SDP component.

By default, the PBB etype is 0x88e7 (which is the standard one defined in 802.1ah for the I-TAG) but this PBB etype can be changed if required due to interoperability reasons. This is the way to change it at port and/or SDP level:

```
A:MTU-1# configure port 1/1/1 ethernet pbb-etype
- pbb-etype <0x0600..0xffff>
- no pbb-etype

<0x0600..0xffff>      : [1536..65535] - accepts in decimal or hex

A:MTU-1# configure service sdp 111 pbb-etype
- no pbb-etype [<0x0600..0xffff>]
- pbb-etype <0x0600..0xffff>

<0x0600..0xffff>      : [1536..65535] - accepts in decimal or hex
```

The following commands are useful to check the actual PBB etype.

```
A:MTU-1# show service sdp 111 detail | match PBB
Bw BookingFactor      : 100                PBB Etype      : 0x88e7

A:MTU-1# show port 1/1/1 | match PBB
PBB Ethertype         : 0x88e7
```

I-VPLS Configuration

Once the common B-VPLS is configured, the next step is to provision the customer I-VPLS instances. The following shows the relevant configuration on MTU-1 for the two I-VPLS instances represented in [Figure 190](#). The I-VPLS instances are configured on the MTU devices, whereas the core PE's are customer-unaware nodes.

```
configure
service
  vpls 1 customer 1 i-vpls create
    pbb
      backbone-vpls 100
    exit
  exit
  stp
    shutdown
  exit
  sap 1/1/4:1 create
  exit
  no shutdown
exit
  vpls 2 customer 1 i-vpls create
    pbb
      backbone-vpls 100 isid 2
    exit
  exit
  stp
    shutdown
  exit
  sap lag-1 create
  exit
  no shutdown
exit
```

The relevant I-VPLS commands are in **bold**.

Note that the keyword **i-vpls** is given at creation time and therefore it cannot be added to a regular existing VPLS instance. After creating the I-VPLS instance, it has to be linked to its corresponding transport B-VPLS instance. That link is given by the **backbone-vpls b-vpls isid** isid command. If no ISID (20 bit customer identification in the ITAG) is specified, the system will take the VPLS instance identifier as the ISID value.

The following considerations will be taken into account when configuring the I-VPLS:

- I-VPLS SAPs:
 - SAPs can be defined on ports with any Ethernet encapsulation type (null, dot1q and qinq)
 - The I-VPLS SAPs can coexist on the same port with SAPs for other business services, for example, VLL and VPLS SAPs.

- I-VPLS SDPs:
 - GRE and MPLS SDPs are supported.
 - No mesh-SDPs are supported, only spoke SDP. Mesh-SDPs can be emulated by using split horizon groups.

Existing SAP processing rules still apply for the I-VPLS case; the SAP encapsulation definition on Ethernet ingress ports defines which VLAN tags are used to determine the service that the packet belongs to:

- Null encapsulation defined on ingress — Any VLAN tags are ignored and the packet goes to a default service for the SAP;
- Dot1q encapsulation defined on ingress — only first VLAN tag is considered;
- QinQ encapsulation defined on ingress — both VLAN tags are considered; wildcard for the inner VLAN tag is supported.
- For dot1q/qinq encapsulations, traffic encapsulated with VLAN tags for which there is no definition is discarded.
- Note that any VLAN tag used for service selection on the I-SAP is stripped before the PBB encapsulation is added. Appropriate VLAN tags are added at the remote PBB PE when sending the packet out on the egress SAP.

Up to 8000 VPLS instances can be defined per system. That number includes I-VPLS, B-VPLS and regular VPLS.

MMRP for Flooding Optimization

When the M:1 model is used (as in this example), any I-VPLS broadcast/multicast/unknown frame is flooded throughout the B-VPLS domain regardless of the nodes where the originating I-VPLS is defined. In other words, in our example in [Figure 189](#), any broadcast/multicast/unknown frame coming from CE-1 would be flooded in the B domain and would reach PE-2 and MTU-2, even though that traffic only needs to go to PE-3 and MTU-3. In order to build customer-based flooding trees and optimize the flooding, MMRP (Multiple MAC Registration Protocol) must be configured on the B-VPLS.

MMRP can be enabled with its default settings just by executing a **mrp no shutdown** command:

```
service
  vpls 100 customer 1 b-vpls create
    service-mtu 2000
    pbb
      source-bmac 00:01:01:01:01:01
    exit
    bgp
      route-target export target:65000:100 import target:65000:100
      pw-template-bind 1
    exit
    bgp-ad
      vpls-id 65000:100
      no shutdown
    exit
    stp
      shutdown
    exit
    mrp
      no shutdown
    exit
    spoke-sdp 111:100 create
    exit
    spoke-sdp 121:100 create
    exit
    no shutdown
  exit
```

There are certain B-VPLS MRP settings that can be modified. These are the default values:

```
*A:MTU-1>config>service>vpls>mrp# info detail
-----
      mmrp
        no end-station-only
        attribute-table-size 2048
        attribute-table-low-wmark 90
        attribute-table-high-wmark 95
        no flood-time
        no shutdown
      exit
      no shutdown
-----
*A:MTU-1>config>service>vpls>mrp#
```

These attributes can be changed in order to control the number of MMRP attributes per B-VPLS and optimize the convergence time in case of failures in the B-VPLS:

- Controlling the number of attributes per B-VPLS

The MMRP exchanges create one entry per attribute (group B-MAC) in the B-VPLS where MMRP protocol is running. When the first registration is received for an attribute, an MFIB entry is created for it. The *attribute-table-size* allows the user to control the number of MMRP attributes (group B-MACs) created on a per B-VPLS basis, between 1 and 2048. Based on the configured size, high and low watermarks can be set (in percentage) so that alarms can be triggered upon exceeding the watermarks. This ensures that no B-VPLS will take up all the resources from the total pool. The maximum number of attributes per B-VPLS is 2048 and 4000 can be configured globally on the system.

- Optimizing the convergence time

Assuming that MMRP is used in a certain B-VPLS, under failure conditions the time it takes for the B-VPLS forwarding to resume may depend on the data plane and control plane convergence plus the time it takes for MMRP exchanges to stabilize the flooding trees on a per ISID basis. In order to minimize the convergence time, the PBB SR OS implementation offers the selection of a mode where B-VPLS forwarding reverts for a short time to flooding so that MMRP has enough time to converge. This mode can be selected through configuration using the **flood-time value** command where value represents the amount of time in seconds (between 3 and 600) that flooding will be enabled. If this behavior is selected, the forwarding plane starts with B-VPLS flooding for a configurable time period, then it reverts back to the MFIB entries installed by MMRP. The following B-VPLS events initiate the switch from per I-VPLS (MMRP) MFIB entries to BVPLS flooding:

- Reception or local triggering of a TCN (Spanning Tree Topology Change Notification)
- B-SAP failure
- Failure of a B-SDP binding
- Pseudowire activation in a primary/standby H-VPLS resiliency solution
- SF/CPM switchover due to STP reconvergence

The IEEE 802.1ak standard, which defines MRP, requires the implementation of different state machines with associated timers that can be tuned. A full MRP participant maintains the following state machines:

- Registrar state machine
- Applicant state machine
- LeaveAll state machine
- PeriodicTransmission state machine

The two first state machines are maintained for each attribute in which the participant is interested, while the two latter are global to all the attributes.

The job of the registrar function is to record declarations of the attribute made by other participants on the LAN. A registrar does not send any protocol messages, as the applicant looks after the interests of all would-be participants.

The job of the applicant is twofold: first, to ensure that this participant's declaration is correctly registered by other participants' registrars, and next, to prompt other participants to register again after one withdraws a declaration.

The associated timers can be tuned on a per SAP/SDP basis:

```
A:MTU-1>config>service>vpls>spoke-sdp# mrp
- mrp

[no] join-time          - Configure timer value in 10th of seconds for sending
                        join-messages
[no] leave-all-time    - Configure timer value in 10th of seconds for refreshing
                        all attributes
[no] leave-time         - Configure timer value in 10th of seconds to hold
                        attribute in leave-state
[no] mrp-policy         - Configure mrp-policy
[no] periodic-time      - Configure timer value in 10th of seconds for
                        re-transmission of attribute declarations
[no] periodic-timer     - Control re-transmission of attribute declarations

A:MTU-1>config>service>vpls>spoke-sdp#

A:MTU-1>config>service>vpls>spoke-sdp>mrp# info detail
-----
                        join-time 2
                        leave-time 30
                        leave-all-time 100
                        periodic-time 10
                        no periodic-timer
                        no mrp-policy
-----

A:MTU-1>config>service>vpls>spoke-sdp>mrp#
```

A brief description of the MRP SAP/SDP attributes follows:

- **Join-time** — This command controls the interval between transmit opportunities that are applied to the applicant state machine. An instance of this join period timer is required on a per-port, per-MRP participant basis. For additional information, refer to IEEE 802.1ak-2007 section 10.7.4.1.
- **Leave-time** — This command controls the period of time that the registrar state machine will wait in the leave state before transitioning to the MT state when it is removed. An instance of the timer is required for each state machine that is in the leave state. The leave period timer is set to the value leave-time when it is started. A registration is normally in

“in” state where there is an MFIB entry and traffic being forwarded. When a “leave all” is performed (periodically around every 10-15 seconds per SAP/SDP binding – see leave-all-time-below), a node sends a message to its peer indicating a leave all is occurring and puts all of its registrations in leave state. The peer refreshes its registrations based on the leave all PDU it receives and sends a PDU back to the originating node with the state of all its declarations. Refer to IEEE 802.1ak-2007 section 10.7.4.2.

- **Leave-all-time** — This command controls the frequency with which the leaveall state machine generates leaveall PDUs. The timer is required on a per-port, per-MRP participant basis. The leaveall period timer is set to a random value, T , in the range $\text{leavealltime} < T < 1.5 * \text{leave-all-time}$ when it is started. Refer to IEEE 802.1ak-2007, section 10.7.4.3.
- **Periodic-time** — This command controls the frequency the periodic transmission state machine generates periodic events if the periodic transmission timer is enabled. The timer is required on a per-port basis. The periodic transmission timer is set to one second when it is started.
- **Periodic-timer** — This command enables or disables the periodic transmission timer.

The following command shows the MRP configuration and statistics on a per SAP/SDP basis within the B-VPLS:

```
A:MTU-1# show service id 100 all | match post-lines 10 MRP
```

```
Sdp Id 111:100 MRP Information
```

```
-----
Join Time           : 0.2 secs           Leave Time          : 3.0 secs
Leave All Time       : 10.0 secs          Periodic Time        : 1.0 secs
Periodic Enabled    : false
Mrp Policy          : N/A
Rx Pdus             : 2542               Tx Pdus             : 2963
Dropped Pdus        : 0
Rx New Event        : 0                  Rx Join-In Event    : 1986
Rx In Event         : 0                  Rx Join Empty Evt   : 1533543
Rx Empty Event      : 3                  Rx Leave Event      : 126
SDP MMRP Information
-----
```

```
MAC Address      Registered      Declared
-----
```

```
01:1e:83:00:00:01 Yes           Yes
-----
```

```
01:1e:83:00:00:02 Yes           Yes
-----
```

```
Number of MACs=2 Registered=2 Declared=2
-----
```

```
Sdp Id 112:100 MRP Information
```

```
-----
Join Time           : 0.2 secs           Leave Time          : 3.0 secs
Leave All Time       : 10.0 secs          Periodic Time        : 1.0 secs
Periodic Enabled    : false
Mrp Policy          : N/A
Rx Pdus             : 0                  Tx Pdus             : 0
Dropped Pdus        : 0
Rx New Event        : 0                  Rx Join-In Event    : 0
Rx In Event         : 0                  Rx Join Empty Evt   : 0
Rx Empty Event      : 0
-----
```

PBB-VPLS M:1 Service Configuration

```

Rx Empty Event      : 0
SDP MMRP Information
-----
MAC Address          Registered      Declared
-----
Number of MACs=0 Registered=0 Declared=0
-----
Number of SDPs : 2
-----
* indicates that the corresponding row element may have been truncated.
Service MRP Information
=====
Admin State          : enabled
-----
MMRP
-----
Admin Status         : enabled          Oper Status         : up
Register Attr Cnt    : 2                Declared Attr Cnt: 2
End-station-only     : disabled
Max Attributes       : 2048             Attribute Count    : 2
Hi Watermark         : 95%              Low Watermark      : 90%
Failed Registers     : 0                Flood Time         : Off
-----
MVRP
-----
MRP SAP Table
=====
SAP                  Join      Leave      Leave All Periodic
                   Time(sec) Time(sec) Time(sec) Time(sec)
-----
=====
MRP SDP-BIND Table
=====
SDP-BIND             Join      Leave      Leave All Periodic
                   Time(sec) Time(sec) Time(sec) Time(sec)
-----
111:100              0.2      3.0      10.0      1.0
112:100              0.2      3.0      10.0      1.0
-----
=====
A:MTU-1#

```

The following command is useful to check the MRP configuration and status.

```

A:MTU-1# show service id 100 mrp
=====
Service MRP Information
=====
Admin State          : enabled
-----
MMRP
-----
Admin Status         : enabled          Oper Status         : up
Register Attr Cnt    : 2                Declared Attr Cnt: 2

```



```

End-station-only      : disabled
Max Attributes         : 2048
Hi Watermark          : 95%
Failed Registers       : 0
Attribute Count        : 2
Low Watermark          : 90%
Flood Time             : Off

```

MVRP

```

Admin Status          : disabled
Max Attr              : 4095
Register Attr Count    : 0
Hi Watermark          : 95%
Hold Time             : disabled
Oper Status           : down
Failed Register        : 0
Declared Attr         : 0
Low Watermark          : 90%
Attr Count            : 0

```

=====

MRP SAP Table

=====

SAP	Join Time(sec)	Leave Time(sec)	Leave All Time(sec)	Periodic Time(sec)

=====				

=====

MRP SDP-BIND Table

=====

SDP-BIND	Join Time(sec)	Leave Time(sec)	Leave All Time(sec)	Periodic Time(sec)

111:100	0.2	3.0	10.0	1.0
112:100	0.2	3.0	10.0	1.0
=====				

=====

A:MTU-1#

In the example throughout the document, as soon as MMRP is enabled, an optimized flooding tree will be built for ISID 1, since the I-VPLS 1 is only defined in MTU-1 and MTU-3, but not in MTU-2. A good way to track the flooding tree for a particular ISID is the following command:

*A:MTU-1# show service id 100 mmrp mac

SAP/SDP	MAC Address	Registered	Declared

sdp:111:100	01:1e:83:00:00:01	Yes	Yes
sdp:111:100	01:1e:83:00:00:02	Yes	Yes

Number of Entries=2 SAPs=0 SDPs=2

*A:MTU-1#

*A:MTU-2# show service id 100 mmrp mac

SAP/SDP	MAC Address	Registered	Declared

sdp:212:100	01:1e:83:00:00:01	Yes	No
sdp:212:100	01:1e:83:00:00:02	Yes	No

Number of Entries=2 SAPs=0 SDPs=2

*A:MTU-2#

The group B-MAC ending in **01** corresponds to the I-VPLS 1 whereas the one ending in **02** to the I-VPLS 2. Note that MMRP PDUs for the two attributes are sent throughout the loop-tree topology (not over STP blocked ports or standby spoke SDPs and observing the split horizon rules). The two attributes are registered on every B-VPLS virtual port; however, the tree is only built on those ports where the attribute is also declared, and not only registered. For instance, the spoke-sdp 212:100 in MTU-2 will not be part of the ISID 1 or ISID 2 flooding trees. Neither attribute is declared since: I-VPLS 1 doesn't exist on MTU-2 and I-VPLS 2 is operationally down on MTU-2 (MC-LAG SAP is in standby state hence the I-VPLS down).

Note that as soon as a group B-MAC attribute is registered on a particular port, an MFIB entry is added for that B-MAC on that port, regardless of the declaration state for that attribute on the port. For instance, neither B-MAC is declared on MTU-2 however the two MFIB entries are created as soon as the attributes are registered:

```
A:MTU-2# show service id 100 mfib
=====
Multicast FIB, Service 100
=====
Source Address  Group Address      Sap/Sdp Id          Svc Id  Fwd/Blk
-----
*               01:1E:83:00:00:01  b-sdp:212:100      Local   Fwd
*               01:1E:83:00:00:02  b-sdp:212:100      Local   Fwd
-----
Number of entries: 2
=====
A:MTU-2#
```

MAC Flush: Avoiding Blackholes

Both the I-VPLS and B-VPLS components inherit the MAC flush capabilities of a regular VPLS clearing the related C-MAC and respectively B-MAC FIBs. All types of MAC flush – flush-all-but-mine and flush-all-from-me – are supported together with the related CLI. In addition to these features, some extensions have been added so that MAC flush can be triggered on the B-VPLS based on some events happening on the I-VPLS. The following diagram shows a potential scenario where blackholes can occur if the proper configuration is not added.

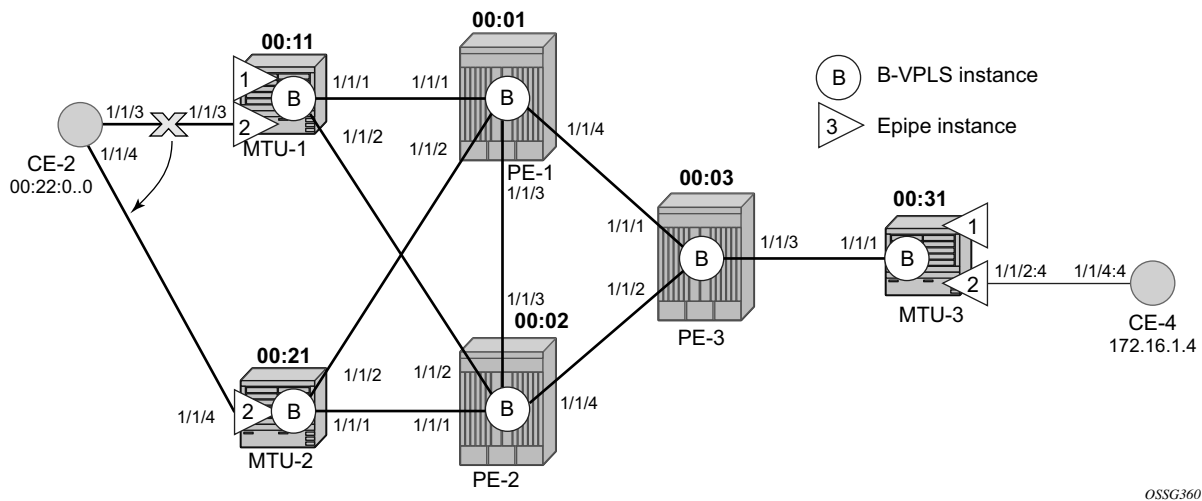


Figure 191: Blackhole

Under normal conditions the I-VPLS 2 FIB on MTU-3 shows that CE-2 MAC address is learned through B-MAC 00:11 (MTU-1's B-MAC):

```
A:MTU-3# show service id 2 fdb pbb
=====
Forwarding Database, i-Vpls Service 2
=====
MAC                Source-Identifier    B-Svc    b-Vpls MAC        Type/Age
-----
00:22:00:00:00:00  b-sdp:33:100        100      00:11:11:11:11:11  L/0
00:44:00:00:00:00  sap:1/1/2:4         100      N/A                L/0
=====
A:MTU-3#
```

When a failure happens in the CE-2 MC-LAG active link, the link to MTU-2 takes over. However, the FIB on MTU-3 still points at MTU-1's B-MAC and that will still be the B-MAC used in the PBB encapsulation. Therefore a blackhole occurs until either bidirectional traffic is sent or the FIB aging timer expires.

PBB-VPLS M:1 Service Configuration

The following command will solve the blackhole:

```
# on MTU-1
configure
  service
    vpls 2
      pbb
        backbone-vpls 100 isid 2
        exit
        send-bvpls-flush all-from-me
      exit
    stp
      shutdown
    exit
  sap lag-1 create
  exit
no shutdown

*A:MTU-1# configure service vpls 2 pbb
*A:MTU-1>config>service>vpls>pbb# send-bvpls-flush
  - send-bvpls-flush { [all-but-mine] [all-from-me] }
  - no send-bvpls-flush

<all-but-mine>      : keyword
<all-from-me>       : keyword
```

By enabling **send-bvpls-flush all-from-me** on I-VPLS 2, a failure on the MC-LAG active link on I-VPLS 2 will trigger an LDP MAC flush **flush-all-from-me** into the B-VPLS that will flush the FIB in MTU-3 for I-VPLS 2, avoiding the blackhole. A MC-LAG failure is emulated below:

```
*A:MTU-1# configure lag 1 shutdown

# on MTU-1

1 2015/03/11 14:56:47.07 UTC MINOR: DEBUG #2001 Base LDP
"LDP: LDP
Send Address Withdraw packet (msgId 208) to 192.0.2.1:0
Protocol version = 1
MAC Flush (All MACs learned from me)
Service FEC PWE3: ENET(5)/100 Group ID = 0 cBit = 0
Number of PBB-BMACs = 1
BMAC 1 = 00:11:11:11:11:11
Number of PBB-ISIDs = 1
ISID 1 = 2
Number of Path Vectors : 1
Path Vector( 1) = 192.0.2.11
"

# on MTU-3

1 2015/03/11 14:56:48.13 UTC MINOR: DEBUG #2001 Base LDP
"LDP: LDP
Recv Address Withdraw packet (msgId 206) from 192.0.2.3:0
Protocol version = 1
MAC Flush (All MACs learned from me)
Service FEC PWE3: ENET(5)/100 Group ID = 0 cBit = 0
```

```

Number of PBB-BMACs = 1
BMAC 1 = 00:11:11:11:11:11
Number of PBB-ISIDs = 1
ISID 1 = 2
Number of Path Vectors : 3
Path Vector( 1) = 192.0.2.11
Path Vector( 2) = 192.0.2.1
Path Vector( 3) = 192.0.2.3

```

Immediately after receiving the MAC flush, the CE-2 MAC is flushed and re-learned but this time linked to the B-MAC 00:21, which is the MTU-2 B-MAC.

```

A:MTU-3# show service id 2 fdb pbb
=====
Forwarding Database, i-Vpls Service 2
=====
MAC                Source-Identifier      B-Svc    b-Vpls MAC        Type/Age
-----
00:44:00:00:00:00  sap:1/1/2:4          100      N/A                L/30
=====
A:MTU-3#
*A:MTU-3# show service id 2 fdb pbb
=====
Forwarding Database, i-Vpls Service 2
=====
MAC                Source-Identifier      B-Svc    b-Vpls MAC        Type/Age
-----
00:22:00:00:00:00  b-sdp:33:100         100      00:21:21:21:21:21 L/0
00:44:00:00:00:00  sap:1/1/2:4          100      N/A                L/0
=====
*A:MTU-3#

```

The following I-VPLS events are propagated into the B-VPLS depending on the **flush-all-but-mine** or **flush-all-from-me** keywords used in the configuration:

If the **flush-all-but-mine** keyword is configured (positive flush), the following events in the I-VPLS trigger a MAC flush into the B-VPLS:

1. TCN event in one or more of the related I-VPLS/M-VPLS.
2. Pseudowire/SDP binding activation with active/standby pseudowire (standby to active or down to up).
3. Reception of an LDP MAC withdraw flush-all-but-mine in the related I-VPLS.

If the **flush-all-from-me** keyword is configured (negative flush) the following events in the I-VPLS trigger a MAC flush into the B-VPLS:

1. MC-LAG active link failure (in our example).
2. Failure of a local SAP – requires **send-flush-on-failure** to be enabled in I-VPLS.
3. Failure of a local pseudowire/SDP binding – requires **send-flush-on-failure** to be enabled in I-VPLS.

4. Reception of an LDP MAC withdraws flush-all-from-me in the related I-VPLS.

In addition to this and regardless of what type, MAC flush has been optimized to avoid flushing in the core PEs, flushing only the C-MACs mapped to a certain B-MAC (belonging to a specific ISID FIB) and the ability to indicate to core PEs which messages should always be forwarded endpoint-to-endpoint towards all PBB PEs regardless of the propagate-mac-flush setting in B-VPLS. All of this is implemented without the need of any additional CLI commands and it is part of draft-balus-l2vpn-pbb-ldp-ext-00.

Another extension supported to avoid blackholes within this mix of I- and B-VPLS environments is the block-on-mesh-failure feature in PBB. When the VPLS mesh exists only in I-VPLS or in B-VPLS, and the block-on-mesh-failure feature is enabled, the regular VPLS behavior will apply (when all the mesh SDPs go down an LDP notification with pseudowire status bits = 0x01 - Pseudo Wire Not Forwarding – is sent over the spoke SDPs). When the active/standby pseudowire resiliency is implemented in I-VPLS such that the PBB PE performs the role of a PE-rs, the B-VPLS core replaces the pseudowire (SDP binding) mesh. The block-on-mesh-notification (LDP notification indicating pseudowire not forwarding) will be sent to the MTUs only when the related B-VPLS is operationally down. The B-VPLS core is operationally down only when all of its SAPs and SDPs are down.

The final feature that can be enabled in an I-VPLS with CLI is the send-flush-on-bvpls-failure feature.

```
A:PE-1>config>service>vpls>pbb# send-flush-on-bvpls-failure
- no send-flush-on-bvpls-failure
- send-flush-on-bvpls-failure
```

This feature is required to avoid blackholes when there is a full-mesh of pseudowires in the I-VPLS domain and the B-VPLS instance can go operationally down. The following figure shows a typical scenario where this feature is needed (normally when PBB-VPLS and multi-chassis end point are combined together).

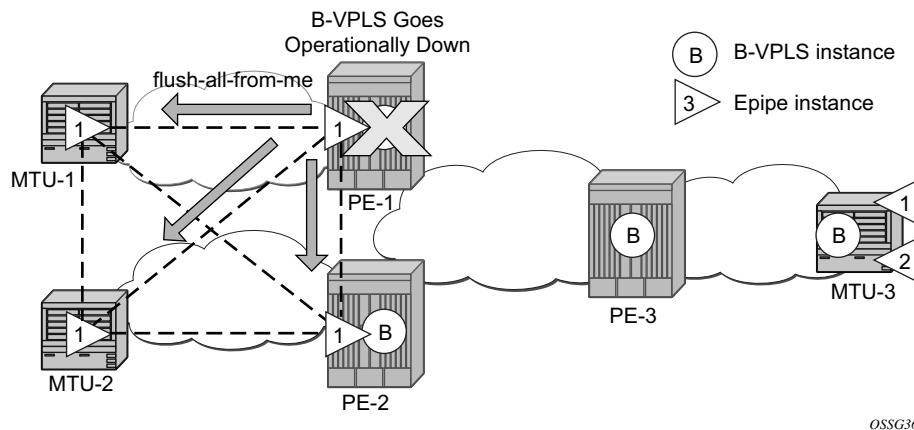


Figure 192: Send Flush on BVPLS Failure Example

Access Dual-Homing and MAC Notification

Although this section is focused on PBB in a MPLS based network, Alcatel-Lucent PBB implementation also allows the operator to use a native Ethernet infrastructure in the PBB core. Native Ethernet tunneling can be emulated using Ethernet SAPs to interconnect the related B-VPLS instances. In those cases, there is no LDP signaling available; hence, there is no MAC flush sent when the active link in a multi-homed access device fails.

The SR OS supports a mechanism to avoid potential blackholes in native Ethernet PBB networks. In addition to the source B-MAC associated with each B-VPLS, an additional B-MAC is associated with each MC-LAG supporting Multi-homed I-VPLS SAPs. The nodes that are in a multi-homed MC-LAG configuration share a common B-MAC on the related MC-LAG interfaces. When the **mac-notification no shutdown** command is executed, an Ethernet CFM notification message is sent from the node holding the active link. That message will be flooded in the B-VPLS domain using the MC-LAG SAP B-MAC as the source MAC address. The remote nodes will learn the customer MAC addresses behind the MC-LAG and will link them to this new SAP B-MAC. MC-LAG will keep track of the active link for each particular LAG associated to a SAP B-MAC. Should MC-LAG detect any new active link in a node, a new CFM notification message will be flooded from the new active node.

The following caveats and considerations must be taken into account:

- The MAC notification for access dual-homed devices is only supported in “network mode D”.
 - All network ports and B-SAPs must reside on slots populated with either an IOM3-XP or IMM. All other SAPs may reside on any IOM or IMM populated slot. This is enforced by the CLI.
 - MC-LAG SAPs must be on IOM3-XP/IMM.
 - Other SAPs may be on IOM1/IOM2 (non-PBB services, PBB Epipe SAP with no other SAP on local MC-LAG, PBB I-VPLS SAPs even if some of the other I-SAPs are on the MC-LAG).
 - This is automatically applicable to the 7750 SR-c4/12.
- Only MC-LAG is supported as dual-home mechanism.
- This mechanism is supported for native PBB and/or MPLS-based PBB-VPLS. Although it is mostly beneficial when native PBB is used in the core, it can also help to optimize the re-learning process in a MPLS-based core in case of MC-LAG failures, in addition to the existing LDP MAC flush procedures.

The example of this configuration shows the setup being used in this configuration example. MAC-notification will be configured in MTU-1 and MTU-2 for the dual-homed CE-2.

The first step is to configure the sap-bmac that will be used for the mac-notification messages. The source-bmac-lsb (source backbone MAC least significant bits) command has been added to the mc-lag branch so that the operator can decide the two last octets to be used in the sap-bmac. Those two last octets can be derived from the lacp-key (if the use-lacp-key statement is used) or can be specifically defined.

```
*A:MTU-1>config>redundancy>mc>peer>mc-lag# lag
- lag <lag-id> lacp-key <admin-key> system-id <system-id> [remote-lag <remote-lag-id>]
  system-priority <system-priority> source-bmac-lsb use-lacp-key
- lag <lag-id> lacp-key <admin-key> system-id <system-id> [remote-lag <remote-lag-id>]
  system-priority <system-priority> source-bmac-lsb <MAC-Lsb>
- lag <lag-id> lacp-key <admin-key> system-id <system-id> [remote-lag <remote-lag-id>]
  system-priority <system-priority>
- lag <lag-id> [remote-lag <remote-lag-id>]
- no lag <lag-id>

<lag-id>                : [1..200]
<admin-key>             : [1..65535]
<system-id>             : xx:xx:xx:xx:xx:xx      - xx [00..FF]
<remote-lag-id>         : [1..800]
<system-priority>       : [1..65535]
<MAC-Lsb>               : [1..65535] or xx-xx or xx:xx
```

```
*A:MTU-1>config>redundancy>mc>peer>mc-lag#
```

There must be a different sap-bmac per MC-LAG. The use of the lacp-key as a default for two least significant octets makes the operations simpler. In our example, the sap-bmac last two octets will come from the lacp-key:


```
*A:MTU-1>config>redundancy>mc>peer>mc-lag# info
-----
lag 1 lacp-key 15 system-id 00:00:00:00:00:01 system-priority 65535
source-bmac-lsb use-lacp-key
                        no shutdown
-----
*A:MTU-1>config>redundancy>mc>peer>mc-lag#
```

Therefore, the sap-bmac will be formed in the following way:

```
[4 first bytes of the source bmac + 2 bytes from source-bmac-lsb]
```

Finally, enable the mac-notification and instruct the b-vpls to use the sap-bmac at service level on all MTUs:

```
configure
  service
    pbb
      mac-name "MTU-1" 00:11:11:11:11:11
      mac-name "MTU-2" 00:21:21:21:21:21
      mac-name "MTU-3" 00:31:31:31:31:31
    exit
  vpls 100 customer 1 b-vpls create
    mac-notification
      no shutdown
    exit
  no shutdown
exit
```

The **mac-notification** command activates the described mechanism and has the following parameters:

```
*A:MTU-1# configure service vpls 100 mac-notification
- mac-notification

[no] count          - Configure count for MAC-notification messages
[no] interval       - Configure interval for MAC-notification messages
[no] renotify       - Configure re-notify interval for MAC-notification messages
[no] shutdown       - Configure admin state for MAC-notification messages

*A:MTU-1#
```

Where:

- interval <value> controls how often the subsequent MAC notification messages are sent. Default = 100 ms. Required values: 100 ms – 10 sec, in increments of 100 ms.
- count <value> controls how often the MAC notification messages are sent. Default = 3. Range: 1-10.

Note that the “count” and “interval” parameters can also be configured at the service context. The settings configured at the B-VPLS service context take precedence though.

PBB-VPLS M:1 Service Configuration

```
*A:MTU-1# configure service mac-notification
- mac-notification

[no] count          - Configure count for MAC-notification messages
[no] interval       - Configure interval for MAC-notification messages

*A:MTU-1#
```

The **use-sap-bmac** statement enables (on a per B-VPLS basis) the use of the source B-MAC allocated to the multi-homed SAPs (assigned to the MC-LAG) in the related I-VPLS service (could be Epipe service as well). Note that the command will fail if the value of the source-bmac assigned to the B-VPLS is the hardware (chassis) B-MAC. In other words, the source-bmac must be a configured one. The **use-sap-bmac** statement is by default off.

```
*A:MTU-1# configure service
    vpls 100
    pbb
        source-bmac 00:aa:aa:aa:aa:11
        use-sap-bmac
    exit all

*A:MTU-2# configure service
    vpls 100
    pbb
        source-bmac 00:aa:aa:aa:aa:22
        use-sap-bmac
    exit all
```

```
A:MTU-3# show service id 2 fdb pbb
=====
Forwarding Database, i-Vpls Service 2
=====
MAC                Source-Identifier    B-Svc    b-Vpls MAC        Type/Age
-----
00:22:00:00:00:00  b-sdp:33:100        100      00:aa:aa:aa:00:0f  L/30
00:44:00:00:00:00  sap:1/1/2:4         100      N/A                L/30
=====
A:MTU-3#
```

As soon as the **mac-notification no shutdown** command is executed, an Ethernet CFM notification message is sent from MTU-1, which is the node where the active MC-LAG link resides. The CFM message will have the source mac “00:aa:aa:aa:00:0f” (4 first bytes of the configured source bmac + 2 bytes from the configured source-bmac-lsb, which is 15 in hex) and will be flooded throughout the B-VPLS domain. Should the link between CE-2 and MTU-1 fail, the MC-LAG protocol will activate the redundant link and MTU-2 will immediately issue a CFM message with the shared sourced sap-bmac that will be flooded in the B-VPLS domain.

PBB and IGMP Snooping

IGMP snooping can be enabled on I-VPLS SAPs and SDPs (it cannot be enabled on B-VPLS). The 7x50 can keep track of IGMP joins received over individual B-SDPs or B-SAPs, and it starts flooding the Multicast Group (and only the multicast group) to all B-components (using the Group B-MAC for I-SID) as soon as the first IGMP join for that Multicast Group is received in one of the B-SAP/SDP components.

The first IGMP join message received over the local B-VPLS will add all the B-VPLS SAP/SDP components into the related multicast table associated with the I-VPLS context. When the querier is connected to a remote I-VPLS instance, over the B-VPLS infrastructure, its location is identified by the B-VPLS SDP/SAP on which the query was received and also by the source B-MAC address used in the PBB header for the query message, the B-MAC associated with the B-VPLS instance on the remote PBB PE.

The following excerpt shows an I-VPLS with IGMP snooping enabled and some static groups added on a SAP. Note that we are also configuring the location of the querier by adding the B-MAC where the querier is connected to (in this example MTU-3) and adding the two B-VPLS spoke-sdps as mrouter ports (note that the B-VPLS mrouter ports are added under the I-VPLS backbone-vpls context).

The **mac-name** command translates MACs into strings so that the names can be used instead of typing the entire MAC address every time needed.

```
# on MTU-1
configure
  service
    pbb
      mac-name "MTU-1" 00:11:11:11:11:11
      mac-name "MTU-2" 00:21:21:21:21:21
      mac-name "MTU-3" 00:31:31:31:31:31
    exit
  vpls 1 customer 1 i-vpls create
    send-flush-on-failure
    pbb
      backbone-vpls 100
        igmp-snooping
          mrouter-dest "MTU-3"
        exit
      sdp 111:100
        igmp-snooping
          mrouter-port
        exit
      exit
      sdp 112:100
        igmp-snooping
          mrouter-port
        exit
      exit
    exit
  send-flush-on-bvpls-failure
  send-bvpls-flush all-from-me
```

PBB-VPLS M:1 Service Configuration

```
exit
stp
  shutdown
exit
igmp-snooping
  no shutdown
exit
sap 1/1/4:1 create
  igmp-snooping
    static
      group 228.0.0.1
        starg
      exit
      group 228.0.0.2
        starg
      exit
      group 239.0.0.1
        source 172.16.99.99
      exit
    exit
  exit
exit
no shutdown

---snipped---
```

As in regular VPLS instances, mrouter ports are added to all the multicast groups:

```
*A:MTU-1# show service id 1 mfib
=====
Multicast FIB, Service 1
=====
Source Address  Group Address      Sap/Sdp Id          Svc Id  Fwd/Blk
-----
*               *               b-sdp:111:100       100     Fwd
               *               b-sdp:112:100       100     Fwd
*               228.0.0.1      sap:1/1/4:1         Local   Fwd
               *               b-sdp:111:100       100     Fwd
               *               b-sdp:112:100       100     Fwd
*               228.0.0.2      sap:1/1/4:1         Local   Fwd
               *               b-sdp:111:100       100     Fwd
               *               b-sdp:112:100       100     Fwd
172.16.99.99    239.0.0.1          sap:1/1/4:1         Local   Fwd
               *               b-sdp:111:100       100     Fwd
               *               b-sdp:112:100       100     Fwd
-----
Number of entries: 4
=====
*A:MTU-1#
```

Note that when the “show service id x mfib” command is issued in a B-VPLS, the group B-MAC entries are shown, whereas when the same command is issued in a I-VPLS, the IGMP (S,G) and (*,G) entries for the I and B components are shown if IGMP snooping is enabled.

MMRP Policies and ISID-Based Filtering for PBB Inter-Domain Expansion

As discussed in the [MMRP for Flooding Optimization on page 1280](#), MMRP is used in the backbone VPLS instances to build per I-VPLS flooding trees. Each I-VPLS has an associated group B-MAC in the B-VPLS, which is derived from the ISID, and is advertised by MMRP throughout the whole B-VPLS context, regardless of whether a certain I-VPLS is present in one or all the B-VPLS PEs.

In an inter-domain environment, the same B-VPLS can be defined in different domains and as such MMRP will advertise all the group B-MACs in every domain, wasting resources in all the PEs no matter if a particular ISID, and hence its group B-MAC, are not required in one of the domains. When MMRP is enabled in a particular PE, data plane and control plane resources are consumed and they must be taken into consideration when designing PBB-VPLS networks:

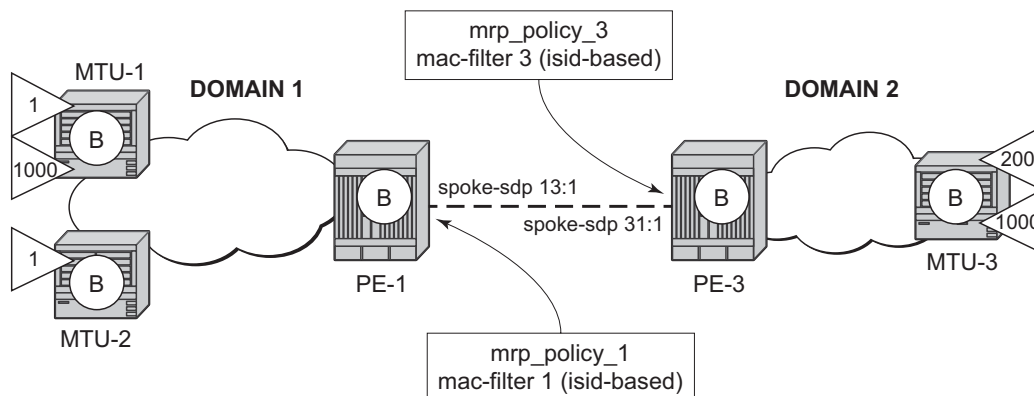
- Control plane – MMRP processing takes CPU cycles and the number of attributes that can be advertised is not unlimited
- Data plane – each group B-MAC registration takes one MFIB entry (the MFIB is shared between MMRP and IGMP/PIM snooping)

The 7x50 supports MMRP policies and ISID-based filters so that control plane and data plane resources can be saved when I-VPLS instances are not defined in all the domains.

The following figure illustrates an example of usage for MMRP policies and ISID-based filters that will be configured in this section. “Domain 1” and “domain 2” will have a range of local ISIDs each and a range of “inter-domain” ISIDs:

- Domain 1 local ISIDs: from 1 to 100
- Domain 2 local ISIDs: from 101 to 200
- Inter-domain ISIDs: from 1000 to 2000

By applying the MMRP policies indicated in the figure, domain 1 attributes will be prevented from being declared and registered in domain 2 and vice versa, domain 2 attributes from being declared and registered in domain 1. The egress mac-filters will drop any traffic sourced from a local ISID preventing it to be transmitted to the remote domain.



OSSG667

Figure 193: Inter-Domain B-VPLS and MMRP Policies/ISID-Based Filters Example

MMRP Policies

The following shows the MMRP policy configuration in node PE-1. This policy will block any registration/declaration except those for ISIDs 1000-2000. Note that packets will be compared against the configured matching ISIDs as long as the pbb-etype matches the one configured on the port or SDP.

```
# on PE-1
configure
service
mrp
  mrp-policy "mrp_policy_1" create
  description "allow-inter-domain-isids"
  default-action block
  entry 10 create
  action allow
  match
    isid 1000 to 2000
  exit
exit
exit
exit
exit
```

Once the MMRP policy is configured, it must be applied on the corresponding SAP or sdp-binding. Note that an mrp-policy can be applied to a B-VPLS SAP, B-VPLS spoke-sdp or B-VPLS mesh-sdp:

```
# on PE-1
configure
service
```

```

vpls 100 customer 1 b-vpls create
  service-mtu 2000
  pbb
    source-bmac 00:01:01:01:01:01
  exit

  ---snipped---

  mrp
    no shutdown
  exit
  spoke-sdp 111:100 create
    mrp
      mrp-policy "mrp_policy_1"
    exit
  exit
  spoke-sdp 121:100 create
    mrp
      mrp-policy "mrp_policy_1"
    exit
  exit
  no shutdown
exit

```

In the same way, mrp_policy_3 will be configured in PE-3.

Some additional considerations about the MMRP policies:

- Different entries within the same mrp-policy can have overlapping ISID ranges. The entries will be evaluated in the order of their IDs and the first match will cause the implementation to execute the associated action for that entry and then to exit the mrp-policy.
- If no ISID is specified in the match condition then:
 - If the action is “end-station”, no entry is added and the action is block.
 - If the action is different from “end-station”, every ISID is considered for that action.
- The mrp-policy specifies either a forward or a drop action for the group B-MAC attributes associated with the ISIDs specified in the match criteria.

```

*A:PE-1>config>serv>mrp>mrp-policy>entry# action
- action <action>
- no action

<action>                : none|block|allow|end-station

*A:PE-1>config>serv>mrp>mrp-policy>entry#

```

- Note that there is an additional action called **end-station**. This action specifies that an end-station emulation is present on the SAP/SDP-binding where the policy has been applied. The matching ISIDs will not get declared/registered in the SAP/SDP-binding

(just like the **block** action), however those attributes will get mapped as static MMRP entries on the SAP/SDP-binding, which implicitly get instantiated in the data plane as MFIB entries associated with that SAP/SDP-binding for the related group B-MAC. When the action is “end-station”, the default-action must be block:

```
*A:PE-3>config>serv>mrp>mrp-policy# default-action allow
MINOR: SVCNMR #5904 MRP-policy default-action must be block when end-station action exists
```

- The **end-station** action can be used in the inter-domain gateways when, for instance, we do not want MMRP control plane exchanges between domains. The following output shows how to define the static MMRP entries 1000-2000 in PE-3 without receiving any declaration for any of those attributes or having any of those locally configured.

```
# on PE-3
configure
  service
    mrp
      mrp-policy "mrp_policy_3" create
        default-action block
        entry 10 create
          action end-station
          match
            isid 1000 to 2000
          exit
        exit
      exit
    exit
  exit
exit
```

```
*A:PE-3# show service id 100 mfib
```

```
=====
Multicast FIB, Service 100
```

```
=====
Source Address  Group Address          Sap/Sdp Id              Svc Id  Fwd/Blk
-----
*               01:1E:83:00:03:E8    b-sdp:33:100            Local   Fwd
*               01:1E:83:00:03:E9    b-sdp:33:100            Local   Fwd
*               01:1E:83:00:03:EA    b-sdp:33:100            Local   Fwd
*               01:1E:83:00:03:EB    b-sdp:33:100            Local   Fwd
*               01:1E:83:00:03:EC    b-sdp:33:100            Local   Fwd
```

```
---snipped---
```

```
*               01:1E:83:00:07:CD    b-sdp:33:100            Local   Fwd
*               01:1E:83:00:07:CE    b-sdp:33:100            Local   Fwd
*               01:1E:83:00:07:CF    b-sdp:33:100            Local   Fwd
*               01:1E:83:00:07:D0    b-sdp:33:100            Local   Fwd
```

```
-----
Number of entries: 1001
```

```
=====
*A:PE-3#
```


- The mrp-policy can be applied to multiple B-VPLS services as long as the scope of the policy is **template** (the scope can also be **exclusive**).
- Any changes made to the existing policy will be applied immediately to all services where this policy is applied. For this reason, when many changes are required on a mrp-policy, it is recommended that the policy be copied to a work-in-progress policy. That work-in-progress policy can be modified until complete and then written over the original mrp-policy. You can use the **config mrp-policy copy** command to work with the policies in this manner. The **renum** command can also help to change the entries sequence order.

```
A:PE-3# configure service mrp copy
- copy <src-mrp-policy> to <dst-mrp-policy>

<src-mrp-policy>      : [32 chars max]
<dst-mrp-policy>      : [32 chars max]

A:PE-3#

A:PE-3# configure service mrp mrp-policy "mrp_policy_3" renum
- renum <src-entry-id> to <dst-entry-id>

<src-entry-id>        : [1..65535]
<dst-entry-id>        : [1..65535]

A:PE-3#
```

- The **no** form of the **mrp-policy** command deletes the mrp-policy. An mrp policy cannot be deleted until it is removed from all the SAPs/SDP-bindings where it is applied.

ISID-Based Filters

The MMRP policies help to control the exchange of group B-MAC attributes across domains. Based on the registration state of a specific group B-MAC on a SAP/SDP-binding, the broadcast/unknown-unicast/multicast traffic for a particular I-VPLS will be allowed or dropped. However, to avoid that ANY local ISID packet is flooded to the remote B-VPLS domain, all the packets tagged with the local ISIDs at the gateway PEs need to be filtered at the data plane. ISID-based filters will prevent the local ISIDs from sending any packet with unicast B-MAC to the remote domain. This is particularly useful for PBB-epipe services across domains, where all the frames use unicast B-MACs and MMRP policies cannot help since they only act on group B-MAC packets.

The following CLI output shows how to configure an ISID-based filter that drops all the traffic sourced from the local ISIDs on PE-1 (note that the default action is drop and it does not show up in the configuration).

```
*A:PE-1# configure filter
```

PBB-VPLS M:1 Service Configuration

```
mac-filter 1 create
  description "drop_local_isids"
  type isid
  entry 10 create
    match
      isid 1000 to 2000
    exit
  log 101
  action forward
exit all
```

Once the filter is configured, it must be applied on a B-VPLS SAP or SDP-binding and always at egress.

```
*A:PE-1# configure service
  vpls 100 customer 1 b-vpls create
    spoke-sdp 111:100 create
      egress
        filter mac 1
      exit
    mrp
      mrp-policy "mrp_policy_1"
    exit
  exit
no shutdown
spoke-sdp 121:100 create
  egress
    filter mac 1
  exit
  mrp
    mrp-policy "mrp_policy_1"
  exit
exit
no shutdown
exit all
```

Some additional comments about ISID-based filters:

- The **type isid** statement must be added before introducing any ISID in the match command, otherwise the system will show an error:

```
*A:PE-1>config>filter>mac-filter>entry>match$ isid 1000 to 2000
MINOR: FILTER #1533 The match criteria entered are not compatible with the Mac
filter type - On a normal filter no ISID or VID match criteria are allowed
```

```
*A:PE-1>config>filter>mac-filter$ type isid
MINOR: FILTER #1561 Cannot change filter type when filter contains entries
```

- Once the operator sets the “type isid”, the filter cannot be applied at ingress. Only egress ISID-based filters are allowed:

```
*A:PE-1>config>service>vpls>mesh-sdp# ingress filter mac 1
MINOR: SVCMGR #2050 Can not apply filter of type 'isid' on ingress
```

- Like any filter or MMRP policy, the filter can be applied to multiple B-VPLS services as long as the scope of the policy is “template” (the scope can also be “exclusive”).
- The following command shows the filter configuration and packets that have matched the filter (field “Egr. Matches”):

```
*A:PE-1# show filter mac 1
=====
Mac Filter
=====
Filter Id      : 1                               Applied      : Yes
Scope         : Template                         Def. Action   : Drop
Entries       : 1                               Type         : isid
Description   : drop_local_isids
-----
Filter Match Criteria : Mac
-----
Entry         : 10                               FrameType    : Ethernet
Description   : (Not Specified)
Log Id        : 101
ISID          : 1000..2000
Match action  : Forward
Next Hop      : Not Specified
Ing. Matches  : 0 pkts
Egr. Matches  : 14 pkts (1624 bytes)
=====
*A:PE-1#
```

- Like any other filter, the matching packets can be “logged”. An example follows (note that the Ethertype is 0x88e7, which is the default standard etype for PBB):

```
*A:PE-1# show filter log 101
=====
Filter Log
=====
Admin state   : Enabled
Description   : Default filter log
Destination   : Memory
Wrap          : Enabled
-----
Maximum entries configured : 1000
Number of entries logged   : 45
-----
2015/03/12 09:16:48 Mac Filter: 1:10 Desc:
Interface: int-PE-1-MTU-1 Direction: Egress Action: Forward
VID match: 0
Src MAC: 00-31-31-31-31-31 Dst MAC: 00-11-11-11-00-0f EtherType: 88e7
Hex: 00 00 00 02 4a c4 ff 00 01 41 4a c6 01 02 00 04
      08 06 00 01 08 00 06 04 00 02 4a c6 01 02 00 04
      ac 10 18 02 4a c4 ff 00 01 41 ac 10 18 01 00 00*

2015/03/12 09:16:48 Mac Filter: 1:10 Desc:
Interface: int-PE-1-MTU-1 Direction: Egress Action: Forward
VID match: 0
Src MAC: 00-31-31-31-31-31 Dst MAC: 00-11-11-11-00-0f EtherType: 88e7
Hex: 00 00 00 02 4a c4 ff 00 01 41 4a c6 01 02 00 04
```

PBB-VPLS M:1 Service Configuration

```
08 00 45 00 00 54 12 56 00 00 40 01 e0 2f ac 10
18 02 ac 10 18 01 00 00 0b 88 c0 04 00 01 55 01*
```

---snipped---

=====

* indicates that the corresponding row element may have been truncated.

*A:PE-1#

B-VPLS and I-VPLS Show and Debug Commands

The following commands can help to check the B-VPLS and I-VPLS configuration and their related parameters. The first is for the B-VPLS, the second for the I-VPLS.

```
*A:MTU-1# show service id 100 base
=====
Service Basic Information
=====
Service Id      : 100                Vpn Id      : 0
Service Type    : b-VPLS
Name            : (Not Specified)
Description     : (Not Specified)
Customer Id     : 1                  Creation Origin : manual
Last Status Change: 03/12/2015 08:50:12
Last Mgmt Change : 03/12/2015 13:34:07
Etree Mode     : Disabled
Admin State     : Up                 Oper State    : Up
MTU             : 2000               Def. Mesh VC Id : 100
SAP Count       : 0                 SDP Bind Count : 2
Snd Flush on Fail : Disabled         Host Conn Verify : Disabled
Propagate MacFlush: Disabled         Per Svc Hashing  : Disabled
Allow IP Intf Bind: Disabled
Temp Flood Time : Disabled           Temp Flood     : Inactive
Temp Flood Chg Cnt: 0
VSD Domain      : <none>
SPI load-balance : Disabled
Oper Backbone Src : 00:11:11:11:11:11
Use SAP B-MAC    : Enabled
i-Vpls Count     : 2
Epipe Count      : 0
=====
Service Access & Destination Points
=====
Identifier                               Type      AdmMTU  OprMTU  Adm  Opr
-----
sdp:111:100 S(192.0.2.1)                 Spok      8000    8000    Up   Up
sdp:112:100 S(192.0.2.2)                 Spok      8000    8000    Up   Up
=====
*A:MTU-1#
*A:MTU-1# show service id 1 base
=====
Service Basic Information
=====
Service Id      : 1                Vpn Id      : 0
Service Type    : i-VPLS
Name            : (Not Specified)
Description     : (Not Specified)
Customer Id     : 1                  Creation Origin : manual
Last Status Change: 03/12/2015 08:52:50
Last Mgmt Change : 03/12/2015 09:06:21
Etree Mode     : Disabled
Admin State     : Up                 Oper State    : Up
MTU             : 1514               Def. Mesh VC Id : 1
SAP Count       : 1                 SDP Bind Count : 0
Snd Flush on Fail : Enabled         Host Conn Verify : Disabled
Propagate MacFlush: Disabled         Per Svc Hashing  : Disabled
```

PBB-VPLS M:1 Service Configuration

```

Allow IP Intf Bind: Disabled
Temp Flood Time   : Disabled           Temp Flood       : Inactive
Temp Flood Chg Cnt: 0
VSD Domain        : <none>
SPI load-balance  : Disabled
b-Vpls Id         : 100                 Oper ISID        : 1
b-Vpls Status     : Up
Snd Flush in bVpls: All-from-me
Flsh On bVpls Fail: Enabled             Prop Flsh fr bVpls: Disabled
Force QTag Fwd    : Disabled

```

----- Service Access & Destination Points

```

-----
Identifier                Type          AdmMTU  OprMTU  Adm  Opr
-----
sap:1/1/4:1               q-tag          1518    1518    Up   Up
=====
*A:MTU-1#

```

The following command shows all the I-VPLS instances multiplexed into a particular B-VPLS.

```

*A:MTU-1# show service id 100 i-vpls
=====
Related i-Vpls services for b-Vpls service 100
=====
i-Vpls SvcId      Oper ISID      Admin      Oper
-----
1                  1              Up          Up
2                  2              Up          Up
-----
Number of Entries : 2
-----
*A:MTU-1#

```

Some useful commands to check the I and B VPLS FIBs correlating C-MACs and B-MACs:

```

*A:MTU-1# show service id 1 fdb pbb
=====
Forwarding Database, i-Vpls Service 1
=====
MAC                Source-Identifier  B-Svc  b-Vpls MAC      Type/Age
-----
00:11:00:00:00:00 sap:1/1/4:1        100    N/A              L/180
00:33:00:00:00:00 b-sdp:111:100     100    00:31:31:31:31:31 L/180
=====
*A:MTU-1#
*A:MTU-1# show service id 100 fdb pbb
=====
Forwarding Database, b-Vpls Service 100
=====
MAC                Source-Identifier  iVplsMACs  Epipes  Type/Age
-----
00:31:31:31:31:31 sdp:111:100       2          0        L/240
4a:c4:ff:00:00:00 sdp:111:100       0          0        L/0
=====
*A:MTU-1#

```

If **mac-names** are used in the configuration, the following commands can help to show the translations:

```
*A:MTU-1# show service pbb mac-name
=====
MAC Name Table
=====
MAC-Name                               MAC-Address
-----
MTU-1                                  00:11:11:11:11:11
MTU-2                                  00:21:21:21:21:21
MTU-3                                  00:31:31:31:31:31
=====
*A:MTU-1#
*A:MTU-1# show service pbb mac-name "MTU-3" detail
=====
Services Using MAC name='MTU-3' addr='00:31:31:31:31:31'
=====
Svc-Id                                ISID
-----
1                                      N/A
-----
Number of services: 1
=====
*A:MTU-1#
```

The following command shows the base MAC notification parameters as well as the source B-MAC configured at the service PBB level. Note that those values are overridden by any potential mac-notification or source B-MAC values configured under the B-VPLS service context.

```
*A:MTU-1# show service pbb base
=====
PBB MAC Information
=====
MAC-Notif Count                        : 3
MAC-Notif Interval                     : 1
Source BMAC                            : 00:11:11:11:11:11
=====
*A:MTU-1#
```

If mac-notification is used in a particular B-VPLS, the configured least significant bits for the sap-bmac on a particular MC-LAG can be shown by using the detailed view of the **show lag** command:

```
*A:MTU-1# show lag 1 detail
=====
LAG Details
=====
Description                            : N/A
-----
Details
-----
Lag-id                                : 1                               Mode                : access
Adm                                    : up                               Opr                  : up
```

PBB-VPLS M:1 Service Configuration

---snipped---

MC Peer Address	: 192.0.2.21	MC Peer Lag-id	: 1
MC System Id	: 00:00:00:00:00:01	MC System Priority	: 65535
MC Admin Key	: 15	MC Active/Standby	: active
MC Lacp ID in use	: true	MC extended timeout	: false
MC Selection Logic	: local master decided		
MC Config Mismatch	: no mismatch		
Source BMAC LSB	: use-lacp-key	Oper Src BMAC LSB	: 00:0f

---snipped---

*A:MTU-1#

The following commands allow the operator to check the LDP label mapping, label withdrawal, messages and also the MAC-flush messages for regular VPLS, for I-VPLS and B-VPLS including the PBB extensions and TLVs.

```
*A:MTU-1# show debug
debug
  router "Base"
    ldp
      peer 192.0.2.1
        event
        exit
        packet
          init detail
          label detail
        exit
      exit
    peer 192.0.2.2
      event
      exit
      packet
        init detail
        label detail
      exit
    exit
  exit
exit
*A:MTU-1#
```


The following debug commands can help the operator to troubleshoot MMRP.

```
*A:MTU-1# debug service id 100 mrp
- mrp
- no mrp

all-events      - Enable/disable MRP debugging for all events
[no] applicant-sm - Enable/disable MRP debugging for applicant state machine changes
[no] leave-all-sm - Enable/disable MRP debugging for leave all state machine changes
[no] mmrp-mac    - Enable/disable MRP debugging for a particular MAC address
[no] mrpdu       - Enable/disable MRP debugging for Rx/Tx MRP PDUs
[no] mvrp-vlan   - Enable/disable debugging for a particular vlan
[no] periodic-sm - Enable/disable MRP debugging for periodic state machine changes
[no] registrant-sm - Enable/disable MRP debugging for registrant state machine changes
[no] sap         - Enable/disable MRP debugging for a particular SAP
[no] sdp         - Enable/disable MRP debugging for a particular SDP

*A:MTU-1#
```

Conclusion

PBB-VPLS allows the service providers to scale VPLS services by multiplexing customer I-VPLS instances into one or more B-VPLS instances. This multiplexing dramatically reduces the number of services, pseudowires and MAC addresses in the core and therefore allows the service provider to scale Layer 2 multi-point networks and provide services across international backbones.

The example used in this section shows the configuration of the customer and backbone VPLS instances as well as all the related features which are required for this environment. Show and debug commands have also been suggested so that the operator can verify and troubleshoot the service.

Shortest Path Bridging for MAC

In This Chapter

This section describes advanced shortest path bridging for MAC configurations.

Topics in this section include:

- [Applicability on page 1312](#)
- [Overview on page 1313](#)
- [Configuration on page 1315](#)
- [Conclusion on page 1344](#)

Applicability

The example presented in this section is applicable to the 7950 XRS, 7750 SR-c4/c12, 7750 SR-7/12 and 7450 ESS-6/6v/7/12, and requires IOM3-XP/IMM or higher-based line cards.

It is not supported in 7750 SR-1, 7450 ESS-1 or IOM-2 or lower-based line cards.

The configuration was tested on release 13.0.R3. Shortest Path Bridging for MAC (SPBM) is supported from 10.0.R4. SPB Static MAC, static Backbone-Service Instance Identifiers (ISIDs) and ISID-policies for SPB are supported from 11.0.R4 onwards.

Overview

SPB enables a next generation control plane for Provider Backbone Bridges (PBB) and PBB-VPLS that adds the stability and efficiency of link state to unicast and multicast services (I-VPLS and Epipes). In addition, SPBM provides resiliency, load-balancing and multicast optimization without the need for any other control plane in the B-VPLS (for example, there is no need for Spanning Tree or G.8032 or Multiple MAC Registration Protocol (MMRP)).

SPBM exploits the complete knowledge of backbone addressing, which is a key consequence of the PBB hierarchy, by advertising and distributing the Backbone MACs (BMACs) through a link-state protocol, namely IS-IS. An immediate effect of this is that the old “flood-and-learn” can at last be turned off in the backbone and every B-VPLS node in the network will know what destination BMAC addresses are expected and valid. As a result of that, receiving an unknown unicast BMAC on a B-VPLS SAP/PW is indicative of an error, whereupon the frame is discarded (due to the Reverse Path Forwarding Check – RPFC – performed in SPBM) instead of flooded. Furthermore, SPBM allows condensing all the relevant information distribution (unicast and multicast) into a single control protocol: IS-IS.

SPBM can be easily enabled on the existing B-VPLS instances being used for multiplexing I-VPLS/Epipe services, providing the benefits summarized below:

- Per-service flood containment (for I-VPLS services) without the need for an additional protocol such as MMRP.
- Loop avoidance in the B-VPLS domain without the need for MSTP or other technologies.
- No unknown BMAC flooding in the B-VPLS domain.
- No need for MAC notification mechanisms or vMEPs in the B-VPLS to update the B-VPLS Forwarding Data Bases (FDBs) (vMEPs can still be configured though for OAM purposes).

Some other characteristics of the SPB implementation in the SR OS are listed below:

- The SR OS SPB implementation always uses Multi-Topology (MT) topology instance zero. However, up to four logical instances (that is, SPB instances in different B-VPLS services) are supported if different topologies are required for different services.
- Area addresses are not used and SPB is assumed to be a single area. SPB must be consistently configured on nodes in the system. SPB Regions information and IS-IS hello logic that detect mismatched configuration are not supported. IS-IS area is always zero.
- SPB uses All-Intermediate-Systems 09-00-2B-00-00-05 destination MAC to communicate.
- SPB Source ID is always zero.
- SPB uses a separate instance of IS-IS from the base IP IS-IS. IS-IS for SPB is configured in the SPB context under the B-VPLS component. Up to four ISIS-SPB instances are

supported, where the instance identifier can be any number between 1024 and 2047. The instance number is not in TLVs.

- Two ECT-ALGORITHMS (IEEE 802.1aq Equal Cost Tree Algorithms) per SPB instance are supported: low-path-id and high-path-id algorithms.
- SPB Link State Protocol Data Units (Link State Packets) contain BMACs, ISIDs (for multicast services) and link and metric information for an IS-IS database.
 - Epipe ISIDs are not distributed in SR OS SPB allowing high scalability of PBB Epipes.
 - I-VPLS ISIDs are distributed in SR OS SPB and the respective multicast group addresses (composed of PBB-OUI plus ISID) are automatically populated in a manner that provides automatic pruning of multicast to the subset of the multicast tree that supports an I-VPLS with a common ISID. This replaces the function of MMRP and is more efficient than MMRP.
- Multiple ISIS-SPB adjacencies between two nodes are not supported as per the IEEE 802.1aq standard specification. If multiple links between two nodes exist, LAG must be used.

Configuration

This section will describe the configuration of SPBM on the 7x50 as well as the available troubleshooting commands.

Basic SPBM Configuration

Figure 194 shows the topology used as an example of a basic SPBM configuration.

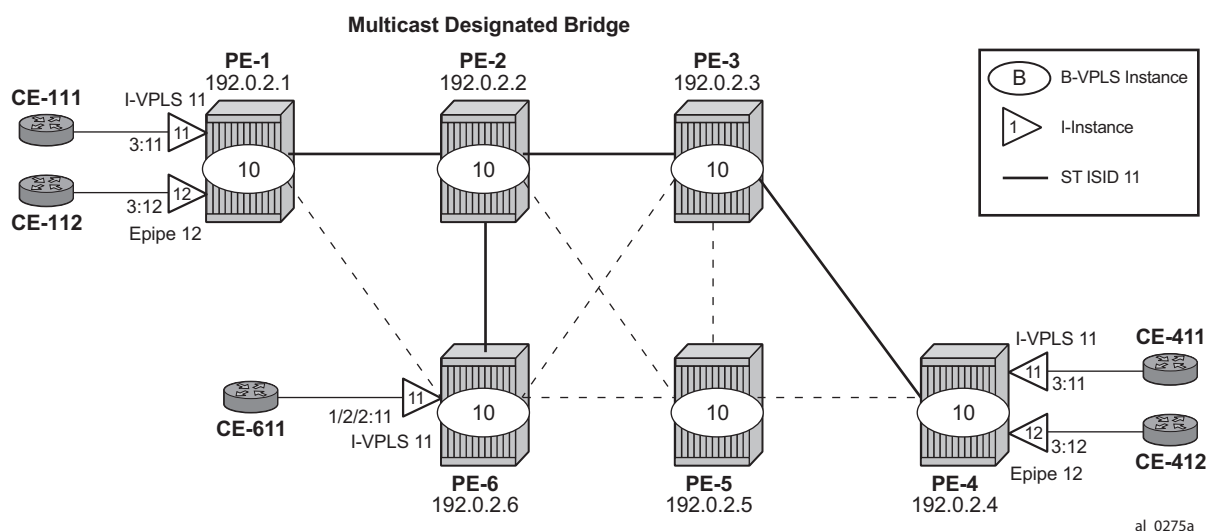


Figure 194: Basic SPBM Topology

Assume the following protocols and objects are configured beforehand:

- The six PEs shown in Figure 194 are running IS-IS for the global routing table with all the interfaces being Level-2.
- LDP is used as the MPLS protocol to signal transport tunnel labels.
- LDP SDPs are configured among the six PEs, in the way it is depicted in Figure 194 (dashed lines and bold lines among PEs).

Once the network infrastructure is properly running, the actual service configuration can be carried out. In the example, B-VPLS 10 will provide backbone connectivity for the services I-VPLS 11 and Epipe 12.

Basic SPBM Configuration

The SPBM configuration is only relevant to the B-VPLS instance and can be added to an existing B-VPLS, assuming that such a B-VPLS does not contain any non-SPB-compatible configuration parameters. The following parameters are not supported in SPB-enabled B-VPLS instances:

- Mesh SDPs (only SAPs or spoke-sdps are supported in SPB-enabled B-VPLS)
- Spanning Tree Protocol (STP)
- Split-Horizon Groups
- Non-conditional Static-macs (configured under sap/spoke-sdps, see section about static BMAC configuration)
- G.8032
- Propagate-mac-flush and send-flush-on-failure
- Maximum number of MACs (max-nbr-mac-addr)
- Bridge Protocol Data Unit (BPDU) translation
- Layer 2 Protocol Termination (L2PT)
- MAC-pinning
- Oper-groups
- MAC-move
- Any BGP, BGP-AD (BGP Autodiscovery) or BGP-VPLS (BGP Virtual Private LAN Services) parameters
- Endpoints
- Local/remote age
- MAC-notification
- MAC-protect
- Multiple MAC Registration Protocol (MMRP)
- Provider-tunnel
- Temporary flooding

Assuming all the parameters mentioned above are not configured in the B-VPLS (B-VPLS 10 in the example), SPBM can be enabled. The SPBM parameters are all configured under the **config>service>vpls(b-vpls)>spb** and **config>service>vpls(b-vpls)>spoke-sdp/sap>spb** contexts:

```
*A:PE-1# configure service vpls 10 spb ?
- no spb
- spb [<isis-instance>] [fid <fid>] [create]
<isis-instance>      : [1024..2047]
<fid>                 : [1..4095]
    level             + Configure SPB level information
[no] lsp-lifetime      - Configure LSP lifetime
[no] lsp-wait          - Configure ISIS LSP wait times
[no] overload          - Configure the local router so that it appears to be overloaded
```



```

[no] overload-on-bo* - Configure the local router so that it appears to be overloaded at
boot up
[no] shutdown        - Administratively enable or disable the operation of ISIS
[no] spf-wait        - Configure ISIS SPF wait times

*A:PE-1# configure service vpls 10 spoke-sdp 35:10 spb ?
  - no spb
  - spb [create]
    <create>          : keyword
      level          + Configure SPB level information
[no] lsp-pacing-int* - Configure the interval between LSP packets are sent from the inter-
face
[no] retransmit-int* - Configure the minimum interval between LSP packets retransmission
for the given interface
[no] shutdown        - Administratively Enable/disable the interface

*A:PE-1# configure service vpls 10 spoke-sdp 35:10 spb level 1 ?
  - level <[1..1]>
[no] hello-interval - Configure hello-interval for this interface
[no] hello-multipli* - Configure hello-multiplier for this level
[no] metric         - Configure IS-IS interface metric for IPv4 unicast

```

The parameters configured under the spb context refer to the SPB IS-IS and they should be configured following the same considerations as for the IS-IS base instance:

- spb [<isis-instance>] [fid <fid>] [create]
 - <isis-instance> identifies the SPB ISIS process. Up to four different ISIS SPB processes can be run in a system (range 1024 to 2047).
 - <fid> or *forwarding identifier* identifies the standard SPBM B-VID which is signaled in IS-IS with each advertised BMAC. Each B-VPLS has a single configurable FID.
- spb>lsp-lifetime <seconds> : [350..65535]
- spb>lsp-wait <lsp-wait> [<lsp-initial-wait> [<lsp-second-wait>]]
 - <lsp-wait> : [1..120] – seconds
 - <initial-wait> : [0..100] – seconds
 - <second-wait> : [1..100] – seconds
- spb>overload
- spb>overload-on-boot
- spb>spf-wait <spf-wait> [<spf-initial-wait> [<spf-second-wait>]]
 - <spf-wait> : [1..120] – seconds
 - <initial-wait> : [10..100000] – milliseconds
 - <second-wait> : [1..100000] – milliseconds

- spoke-sdp/sap>spb>lsp-pacing-interval <milli-seconds> : [0..65535]
 - spoke-sdp/sap>spb>retransmit-interval <seconds> : [1..65535]
 - spoke-sdp/sap>spb>level 1>hello-interval <seconds> : [1..20000]
 - spoke-sdp/sap>spb>level 1>hello-multiplier <multiplier> : [2..100]

In the same way lsp-wait (initial-wait) and spf-wait (initial wait) can be tuned in the base router IS-IS instance to minimize the convergence time (to 0 and 10 respectively), the equivalent SPB IS-IS parameters should also be adjusted so that failover time is minimized at the service level.

The following parameters are specific to SPBM (note that only IS-IS level 1 is supported for SPB):

- spb>level 1>bridge-priority <bridge-priority> : [0..15]
 - This parameter will influence the election of the Multicast Designated bridge through which all the ST (Single Trees) for the multicast traffic will be established. The default value will be lowered on that node where the Multicast Designated bridge function is desired, normally because that node is the best connected node. Note that in the example of this document, PE-2 is the Multicast Designated bridge for B-VPLS 10 and therefore, PE-2 will be the root of the STs for the I-VPLS instances in that B-VPLS. Default value = 8.
- spb>level 1>ect-algorithm fid-range <fid-range> {low-path-id|high-path-id}
 - This command defines the ect-algorithm used and the FIDs assigned. Two algorithms are supported: low-path-id and high-path-id. They can provide the required path diversity for an efficient load balancing in the B-VPLS. Default = fid-range 1-4095 low-path-id
- spb>level 1>forwarding-tree-topology unicast {spf|st}
 - This command configures the type of tree that will be used for unicast traffic: shortest path tree or single tree. The multicast traffic (that encapsulated I-VPLS broadcast, unicast and multicast (BUM) traffic always uses the ST path (Single Tree path). Note that using SPF for unicast traffic can produce some packet re-ordering for unicast traffic compared to BUM traffic as different trees are used, therefore, when the B-VPLS transports I-VPLS traffic and the unicast and multicast trees do not follow the same path, it is recommended to use ST paths for unicast and multicast. Default value = spf.
- spoke-sdp/sap>spb>level 1>metric <ipv4-metric> : [1..16777215]
 - This command configures the metric for each SPB interface (spoke-SDP or SAP). This value helps influence the SPF calculation in order to pick a certain path for the traffic to a remote system BMAC. Note that when the SPB link metric advertised by two peers is different, the maximum value is chosen according to the RFC 6329. Default metric = 10.

As an example, the following CLI output shows the relevant configuration of PE-1 and PE-2 (the Multicast Designated Bridge). SPB has to be created and enabled (no shutdown) at B-VPLS service level first and then created and enabled under each and every SAP/spoke-sdp in the B-VPLS. Non-SPB-enabled SAPs/spoke-sdps can exist in the SPB B-VPLS only if conditional static-macs are configured for them (refer to [Static BMACs and Static ISIDs Configuration on page 1335](#)). Note that, as for regular B-VPLS services, the service-mtu has to be changed from the default value (1500) to a number 18-bytes greater than the I-VPLS service-mtu in order to allow for the PBB encapsulation.

```
*A:PE-1# configure service
      pbb
        source-bmac 00-00-00-01-01-01
        mac-name "PE-1" 00-00-00-01-01-01
        mac-name "PE-2" 00-00-00-02-02-02
        mac-name "PE-3" 00-00-00-03-03-03
        mac-name "PE-4" 00-00-00-04-04-04
        mac-name "PE-5" 00-00-00-05-05-05
        mac-name "PE-6" 00-00-00-06-06-06
      exit
    vpls 10 customer 1 b-vpls create
      service-mtu 2000
      spb 1024 fid 10 create
        overload-on-boot timeout 60
        spf-wait 2 50 100
        lsp-wait 8 0 1
        no shutdown
      exit
      spoke-sdp 12:10 create
        spb create
        no shutdown
      exit
      no shutdown
    exit
    spoke-sdp 16:10 create
      spb create
      no shutdown
    exit
    no shutdown
  exit
  no shutdown
vpls 11 customer 1 i-vpls create
  pbb
    backbone-vpls 10
  exit
  exit
  sap 1/1/3:11 create
  exit
  no shutdown
exit
epipe 12 customer 1 create
  pbb
    tunnel 10 backbone-dest-mac "PE-4" isid 12
  exit
  sap 1/1/3:12 create
  exit
  no shutdown
```

Basic SPBM Configuration

```
exit
```

As discussed, the bridge-priority will influence the election of the Multicast Designated Bridge. By making PE-2's bridge-priority zero, it ensures that PE-2 becomes the root of all the STs for B-VPLS 10 as long as the priority for the rest of the PEs is larger than zero. In case of a tie, the PE owning the lowest system BMAC will be elected as Multicast Designated Bridge. [Figure 194](#) shows the ST for I-VPLS 11 (see a thicker continuous line representing the ST). Note that PE-2 is the root of the ST tree.

```
A:PE-2# configure service
      pbb
        source-bmac 00:00:00:02:02:02
        mac-name "PE-1" 00:00:00:01:01:01
        mac-name "PE-2" 00:00:00:02:02:02
        mac-name "PE-3" 00:00:00:03:03:03
        mac-name "PE-4" 00:00:00:04:04:04
        mac-name "PE-5" 00:00:00:05:05:05
        mac-name "PE-6" 00:00:00:06:06:06
      exit
    vpls 10 customer 1 b-vpls create
      service-mtu 2000
      spb 1024 fid 10 create
        level 1
          bridge-priority 0
        exit
        overload-on-boot timeout 60
        spf-wait 2 50 100
        lsp-wait 8 0 1
        no shutdown
      exit
    spoke-sdp 21:10 create
      spb create
        no shutdown
      exit
      no shutdown
    exit
    spoke-sdp 23:10 create
      spb create
        no shutdown
      exit
      no shutdown
    exit
    spoke-sdp 25:10 create
      spb create
        no shutdown
      exit
      no shutdown
    exit
    spoke-sdp 26:10 create
      spb create
        no shutdown
      exit
      no shutdown
    exit
    no shutdown
  exit
```

The rest of the nodes will be configured accordingly. Note that SPB instance 1024 will set up SPF (Shortest Path First) trees for unicast traffic and a ST (Single Tree) per ISID with PE-2 as the root-bridge (since it has the lowest bridge-priority 0 configured) for BUM traffic. The ect-algorithm chosen for the B-VPLS FID (10) is the low-path-id (default one).

Once SPBM is configured as discussed above on all the six nodes, the six system BMACs and the ISID 11 will be advertised by SPB IS-IS.

The following show commands can help understand the IS-IS configuration for SPB 1024 and the BMACs populated by IS-IS:

- `show service id spb base`: provides the SPB configuration and parameters for a particular SPB B-VPLS.
- `show service id 10 spb fdb`: provides the B-VPLS FDB that has been populated by IS-IS, for the unicast and multicast entries.

```
A:PE-1# show service id 10 spb base
=====
Service SPB Information
=====
Admin State      : Up                Oper State      : Up
ISIS Instance    : 1024              FID             : 10
Bridge Priority   : 8                Fwd Tree Top Ucast : spf
Fwd Tree Top Mcast : st
Bridge Id        : 80:00:00:00:00:01:01:01
Mcast Desig Bridge : 00:00:00:00:00:02:02:02
=====
Router Base ISIS Instance 1024 Interfaces
=====
Interface                Level CircID  Oper State  L1/L2 Metric
-----
sdp:12:10                 L1      65536    Up          10/-
sdp:16:10                 L1      65537    Up          10/-
-----
Interfaces : 2
=====
FID ranges using ECT Algorithm
-----
1-4095    low-path-id
=====
A:PE-1#
*A:PE-1# show service id 10 spb fdb
=====
User service FDB information
=====
MAC Addr      UCast Source      State  MCast Source      State
-----
00:00:00:02:02:02 12:10             ok     12:10             ok
00:00:00:03:03:03 12:10             ok     12:10             ok
00:00:00:04:04:04 12:10             ok     12:10             ok
00:00:00:05:05:05 12:10             ok     12:10             ok
00:00:00:06:06:06 16:10             ok     12:10             ok
-----
Entries found: 5
```

```
=====
*A:PE-1#
```

It can be seen above that the unicast (SPF) tree and the multicast (ST) tree differ with respect to PE-6.

The following commands help check the unicast and multicast topology for B-VPLS 10:

- **show service id 10 spb routes** provides a detailed view of the unicast and multicast routes computed by SPF. As shown below, the SPB unicast and multicast routes match on PE-2 since this node is the Multicast Designated Bridge. Unicast and multicast routes will differ on most other nodes.
- **show service id 10 spb mfib** and **show service id 10 mfib** shows information of the MFIB entries generated in the B-VPLS as well as the outgoing interface (OIF) associated with those MFIB entries.

```
*A:PE-2# show service id 10 spb routes
```

```
=====
MAC Route Table
```

```
=====
FID  MAC Addr                               Ver.  Metric
    NextHop If                               SysID
-----
```

```
Fwd Tree: unicast
```

```
-----
10   00:00:00:01:01:01                     4      10
      sdp:21:10                             PE-1
10   00:00:00:03:03:03                     6      10
      sdp:23:10                             PE-3
10   00:00:00:04:04:04                     8      20
      sdp:23:10                             PE-3
10   00:00:00:05:05:05                    12      10
      sdp:25:10                             PE-5
10   00:00:00:06:06:06                    14      10
      sdp:26:10                             PE-6
```

```
Fwd Tree: multicast
```

```
-----
10   00:00:00:01:01:01                     4      10
      sdp:21:10                             PE-1
10   00:00:00:03:03:03                     6      10
      sdp:23:10                             PE-3
10   00:00:00:04:04:04                     8      20
      sdp:23:10                             PE-3
10   00:00:00:05:05:05                    12      10
      sdp:25:10                             PE-5
10   00:00:00:06:06:06                    14      10
      sdp:26:10                             PE-6
```

```
-----
No. of MAC Routes: 10
```

```
=====
ISID Route Table
```

```

=====
FID  ISID                               Ver.
     NextHop If                        SysID
-----
10   11                               17
     sdp:21:10                        PE-1
     sdp:23:10                        PE-3
     sdp:26:10                        PE-6
-----
No. of ISID Routes: 1
=====
*A:PE-2#

*A:PE-2# show service id 10 spb mfib
=====
User service MFIB information
=====
MAC Addr          ISID      Status
-----
01:1E:83:00:00:0B 11      Ok
-----
Entries found: 1
=====
*A:PE-2#

*A:PE-2# show service id 10 mfib
=====
Multicast FIB, Service 10
=====
Source Address    Group Address      Sap/Sdp Id          Svc Id   Fwd/Blk
-----
*                 01:1E:83:00:00:0B  b-sdp:21:10        Local    Fwd
                  b-sdp:23:10        Local    Fwd
                  b-sdp:26:10        Local    Fwd
-----
Number of entries: 1
=====
*A:PE-2#

```

SPB Multicast Trees (STs) are pruned for each particular I-VPLS ISID, based on the advertisement of I-VPLS ISIDs in SPB IS-IS by each individual PE. Multicast B-VPLS traffic not belonging to any particular I-VPLS follows the *Default Tree*. The Default Tree is an ST for the B-VPLS which is not pruned and therefore reaches all the PE nodes in the B-VPLS. For instance, Ethernet-CFM CCM messages sent from vMEPs configured on the SPB B-VPLS will use the default tree. The default tree does not consume MFIB entries and can be checked in each node through the use of the following command:

```

*A:PE-5# tools dump service id 10 spb default-multicast-list
saps : { }
spoke-sdps : { 52:10 }

```

As it can be noted above, PE-5 is not part of the tree for I-VPLS 11. However, as with any SPB node part of B-VPLS 10, PE-5 is part of the default tree. Refer to [Configuration of ISID-Policies in SPB B-VPLS on page 1340](#) to see more use-cases for the Default Tree.

The following tools commands allow the operator to easily see the forwarding path (unicast and multicast) followed by the traffic to a remote node, with the aggregate metric from the source.

```
*A:PE-1# tools dump service id 10 spb fid 10 forwarding-path destination PE-4 forwarding-
tree unicast
Hop  BridgeId                Metric From Src
0    PE-1                    0
1    PE-2                    10
2    PE-3                    20
3    PE-4                    30

*A:PE-1# tools dump service id 10 spb fid 10 forwarding-path destination PE-4 forwarding-
tree multicast
Hop  BridgeId                Metric From Src
0    PE-1                    0
1    PE-2                    10
2    PE-3                    20
3    PE-4                    30
```

In large networks or networks where IP multicast, PBB and PBB-SPB services coexist, the data plane MFIB entries is a hardware resource that should be periodically checked. The tools dump service vpls-mfib-stats shows the total number of hardware MFIB entries (40k entries, in chassis-mode d) and the entries being used by IP multicast or PBB (MMRP or SPB). The tools dump service vpls-pbb-mfib-stats shows the breakdown between MFIB entries populated by MMRP or by SPB and the individual limits, system-wide and per service:

```
*A:PE-2# tools dump service vpls-mfib-stats
Service Manager VPLS MFIB info at 000 00:45:28.450:

Statistics last cleared at 000 00:00:00.000

-----+-----
Statistic | Count
-----+-----
HW limit SG entries | 40959 # total number of MFIB entries
Current SG entries | 1
Limit Non PBB SG entries | 16383 # IP Multicast MFIB limit
Current Non PBB SG entries | 0
---snip---

*A:PE-2# tools dump service vpls-pbb-mfib-stats detail
Service Manager VPLS PBB MFIB statistics at 000 00:45:28.540:

Usage per Service
ServiceId  MFIB User  Count
-----+-----+-----
10         spb          1
-----+-----+-----
Total      1
```



```
MMRP
  Current Usage      :      0
  System Limit       :  8191 Full, 40959 EOnly
  Per Service Limit  : 2048 Full,  8192 EOnly
```

```
SPB
  Current Usage      :      1
  System Limit       :  8191
  Per Service Limit  :  8191
```

Finally the following debug commands can help monitor the SPB IS-IS process and the protocol PDU exchanges:

- debug service id <svcId> spb
- debug service id <svcId> spb adjacency
- debug service id <svcId> spb interface
- debug service id <svcId> spb l2db
- debug service id <svcId> spb lsdb
- debug service id <svcId> spb packet <detail>
- debug service id <svcId> spb spf

Control and User B-VPLS Configuration

The SR OS implementation of SPB allows a single SPB IS-IS instance to control the paths and FDBs of many B-VPLS instances. This is done by using the control B-VPLS, user B-VPLS and fate-sharing concepts.

The control B-VPLS will be SPB-enabled and configured with all the related SPB IS-IS parameters. Although the control B-VPLS might or might not have I-VPLS/Epipes directly attached, it must be configured on all the nodes where SPB forwarding is expected to be active. SPB uses the logical instance and a Forwarding ID (FID) to identify SPB locally on the node. That FID must be consistently configured on all the nodes where the B-VPLS exists. User B-VPLS are other instances of B-VPLS that are usually configured to separate the traffic for manageability reasons, QoS or ECT different treatment.

Figure 195 illustrates the control B-VPLS (B-VPLS 20) and user B-VPLS (B-VPLS 21) concept (in this case there is only one user B-VPLS but there might be many B-VPLS sharing fate with the same control B-VPLS). Note that both B-VPLS must share the same topology and both B-VPLS must share exactly the same interfaces. The user B-VPLS, which is linked to the control B-VPLS by its FID, follows (that is, inherits the state of) the control B-VPLS but may use a different ECT path in case of equal metric paths, like in this example: FID 20, that is, the control B-VPLS, follows the low-path-id ECT, whereas FID 21, for example, the user B-VPLS, follows the high-path-id ECT.

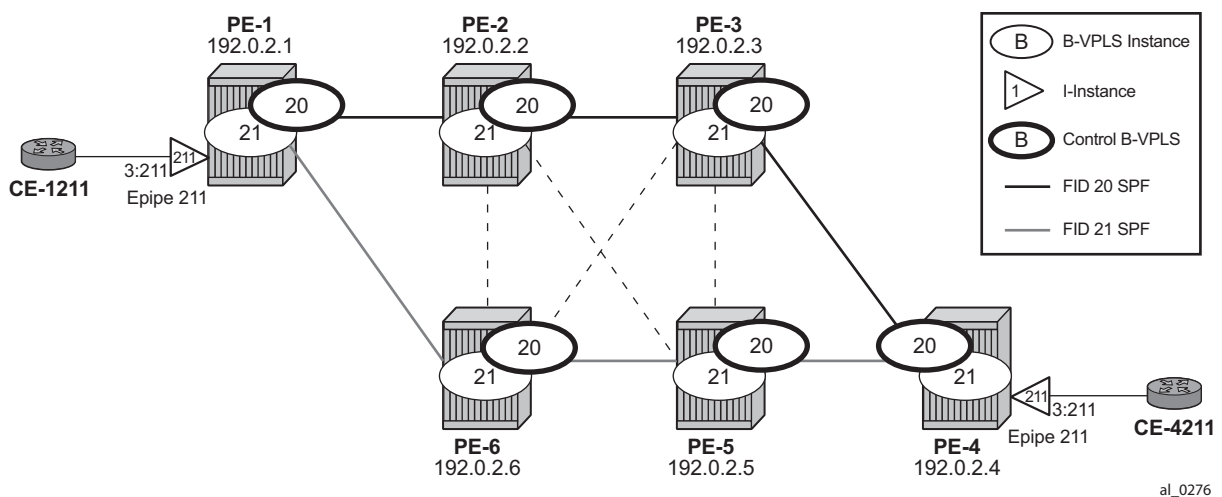


Figure 195: Control and User B-VPLS Test Topology

The configurations of B-VPLSs 20 and 21, on PE-1 and PE-2, are shown below. The **spbm-control-vpls 20 fid 21 in B-VPLS 21** command associates FID 21 to the user B-VPLS and links the B-VPLS to its control B-VPLS 20.

```
*A:PE-1# configure service
      vpls 20 customer 1 b-vpls create
        service-mtu 2000
        spb 1025 fid 20 create
          level 1
            ect-algorithm fid-range 21-4095 high-path-id
          exit
        no shutdown
      exit
    spoke-sdp 12:20 create
      spb create
        no shutdown
      exit
    no shutdown
  exit
  spoke-sdp 16:20 create
    spb create
      no shutdown
    exit
  no shutdown
exit
vpls 21 customer 1 b-vpls create
  service-mtu 2000
  spbm-control-vpls 20 fid 21
  spoke-sdp 12:21 create
    no shutdown
  exit
  spoke-sdp 16:21 create
    no shutdown
  exit
  no shutdown
exit
epipe 211 customer 1 create
  pbb
    tunnel 21 backbone-dest-mac "PE-4" isid 211
  exit
  sap 1/1/3:211 create
  exit
  no shutdown
exit

*A:PE-2# configure service
      vpls 20 customer 1 b-vpls create
        service-mtu 2000
        spb 1025 fid 20 create
          level 1
            ect-algorithm fid-range 21-4095 high-path-id
          exit
        no shutdown
      exit
    spoke-sdp 21:20 create
```

Control and User B-VPLS Configuration

```
        spb create
          no shutdown
        exit
      no shutdown
    exit
    spoke-sdp 23:20 create
      spb create
        no shutdown
      exit
    no shutdown
  exit
  spoke-sdp 25:20 create
    spb create
      no shutdown
    exit
  no shutdown
exit
spoke-sdp 26:20 create
  spb create
    no shutdown
  exit
no shutdown
exit
vpls 21 customer 1 b-vpls create
  service-mtu 2000
  spbm-control-vpls 20 fid 21
  spoke-sdp 21:21 create
    no shutdown
  exit
  spoke-sdp 23:21 create
    no shutdown
  exit
  spoke-sdp 25:21 create
    no shutdown
  exit
  spoke-sdp 26:21 create
    no shutdown
  exit
no shutdown
exit
```

If there is a mismatch between the topology of a user B-VPLS and its control B-VPLS, only the user B-VPLS links and nodes that are in common with the control B-VPLS will function.

User B-VPLS instances supporting only unicast services (PBB-Epipes) may share the FID with the other B-VPLS (control or user). This is a configuration shortcut that reduces the LSP advertisement size for B-VPLS services but results in the same separation for forwarding between the B-VPLS services. In the case of PBB-Epipes only BMACs are advertised per FID but BMACs are populated per B-VPLS in the FIB. If I-VPLS services are to be supported on a B-VPLS that B-VPLS must have an independent FID.

Note that although user B-VPLS 21 does not have any SPB setting (other than the spbm-control-vpls) the spoke-sdps use the same SDPs as the parent control B-VPLS 20. The **show service id**

<user b-vpls> **spb fate-sharing** command shows the control spoke-sdp/saps that control the user spoke-sdp/saps.

```
*A:PE-1# show service id 21 spb fate-sharing
=====
User service fate-shared sap/sdp-bind information
=====
```

Control SvcId	Control Sap/ SdpBind	FID	User SvcId	User Sap/ SdpBind
20	12:20	21	21	12:21
20	16:20	21	21	16:21

```
=====
*A:PE-1#
```

SPBM Access Resiliency Configuration

The following example shows how to configure an I-VPLS/Epipe attached to an SPB-enabled B-VPLS when access resiliency is used.

Multi-chassis LAG (MC-LAG) is the only resiliency mechanism supported for PBB-Epipes. The MC-LAG active node will advertise the MC-LAG BMAC (or sap-bmac) in SPB IS-IS. In case of failure, when the standby node takes over, it will advertise the MC-LAG sap-bmac. Note that without SPB, the MC-LAG solution for PBB-Epipe required the use of mac-notification and periodic mac-notification. SPB provides a faster and more efficient solution without the need for any extra mac-notification mechanism. In the example described in this section, Epipe 31 uses MC-LAG access resiliency to get connected to the B-VPLS 30 on nodes PE-2 and PE-6.

As far as I-VPLS access resiliency is concerned, the same mechanisms supported for regular B-VPLS are supported for SPB-enabled B-VPLS, except for G.8032. A very important aspect of the I-VPLS resiliency is a proper mac-flush propagation when there is a failure at the I-VPLS access links.

If the SPB-enabled B-VPLS uses B-SAPs for its connectivity to the backbone, there is no mac-flush propagation (since there is no TLDP). In this case, if MC-LAG is used and there is an MC-LAG switchover, the new active chassis will keep using the same source BMAC, such as the sap-bmac, and it will advertise it in the B-VPLS domain so that the remote FDBs can be properly updated. No mac-flush is required in this case.

When the B-VPLS uses spoke-sdps for its backbone connectivity, the traditional LDP MAC flush propagation mechanisms and commands can be used as follows:

- **send-flush-on-failure** works as expected when SPB is used at the B-VPLS. When configured, a flush-all-from-me event is triggered upon a SAP or spoke-sdp failure in the I-VPLS.
- **send-bvpls-flush** works as expected when SPB is used at the B-VPLS. Two variants are configurable: all-from-me/all-but-mine. Any I-VPLS SAP/spoke-sdp failure is propagated to the I-VPLS on the peers to flush their respective customer MACs (CMACs). It works only in conjunction with send-flush-on-failure configuration on I-VPLS. The associated ISID list is passed along with the LDP mac-flush message, which is flushed/retained according to the **all-from-me/all-but-me** flag.
- **send-flush-on-bvpls-failure** works as expected when SPB is used at the B-VPLS. A local B-VPLS failure is propagated to the I-VPLS, which then triggers a LDP mac-flush if it has any spoke-sdp on it.
- **propagate-mac-flush-from-bvpls** does not work when SPB is used at the B-VPLS (since failures within the B-VPLS are handled by SPB) and its configuration is blocked.

In the example described later in this section, I-VPLS 32 uses active/standby spoke-sdp resiliency to get connected to the B-VPLS 30 on nodes PE-3 and PE-5.

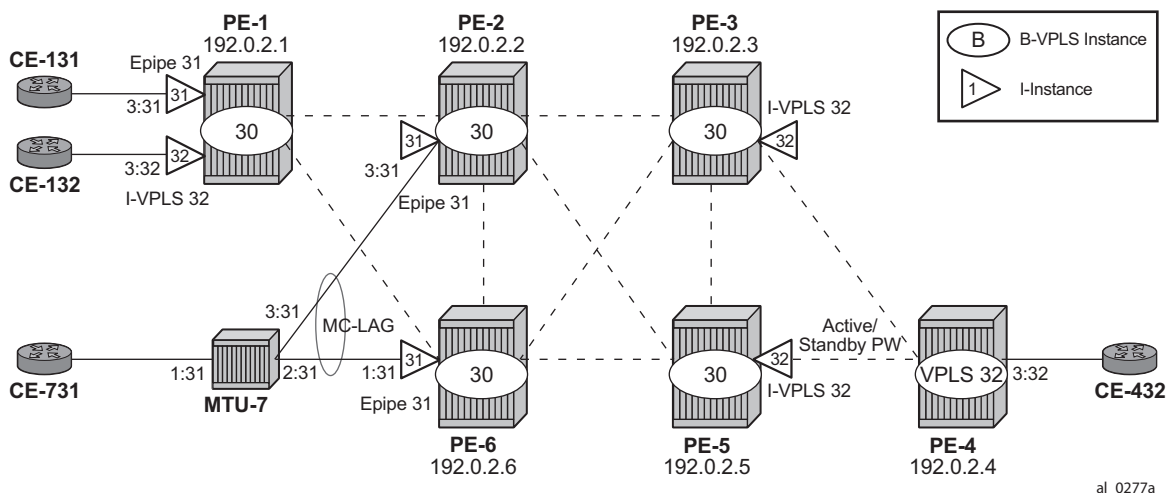


Figure 196: Access Resiliency Test Topology

As an example of MC-LAG connectivity, the Epipe 31 configuration is shown. Just like for regular PBB-VPLS, a sap-bmac is used as source BMAC for the Epipe traffic from PE-2/PE-6 to PE-1. A sap-bmac is a virtual BMAC formed from the configured source-bmac plus the MC-LAG LACP-KEY (if configured this way) and owned by the MC-LAG active chassis. The following CLI output shows the configuration of MC-LAG as well as the generation of the sap-bmac. Once it is properly configured and the MC-LAG and Epipe are up and running, SPB IS-IS will distribute the sap-bmac throughout the B-VPLS, as it does for the system BMACs and OAM vMEP MACs. In this example, PE-2 is the MC-LAG active node, hence the sap-bmac for Epipe 31 is generated from PE-2.

```
*A:PE-2# configure lag 1
mode access
encap-type dot1q
port 1/1/3
lacp active administrative-key 32768
no shutdown
exit

*A:PE-2# configure redundancy
multi-chassis
peer 192.0.2.6 create
mc-lag
lag 1 lacp-key 1 system-id 00:00:00:00:02:06 system-priority 65535
source-bmac-lsb use-lacp-key
no shutdown
exit
no shutdown
exit
```

SPBM Access Resiliency Configuration

```
exit

*A:PE-2# configure service
vpls 30 customer 1 b-vpls create
service-mtu 2000
pbb
    use-sap-bmac
exit
spb 1026 fid 30 create
    level 1
        bridge-priority 0
    exit
    no shutdown
exit
spoke-sdp 21:30 create
    spb create
        no shutdown
    exit
    no shutdown
exit
spoke-sdp 23:30 create
    spb create
        no shutdown
    exit
    no shutdown
exit
spoke-sdp 25:30 create
    spb create
        no shutdown
    exit
    no shutdown
exit
spoke-sdp 26:30 create
    spb create
        no shutdown
    exit
    no shutdown
exit
no shutdown
exit
epipe 31 customer 1 create
pbb
    tunnel 30 backbone-dest-mac "PE-1" isid 31
exit
sap lag-1:31 create
exit
no shutdown
exit
```

```
*A:PE-6# show service id 30 spb fdb
```

```
=====
User service FDB information
```

MAC Addr	UCast Source	State	MCast Source	State
00:00:00:01:01:01	61:30	ok	62:30	ok
00:00:00:02:00:01	62:30	ok	62:30	ok
00:00:00:02:02:02	62:30	ok	62:30	ok


```

00:00:00:03:03:03 63:30          ok      62:30          ok
00:00:00:05:05:05 65:30          ok      62:30          ok
-----
Entries found: 5
=====

```

The configuration for I-VPLS 32 on nodes PE-4 and PE-3 is shown below.

```

*A:PE-4# configure service
      vpls 32 customer 1 create      # Ordinary VPLS, no I-VPLS (no B-VPLS present)
        endpoint "CORE" create
          no suppress-standby-signaling
        exit
      sap 1/1/3:32 create
        exit
      spoke-sdp 43:32 endpoint "CORE" create
        precedence primary
        no shutdown
      exit
      spoke-sdp 45:32 endpoint "CORE" create
        no shutdown
      exit
      no shutdown
    exit

*A:PE-3# configure service
      vpls 30 customer 1 b-vpls create
        service-mtu 2000
        spb 1026 fid 30 create
          no shutdown
        exit
      spoke-sdp 32:30 create
        spb create
        no shutdown
      exit
      no shutdown
    exit
      spoke-sdp 35:30 create
        spb create
        no shutdown
      exit
      no shutdown
    exit
      spoke-sdp 36:30 create
        spb create
        no shutdown
      exit
      no shutdown
    exit
      no shutdown
    exit
  vpls 32 customer 1 i-vpls create
    send-flush-on-failure
    pbb
      backbone-vpls 30
    exit
    send-flush-on-bvpls-failure
    send-bvpls-flush all-from-me

```

SPBM Access Resiliency Configuration

```
exit
spoke-sdp 34:32 create
    no shutdown
exit
no shutdown
exit
```

As discussed, **send-flush-on-failure** and **send-bvpls-flush all-from-me** are configured in the I-VPLS. When the active spoke-sdp goes down on PE-3, a flush-all-from-me message will be propagated through the backbone and will flush the corresponding CMACs associated to the I-VPLS 32 in node PE-1. MAC flush-all-from-me messages are automatically propagated in the core up to the remote I-VPLS 32 on node PE-1 (there is no need for any propagate-mac-flush in the intermediate nodes). Note that the *send-flush-on-bvpls-failure* command works as expected. The command *propagate-mac-flush-from-bvpls* is never used when the B-VPLS is SPB-enabled (the command is not allowed).

Static BMACs and Static ISIDs Configuration

From 11.0R4 onwards, SR OS supports the interworking between SPB-enabled B-VPLS and non-SPB B-VPLS instances. With the addition of this feature, SPB networks can be connected to non-SPB capable nodes, for example third party vendor PBB switches or 7210 SAS nodes. This is possible through the use of conditional static BMACs and static ISIDs on the nodes doing the interworking function. Conditional static BMACs and static ISIDs can be associated to non-SPB B-VPLS SAPs or spoke-sdps.

The following example shows an SPB-enabled B-VPLS (40) on nodes PE-2, PE-6, PE-3 and PE-5. Node PE-4 supports PBB, but not SPB and it is connected by a MC-LAG to nodes PE-3 and PE-5. Services I-VPLS 41 and Epipe 42 have end-points on node PE-4. In this example, nodes PE-3 and PE-5 are acting as interworking nodes. They will be configured with the BMAC of PE-4 so that the MC-LAG active node advertises the non-SPB capable node BMAC into SPB IS-IS. The BMAC will be configured as a conditional static BMAC so that a given SPB node, such as PE-3 or PE-5, will only advertise PE-4's BMAC if its connection to PE-4 is active. Besides the conditional static BMAC, nodes PE-3/PE-5 should advertise the I-VPLS ISIDs defined in PE-4. Note that Epipe ISIDs are not advertised in SPB IS-IS, therefore it is not necessary to create a static ISID for Epipe 42.

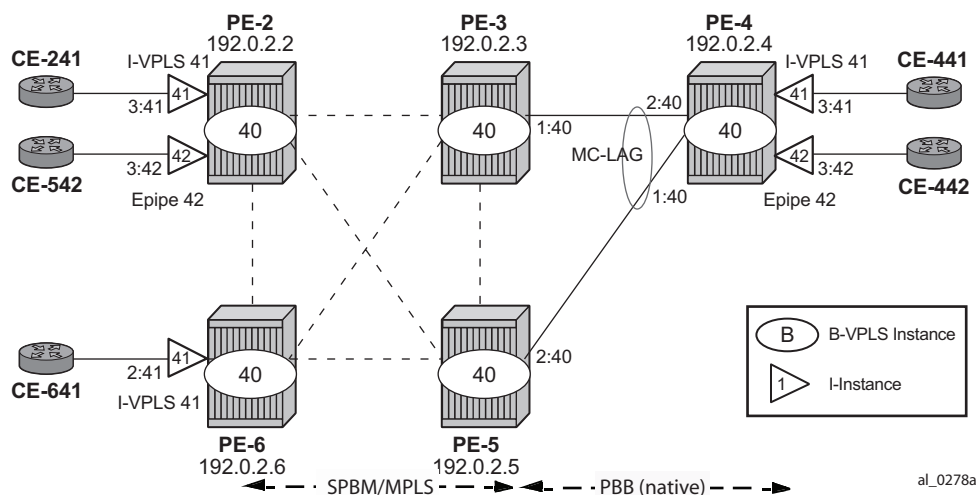


Figure 197: Access Resiliency Test Topology

The commands to configure conditional static BMACs and static ISIDs are shown below.

```
*A:PE-3# configure service vpls 40 static-mac mac
- mac <ieee-address> [create] sap <sap-id> monitor {fwd-status}
- no mac <ieee-address>
- mac <ieee-address> [create] spoke-sdp <sdp-id:vc-id> monitor {fwd-status}
```

Static BMACs and Static ISIDs Configuration

```
*A:PE-3# configure service vpls 40 sap lag-1:40 static-isid range
- no range <range-id>
- range <range-id> isid <isid-value> [to <isid-value>] [create]

<range-id>          : [1..8191]
<isid-value>        : [1..16777215]
<create>            : keyword
```

Note that the **monitor fwd-status** attribute identifies this to be a conditional MAC and is mandatory for static BMACs. This parameter instructs the 7x50 to advertise the BMAC only if the corresponding SAP/spoke-sdp is in forwarding state.

The configuration of the conditional static BMAC and static ISID is shown below.

```
*A:PE-3>config>service # info
-----
...
    vpls 40 customer 1 b-vpls create
        service-mtu 2000
        spb 1027 fid 40 create
            spf-wait 10 10 1000
            no shutdown
        exit
        sap lag-1:40 create
            static-isid
                range 1 create isid 41
            exit
        exit
        spoke-sdp 32:40 create
            spb create
                no shutdown
            exit
            no shutdown
        exit
        spoke-sdp 35:40 create
            spb create
                no shutdown
            exit
            no shutdown
        exit
        spoke-sdp 36:40 create
            spb create
                no shutdown
            exit
            no shutdown
        exit
        static-mac
            mac 00:00:00:04:04:04 create sap lag-1:40 monitor fwd-status
        exit
        no shutdown
    exit
-----
*A:PE-5>config>service# info
...
    vpls 40 customer 1 b-vpls create
        service-mtu 2000
```

```

spb 1027 fid 40 create
    spf-wait 10 10 1000
    no shutdown
exit
sap lag-1:40 create
    static-isid
        range 1 create isid 41
    exit
exit
spoke-sdp 52:40 create
    spb create
        no shutdown
    exit
    no shutdown
exit
spoke-sdp 53:40 create
    spb create
        no shutdown
    exit
    no shutdown
exit
spoke-sdp 56:40 create
    spb create
        no shutdown
    exit
    no shutdown
exit
static-mac
    mac 00:00:00:04:04:04 create sap lag-1:40 monitor fwd-status
exit
    no shutdown
exit

```

Keep in mind that the configuration of the conditional static BMAC is different from the legacy static-mac command, configured within the sap/sdp-binding context. The latter static-mac is not conditional and it is always added to the FDB. The conditional static BMAC is added to the FDB based on the SAP/sdp-binding state (note that the conditional static BMAC is tagged in the FDB as **CStatic**, for Conditional Static).

```

*A:PE-3# show lag 1
=====
Lag Data
=====
Lag-id      Adm      Opr      Weighted Threshold Up-Count MC Act/Stdby
-----
1           up       up       No         0         1       active
=====

```

```

*A:PE-3# show service id 40 fdb pbb
=====
Forwarding Database, b-Vpls Service 40
=====
MAC          Source-Identifier      iVplsMACs  Epipes    Type/Age
-----

```

Static BMACs and Static ISIDs Configuration

```
00:00:00:02:02:02 sdp:32:40          0          0          Spb
00:00:00:04:04:04 sap:lag-1:40       0          0          CStatic
00:00:00:05:05:05 sdp:35:40          0          0          Spb
00:00:00:06:06:06 sdp:36:40          0          0          Spb
=====
```

```
*A:PE-5# show lag 1
```

```
=====
Lag Data
```

```
=====
Lag-id      Adm      Opr      Weighted Threshold Up-Count MC Act/Stdb
-----
1           up       down     No          0          0          standby
=====
```

```
*A:PE-5# show service id 40 fdb pbb
```

```
=====
Forwarding Database, b-Vpls Service 40
```

```
=====
MAC          Source-Identifier      iVplsMACs  Epipes      Type/Age
-----
00:00:00:02:02:02 sdp:52:40          0          0          Spb
00:00:00:03:03:03 sdp:53:40          0          0          Spb
00:00:00:04:04:04 sdp:53:40          0          0          Spb
00:00:00:06:06:06 sdp:56:40          0          0          Spb
=====
```

The static-isid command identifies a set of ISIDs for I-VPLS services that are external to SPBM. These ISIDs are advertised as supported locally on this node unless altered by an isid-policy. Although the example above shows the use of the static-isid associated to a MC-LAG SAP, regular SAPs or spoke SDPs are also supported. ISIDs declared in this way become part of the ISID multicast and consume MFIBs. Multiple SPBM static-isid ranges are allowed under a SAP/spoke SDP. ISIDs are advertised as if they were attached to the local BMAC. Only remote I-VPLS ISIDs need to be defined. In the MFIB, the backbone group MACs are then associated with the active SAP or spoke SDP.

Once the conditional static BMAC for PE-4 and the static-isid 41 (for I-VPLS 41) are configured as discussed, the advertised BMAC and ISID can be checked in the remote SPB nodes:

```
*A:PE-6# show service id 40 spb fdb
```

```
=====
User service FDB information
```

```
=====
MAC Addr      UCast Source      State  MCast Source      State
-----
00:00:00:02:02:02 62:40          ok     62:40          ok
00:00:00:03:03:03 63:40          ok     62:40          ok
00:00:00:04:04:04 63:40          ok     62:40          ok
00:00:00:05:05:05 65:40          ok     62:40          ok
=====
```

```
Entries found: 4
=====
```

```
*A:PE-6# show service id 40 spb mfib
=====
User service MFIB information
=====
MAC Addr          ISID      Status
-----
01:1E:83:00:00:29 41        Ok
-----
Entries found: 1
=====
```

```
*A:PE-6# show service id 40 mfib
=====
Multicast FIB, Service 40
=====
Source Address    Group Address      Sap/Sdp Id          Svc Id   Fwd/Blk
-----
*                 01:1E:83:00:00:29  b-sdp:62:40        Local    Fwd
-----
Number of entries: 1
=====
```

Note that the group address terminates in hex 29, which corresponds to ISID 41.

The configured static-isids can be displayed with the following command (a range 41-100 has been added to the sap lag-1:40 to demonstrate this output):

```
*A:PE-5# configure service
      vpls 40
        sap lag-1:40 create
          static-isid
            range 1 create isid 41 to 100
          exit
        exit
      exit

*A:PE-5# show service id 40 sap lag-1:40 static-isids
=====
Static Isid Entries
=====
Entry      Range
-----
1          41-100
=====
```

Configuration of ISID-Policies in SPB B-VPLS

ISID policies are an optional aspect of SPBM which allow additional control of the advertisement of ISIDs and creation of MFIB entries for I-VPLS (Epipe services do not trigger ISID advertisements or the creation of MFIB entries). By default, if no ISID-policies are used, SPBM automatically advertises and populates MFIB entries for I-VPLS and static-isids. ISID-policies can be used on any SPB-enabled node with locally defined I-VPLS instances or static-isids. The isid-policy parameters are shown below:

```
A:PE-3# configure service vpls 40 isid-policy entry ?
- entry <range-entry-id> [create]
- no entry <range-entry-id>

<range-entry-id>      : [1..8191]
<create>              : keyword

[no] advertise-local - Configure local advertisement of the range
[no] range           - Configure ISID range for the entry
[no] use-def-mcast   - Use default multicast tree to propagate ISID range
```

Where:

- **advertise-local** defines whether the local ISIDs (I-VPLS ISIDs linked to the B-VPLS) or static ISIDs contained in the configured range are advertised in SPBM.
- **use-def-mcast** controls whether the ISIDs contained in the range use MFIB entries (if **no use-def-mcast** is used) or just the default tree which does not use any MFIB entry.

The **isid-policy** becomes active as soon as it is defined, as opposed to other policies in SR OS, which require the policy itself to be applied within the configuration.

The typical use of ISID-policies is to reduce the number of ISIDs being advertised and/or to save MFIB space (in deployments where MFIB space is shared with MMRP and IP Multicast). The use of ISID-policies is recommended for I-VPLS where most of the traffic is unicast or for I-VPLS where the ISID end-points are present in all the Backbone Edge Bridges (BEBs) of the SPB network. In both cases, advertising ISIDs or consuming MFIB entries for those I-VPLSs has little value since no multicast (first case) or the default tree (second case) are as efficient as using MFIB entries.

The following configuration example will use the test topology in [Figure 197](#). In this case, the objective of the isid-policy will be to use the default tree for all the I-VPLS services with ISIDs between 41 and 100, excluding the range 80-90. The following example shows the policy configuration in PE-3. Note that the same policy will be configured in the rest of the SPB nodes, that is, PE-2, PE-6 and PE-5.


```
*A:PE-3# configure service
      vpls 40
        isid-policy
          entry 10 create
            range 80 to 90
          exit
          entry 20 create
            use-def-mcast
            no advertise-local
            range 41 to 79
          exit
          entry 30 create
            use-def-mcast
            no advertise-local
            range 91 to 100
          exit
        exit
      exit
```

Note that the **no advertise-local** option can only be configured if the **use-def-mcast** option is also configured.

```
*A:PE-3# configure service vpls 40 isid-policy entry 40 create no advertise-local
MINOR: SVCNMR #7855 Cannot set AdvLocal for entry - advertise-local or use-def-mcast
option must be specified
```

Overlapping ISID values can be configured as long as the actions are consistent for the same ISID. Conflicting actions are shown in the CLI.

```
*A:PE-3# configure service vpls 40 isid-policy entry 40 create
*A:PE-3>config>service>vpls>isid-policy>entry# range 82 to 85
*A:PE-3>config>service>vpls>isid-policy>entry# use-def-mcast
MINOR: SVCNMR #7854 Cannot set UseDefMctree for entry - Conflicting Actions with Entry-10
```

The isid-policy configured for B-VPLS 40 in all the four nodes makes the SPB network to use the default tree for ISIDs 41-79 and 91-100 and not advertise those ISIDs in SPB IS-IS even if the ISID is locally defined (as in the case for ISIDs 41-100 in PE-3). As discussed in [Basic SPBM Configuration on page 1315](#), the default tree path can be checked from each node by using the **tools dump service id 40 spb default-multicast-list** command.

Due to entry 10 in the policy, ISIDs 80-90 will be advertised by PE-3 (active MC-LAG node). However, nodes PE-2 and PE-6 will not create any MFIB entry for those ISIDs until the corresponding I-VPLS ISIDs are locally created (or configured through static-isids). The following command executed on PE-2 proves that ISIDs 80-90 are indeed being advertised by PE-3:

```
*A:PE-2# show service id 40 spb database detail
=====
Router Base ISIS Instance 1027 Database
=====

Displaying Level 1 database
-----
```

Configuration of ISID-Policies in SPB B-VPLS

```
LSP ID      : PE-2.00-00                      Level      : L1
---snipped---

-----
LSP ID      : PE-3.00-00                      Level      : L1
---snipped---

TLVs :
---snipped---
  MT Capability :
    TLV Len      : 56
    MT ID        : 0
    SPBM Service ID:
    Sub TLV Len   : 52
    BMac Addr     : 00:00:00:03:03:03
    Base VID      : 40
    ISIDs         :
      80          Flags:TR
      81          Flags:TR
      82          Flags:TR
      83          Flags:TR
      84          Flags:TR
      85          Flags:TR
      86          Flags:TR
      87          Flags:TR
      88          Flags:TR
      89          Flags:TR
      90          Flags:TR

-----
LSP ID      : PE-5.00-00                      Level      : L1
---snipped---
=====
```

The **mfib** parameter in the **show service id 40 sap static-isids mfib** command can help understand the state of the MFIB entries added (or not) by the configured static-isid. The following possible states can be shown:

- If the static-isid is configured and programmed in the mfib the status is shown as:
→ ok
- If the static-isid is not configured and not programmed in the mfib, the reasons can be (order of priority):
→ useDefMCTree - isid policy is applied on the service for the isid.
→ sysMFibLimit - system mfib limit has been exceeded
→ addPending - adding pending due to processing delays
- If the static-isid is not configured but present in the mfib:
→ delPending - cleanup pending due to processing delays.

The following examples show some of these possible states:

```
*A:PE-5# show service id 40 sap lag-1:40 static-isids mfib
=====
ISID Detail
=====
ISID          Status
-----
<<snip>>
80             ok
81             ok
<<snip>>
41             useDefMCTree
42             useDefMCTree
<<snip>>
8292           sysMFibLimit
8293           sysMFibLimit
<<snip>>
9253           addPending
9254           addPending
<<snip>>
=====
```

Conclusion

SR OS supports an efficient SPBM implementation in the context of a B-VPLS, where system BMACs, vMEP OAM BMACs and SAP-BMACs are advertised in SPB IS-IS. SPBM provides a simple solution where no other control plane protocol is required in the B-VPLS to take care of the resiliency, load-balancing and multicast optimization. The SPBM implementation in the SR OS provides scale optimization through the use of control and user B-VPLSs, allows the interworking between SPB networks and PBB networks, as well as the optimization of the MFIB resources and advertisement of ISIDs through the use of ISID-policies.

In This Section

This section provides configuration information for the following topics:

- [Carrier Supporting Carrier IP VPNs on page 1347](#)
- [Layer 3 VPN: VPRN Type Spoke on page 1373](#)
- [Multicast in a VPN I on page 1389](#)
- [Multicast in a VPN II on page 1447](#)
- [Multicast VPN: Core Diversity on page 1503](#)
- [Multicast VPN: Inter-AS Option B on page 1533](#)
- [Multicast VPN: Sender-Only, Receiver-Only on page 1559](#)
- [Multicast VPN: Use of Wildcard Selective PMSI on page 1611](#)
- [Source Redundancy in a Multicast VPN on page 1643](#)
- [Spoke Termination for IPv6-6VPE on page 1683](#)
- [VPRN Inter-AS VPN Model C on page 1715](#)

Carrier Supporting Carrier IP VPNs

In This Chapter

This section provides information about carrier supporting carrier IP VPN configurations.

Topics in this section include:

- [Applicability on page 1348](#)
- [Overview on page 1349](#)
- [Configuration on page 1351](#)
- [Conclusion on page 1372](#)

Applicability

This example is applicable to the following platforms: 7950 XRS, 7750 SR-7/12, 7450 ESS-6/7/12 and 7450 SR-c4/c12.

When a 7450 operating in mixed-mode, a 7750, or a 7950 is deployed as a CSC-PE (refer to [Figure 198](#)) all its network interfaces and all its CSC VPRN interfaces must be configured on FP2 or higher hardware.

The configuration in this guide was tested with release 12.0.R1.

Overview

Carrier Supporting Carrier (CSC) is a solution that allows one service provider (the Customer Carrier) to use the IP VPN service of another service provider (the Super Carrier) for some or all of its backbone transport. RFC 4364 defines a Carrier Supporting Carrier solution for BGP/MPLS IP VPNs that uses MPLS at the interconnection points between the two service providers to provide a scalable and secure solution.

A simplified CSC network topology is shown in [Figure 198](#). A CSC deployment involves the following types of devices:

CE — Customer premises equipment dedicated to one particular business/enterprise.

PE — Edge router managed and operated by the Customer Carrier that connects to CEs to provide business VPN or Internet services.

CSC-CE — Peering router managed and operated by the Customer Carrier that is connected to CSC-PEs for purposes of using the associated CSC IP VPN services for backbone transport. The CSC-CE may attach directly to CEs if it is also configured to be a PE for business VPN services.

CSC-PE — A PE router managed and operated by the Super Carrier that supports one or more CSC IP VPN services possibly in addition to other traditional PE services.

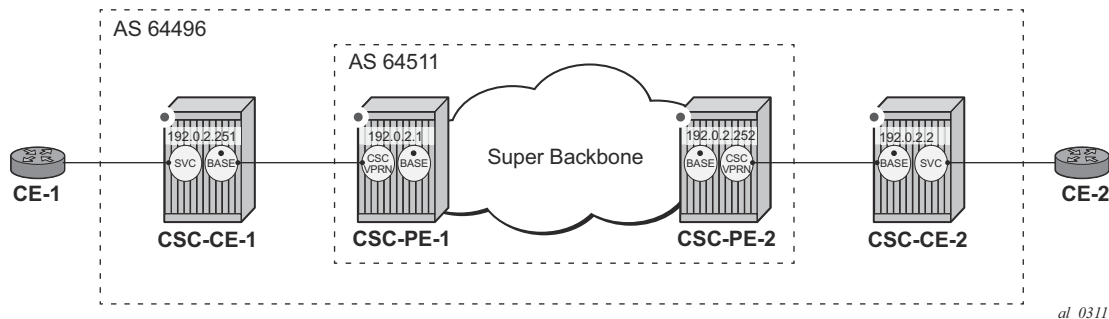


Figure 198: CSC Network Topology

In the CSC solution the CSC-CE and CSC-PE are directly connected by a link that supports MPLS. The CSC-CE distributes an MPLS label for every /32 IPv4 prefix it and any downstream PE uses as a BGP next-hop in routes associated with services offered by the Customer Carrier. Note that BGP **must be used** as the label distribution protocol between CSC-CE and CSC-PE if the latter device is a 7x50. Typically the Customer Carrier and Super Carrier operate as two different Autonomous Systems (AS) and therefore BGP, more specifically EBGP, is the best label distribution protocol even if other options are available. The BGP session between CSC-CE and CSC-PE must be single-hop EBGP (or IBGP) if either device is a 7x50.

In a 7x50 CSC-PE the interface to a CSC-CE is a special type of IP/MPLS interface that belongs to a VPRN configured for CSC mode. This special type of interface is called a CSC VPRN interface throughout the remainder of this example. The CSC VPRN interface has many of the same characteristics as a network interface of the base router but its association with a VRF ensures that the traffic and control plane routes of the Customer Carrier are kept separate from other services.

When a 7x50 CSC-PE receives a labelled-IPv4 route (with label L1, next-hop N1) from a CSC-CE BGP peer the following actions take place in the CSC-PE:

1. The BGP route is installed into the routing table of the CSC VPRN (assuming the BGP route is the best route to the destination).
2. If the BGP route matches the VRF export policy it is advertised to core MP-BGP peers as a VPN-IPv4 route. The advertised label value is changed to L2.
3. BGP programs the line cards with an MPLS forwarding entry that swaps L2 for L1 and sends the MPLS packet over the CSC VPRN interface associated with next-hop N1.

When a 7x50 CSC-PE receives a VPN-IPv4 route (with label L2, next-hop N2) the following actions take place in the CSC-PE:

1. If the VPN-IPv4 route matches the VRF import policy of a CSC VPRN it is installed into the routing table of that CSC VPRN.
2. If the imported (BGP-VPN) route matches the BGP export policy associated with a CSC-CE BGP peer it is advertised to that peer as a labelled-IPv4 route. The advertised label value is changed to L3.
3. BGP programs the line cards with an MPLS forwarding entry that swaps L3 for L2 and sends the packet inside the MPLS tunnel to next-hop N2.

Once a CSC-CE has learned a labelled-IPv4 route for a remote CSC-CE and vice versa the two CSC-CEs can setup a BGP session between themselves and exchange VPN routes over this session if they are both PEs with services. Typically this BGP session will be an IBGP session because the local and remote CSC-CEs belong to the same Autonomous System (AS). The Layer 2 VPN and Layer 3 VPN routes exchanged by the CSC-CEs are resolved by the labelled-IPv4 routes they have for each other's /32 IPv4 address.

Configuration

This section will walk through the steps to configure the CSC solution shown in [Figure 198](#). Note that the IPv4 addresses in [Figure 198](#) are the system IP addresses of the routers.

Step 1. Configure CSC-CE-1.

This example assumes that CSC-CE-1 is a PE router with Layer 2 and Layer 3 VPN services that must extend across the CSC VPN service; assume that there are no further downstream PEs in AS 64496. The configuration of one such Layer 3 VPN service in CSC-CE-1 is shown below:

```
A:csc-ce-1>config>service>vpn# info
-----
route-distinguisher 64496:1
auto-bind mpls
vrf-target target:64496:1
...
no shutdown
-----
A:csc-ce-1>config>service>vpn#
```

For brevity the above configuration sample omits commands related to SAP IP interfaces, spoke-SDP IP interfaces, PE-CE routing protocols, QoS, IP filters, etc.

The base routing instance of the CSC-CE should be configured with the appropriate router-ID and autonomous-system number and the system interface should be given an IPv4 address (usually the same as the router-id). The interface to CSC-PE-1 should then be created and configured. The base router configuration of CSC-CE-1 is shown below:

```
*A:csc-ce-1>config>router# info
-----
#-----
echo "IP Configuration"
#-----
interface "int-csc-ce-1-to-csc-pe-1"
address 192.168.0.1/30
port 1/1/2
no shutdown
exit
interface "system"
address 192.0.2.1/32
no shutdown
exit
autonomous-system 64496
router-id 192.0.2.1
-----
*A:csc-ce-1>config>router#
```

BGP should be configured as the control plane protocol running on the interface to CSC-PE-1, as shown below:

```
*A:csc-ce-1>config>router>bgp# info
-----
      group "csc-pe"
        peer-as 64511
        neighbor 192.168.0.2
          family ipv4
            export "static-to-bgp"
            advertise-label ipv4
            split-horizon
        exit
      exit
    no shutdown
-----
*A:csc-ce-1>config>router>bgp#
```

Note the following about the BGP configuration of CSC-CE-1:

- The peer type is EBGp (**peer-as** is different from the locally configured **autonomous-system**)
- The transport for the EBGp session is IPv4 (the **neighbor** address is an IPv4 address)
- The **advertise-label ipv4** command causes MP-BGP negotiation of the address family for AFI=1 and SAFI=4 (IPv4 NLRI with MPLS labels), as can be observed from the following debug trace (using the command **debug router bgp open**) of the OPEN message from CSC-CE-1.

```
2 2014/04/01 08:35:44.15 EST MINOR: DEBUG #2001 Base BGP
"BGP: OPEN
Peer 1: 192.168.0.2 - Received BGP OPEN: Version 4
  AS Num 64511: Holdtime 90: BGP_ID 192.0.2.251: Opt Length 16
  Opt Para: Type CAPABILITY: Length = 14: Data:
    Cap_Code MP-BGP: Length 4
      Bytes: 0x0 0x1 0x0 0x4
    Cap_Code ROUTE-REFRESH: Length 0
    Cap_Code 4-OCTET-ASN: Length 4
      Bytes: 0x0 0x0 0xfb 0xff
"
```

- The **split-horizon** command is optional. It prevents a best BGP route from the CSC-PE peer from being re-advertised back to that peer.
- The **export** command applies a BGP export policy to the session. The configuration of the policy is shown below:

```
*A:csc-ce-1>config>router>policy-options# info
-----
      prefix-list "system-ip"
        prefix 192.0.2.1/32 exact
      exit
    policy-statement "static-to-bgp"
      entry 10
-----
```

```

        from
            protocol direct
            prefix-list "system-ip"
        exit
        action accept
        exit
    exit
    default-action reject
exit
-----
*A:csc-ce-1>config>router>policy-options#

```

The effect of the BGP export policy is to advertise the system IP address of CSC-CE-1 as a labelled-IPv4 BGP route towards the CSC-PE(s).

Step 2. Configure CSC service on CSC-PE-1.

CSC-PE-1 must be configured with a VPRN in **carrier-carrier-vpn** mode in order to provide CSC service to CSC-CE-1. The entire configuration of the VPRN is shown below:

```

A:csc-pe-1>config>service>vprn# info
-----
carrier-carrier-vpn
router-id 192.0.2.251
autonomous-system 64511
route-distinguisher 64511:1
auto-bind mpls
vrf-target target:64511:1
network-interface "csc-pe-1-to-csc-ce-1" create
    address 192.168.0.2/30
    port 1/1/1
    no shutdown
exit
bgp
    group "csc-ce"
        as-override
        export "bgp-vpn-routes"
        peer-as 64496
        neighbor 192.168.0.1
            family ipv4
            advertise-label ipv4
            split-horizon
        exit
    exit
    no shutdown
exit
no shutdown
-----
A:csc-pe-1>config>service>vprn#

```

Note the following about the VPRN configuration of CSC-PE-1:

- The **carrier-carrier-vpn** command is mandatory. It cannot be configured if the VPRN currently has any SAP or spoke-SDP access interfaces configured; they must first be shutdown if necessary and then deleted.

```
*A:csc-pe-1>config>service>vpn# carrier-carrier-vpn
INFO: PIP #1195 Cannot toggle carrier-carrier-vpn - service interfaces present
*A:csc-pe-1>config>service>vpn#
```

- The **auto-bind** command should be set appropriately for the type of transport desired to other CSC-PEs, but note that GRE is not supported.

```
A:csc-pe-1>config>service>vpn# auto-bind gre
MINOR: SVCMMGR #1538 auto-bind config not supported - carrier-carrier vpn
A:csc-pe-1>config>service>vpn#
```

- The interface to CSC-CE-1 must be a **network-interface**. A **network-interface** can be associated with an entire Ethernet port (as shown in the example above), a VLAN sub-interface of an Ethernet port, an entire LAG or a VLAN sub-interface of a LAG. In all cases the associated Ethernet ports must be configured in network or hybrid mode and must reside on FP2 or higher based cards/systems.

Note the following about the BGP configuration of the CSC VPRN service in CSC-PE-1:

- The peer type is EBGp (**peer-as** is different from the locally configured **autonomous-system**).
- The transport for the EBGp session is IPv4 (the **neighbor** address is an IPv4 address).
- The **advertise-label ipv4** command causes MP-BGP negotiation of the address family for AFI=1 and SAFI=4 (IPv4 NLRI with MPLS labels).
- The **split-horizon** command is optional. It prevents a best BGP route from the CSC-CE peer from being re-advertised back to that peer.
- The **as-override** command replaces CSC-CE-1's AS number (64496) with CSC-PE-1's AS number (64511) in the AS_PATH attribute of routes advertised to CSC-CE-1. Without this configuration CSC-CE-1 may reject routes originated by CSC-CE-2 as invalid due to an AS-path loop.
- The **export** command applies a BGP export policy to the session. The configuration of the policy is shown below:

```
*A:csc-pe-1>config>router>policy-options# info
-----
      policy-statement "bgp-vpn-routes"
        entry 10
          from
            protocol bgp-vpn
          exit
          action accept
          exit
```

```

        exit
        default-action reject
    exit
-----
*A:csc-pe-1>config>router>policy-options#

```

The effect of the BGP export policy is to re-advertise VPN-IPv4 routes imported into the CSC VPRN (and used for forwarding) to CSC-CE-1.

Step 3. Verify exchange of routes between CSC-CE-1 and CSC-PE-1.

When Steps 1 and 2 have been completed properly CSC-CE-1 should now be advertising the labelled-IPv4 route for its system IP address to CSC-PE-1. This can be checked from the perspective of CSC-CE-1 as shown below:

```

*A:csc-ce-1# show router bgp routes 192.0.2.1/32 hunt
=====
BGP Router ID:192.0.2.1      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP IPv4 Routes
=====
RIB In Entries
-----
RIB Out Entries
-----
Network       : 192.0.2.1/32
Nexthop       : 192.168.0.1
Path Id       : None
To            : 192.168.0.2
Res. Nexthop  : n/a
Local Pref.   : n/a
Aggregator AS : None
Atomic Aggr.  : Not Atomic
AIGP Metric   : None
Connector     : None
Community     : No Community Members
Cluster       : No Cluster Members
Originator Id : None
IPv4 Label    : 262142
Origin        : IGP
AS-Path       : 64496
Neighbor-AS   : 64496
-----
Routes : 1
=====
*A:csc-ce-1#

```

Note that CSC-CE-1 has advertised a label value of 262142 with the prefix.

The following output shows the received route from the perspective of CSC-PE-1:

```
*A:csc-pe-1# show router 1 bgp routes 192.0.2.1/32 hunt
=====
BGP Router ID:192.0.2.251      AS:64511      Local AS:64511
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP IPv4 Routes
=====
-----
RIB In Entries
-----
Network      : 192.0.2.1/32
Nexthop      : 192.168.0.1
Path Id      : None
From         : 192.168.0.1
Res. Nexthop : 192.168.0.1
Local Pref.  : None
Aggregator AS : None
Atomic Aggr. : Not Atomic
AIGP Metric  : None
Connector    : None
Community    : No Community Members
Cluster      : No Cluster Members
Originator Id : None
Fwd Class    : None
IPv4 Label   : 262142
Flags        : Used Valid Best IGP
Route Source : External
AS-Path      : 64496
Neighbor-AS  : 64496
Interface Name : csc-pe-1-to-csc-ce-1
Aggregator    : None
MED           : None
Peer Router Id : 192.0.2.1
Priority      : None
-----
RIB Out Entries
-----
-----
Routes : 1
=====
*A:csc-pe-1#
```


Step 4. Configure core connectivity for CSC-PE-1.

The next step is to configure the base router instance of CSC-PE-1 so that it can exchange VPN-IPv4 routes with CSC-PE-2 (and potentially other CSC-PEs). At a minimum this requires:

- Router-id and autonomous-system configuration.
- Network interface creation and configuration, including assignment of an IPv4 address to the system interface.
- Configuration of the IGP protocol. In this example IS-IS is used.
- Configuration of the LDP protocol (optional).
- Configuration of RSVP LSPs used to reach remote CSC-PE devices (optional).
- Configuration of the BGP protocol.

These elements of the base router configuration of CSC-PE-1 are shown below:

```
*A:csc-pe-1>config>router# info
-----
#-----
echo "IP Configuration"
#-----
        interface "csc-pe-1-to-csc-pe-2"
            address 192.168.1.1/30
            port 1/1/2
            no shutdown
        exit
        interface "system"
            address 192.0.2.251/32
            no shutdown
        exit
        autonomous-system 64511
        router-id 192.0.2.251
#-----
echo "ISIS Configuration"
#-----
        isis
            level-capability level-2
            area-id 49.01
            level 2
                wide-metrics-only
            exit
            interface "system"
                level-capability level-2
                passive
                no shutdown
            exit
            interface "csc-pe-1-to-csc-pe-2"
                level-capability level-2
                interface-type point-to-point
                level 2
                    metric 100
                exit
                no shutdown
```

Configuration

```
        exit
        no shutdown
    exit
#-----
echo "LDP Configuration"
#-----
    ldp
        interface-parameters
            interface "csc-pe-1-to-csc-pe-2"
            exit
        exit
        targeted-session
        exit
        no shutdown
    exit
#-----
echo "BGP Configuration"
#-----
    bgp
        group "core"
            peer-as 64511
            neighbor 192.0.2.252
                family vpn-ipv4
            exit
        exit
        no shutdown
    exit
-----
*A:csc-pe-1>config>router#
```

Note the following about the BGP configuration of the base router in CSC-PE-1:

- The peer type is IBGP (**peer-as** is the same as the locally configured **autonomous-system**).
- The transport for the IBGP session is IPv4 (the **neighbor** address is an IPv4 address).
- The **family vpn-ipv4** command causes MP-BGP negotiation of the address family for AFI=1 and SAFI=128, as can be observed from the following debug trace of the OPEN message from CSC-PE-1.

```
12 2014/04/01 09:34:48.64 EST MINOR: DEBUG #2001 Base BGP
"BGP: OPEN
Peer 1: 192.0.2.252 - Send (Active) BGP OPEN: Version 4
  AS Num 64511: Holdtime 90: BGP_ID 192.0.2.251: Opt Length 16
  Opt Para: Type CAPABILITY: Length = 14: Data:
    Cap_Code MP-BGP: Length 4
    Bytes: 0x0 0x1 0x0 0x80
    Cap_Code ROUTE-REFRESH: Length 0
    Cap_Code 4-OCTET-ASN: Length 4
    Bytes: 0x0 0x0 0xfb 0xff
"
```

Step 5. Configure core connectivity for CSC-PE-2

The next step is to configure the base router instance of CSC-PE-2 so that it can exchange VPN-IPv4 routes with CSC-PE-1 (and potentially other CSC-PEs). At a minimum this requires:

- Router-id and autonomous-system configuration.
- Network interface creation and configuration, including assignment of an IPv4 address to the system interface.
- Configuration of the IGP protocol. In this example IS-IS is used.
- Configuration of the LDP protocol (optional).
- Configuration of RSVP LSPs used to reach remote CSC-PE devices (optional).
- Configuration of the BGP protocol.

These elements of the base router configuration of CSC-PE-2 are shown below:

```
A:csc-pe-2>config>router# info
-----
#-----
echo "IP Configuration"
#-----
        interface "csc-pe-2-to-csc-pe-1"
            address 192.168.1.2/30
            port 1/1/2
            no shutdown
        exit
        interface "system"
            address 192.0.2.252/32
            no shutdown
        exit
        autonomous-system 64511
        router-id 192.0.2.252
#-----
echo "ISIS Configuration"
#-----
        isis
            level-capability level-2
            area-id 49.01
            level 2
                wide-metrics-only
            exit
            interface "system"
                level-capability level-2
                passive
                no shutdown
            exit
            interface "csc-pe-2-to-csc-pe-1"
                level-capability level-2
                interface-type point-to-point
                level 2
                    metric 100
                exit
                no shutdown
```

Configuration

```
        exit
        no shutdown
    exit
#-----
echo "LDP Configuration"
#-----
    ldp
        interface-parameters
            interface "csc-pe-2-to-csc-pe-1"
            exit
        exit
        targeted-session
        exit
        no shutdown
    exit
#-----
echo "BGP Configuration"
#-----
    bgp
        group "core"
            cluster 192.0.2.252
            peer-as 64511
            neighbor 192.0.2.251
                family vpn-ipv4
                split-horizon
            exit
        exit
        no shutdown
    exit
-----
A:csc-pe-2>config>router#
```

Note the following about the BGP configuration of the base router in CSC-PE-2:

- The peer type is IBGP (**peer-as** is the same as the locally configured **autonomous-system**).
- The transport for the IBGP session is IPv4 (the **neighbor** address is an IPv4 address).
- The **family vpn-ipv4** command causes MP-BGP negotiation of the address family for AFI=1 and SAFI=128.
- The **cluster** command configures CSC-PE-2 as a route reflector for clients in the BGP group called “core”. This is not required and in a more typical deployment the route reflector would be a separate router from any CSC-PE.

Step 6. Configure CSC service on CSC-PE-2.

CSC-PE-2 must be configured with a VPRN in **carrier-carrier-vpn** mode in order to provide CSC service to CSC-CE-2. The entire configuration of the VPRN is shown below:

```
A:csc-pe-2>config>service>vprn# info
-----
carrier-carrier-vpn
router-id 192.0.2.252
autonomous-system 64511
route-distinguisher 64511:2
auto-bind mpls
vrf-target target:64511:1
network-interface "csc-pe-2-to-csc-ce-2" create
    address 192.168.2.1/30
    port 1/1/3
    no shutdown
exit
bgp
    group "csc-ce"
        as-override
        export "bgp-vpn-routes"
        peer-as 64496
        neighbor 192.168.2.2
            family ipv4
            advertise-label ipv4
            split-horizon
        exit
    exit
    no shutdown
exit
no shutdown
-----
A:csc-pe-2>config>service>vprn#
```

Note the following about the VPRN configuration of CSC-PE-2:

- The **carrier-carrier-vpn** command is mandatory. It cannot be configured if the VPRN currently has any SAP or spoke-SDP “access” interfaces configured; they must first be shutdown if necessary and then deleted.
- The **auto-bind** command should be set appropriately for the type of transport desired to other CSC-PEs, but note that GRE is not supported.
- The interface to CSC-CE-2 must be a **network-interface**. A **network-interface** can be associated with an entire Ethernet port (as shown in the example above), a VLAN sub-interface of an Ethernet port, an entire LAG or a VLAN sub-interface of a LAG. In all cases the associated Ethernet ports must be configured in network or hybrid mode and must reside on FP2 or higher based cards/systems.

Note the following about the BGP configuration of the CSC VPRN service in CSC-PE-2:

- The peer type is EBGp (**peer-as** is different from the locally configured **autonomous-system**).
- The transport for the EBGp session is IPv4 (the **neighbor** address is an IPv4 address).
- The **advertise-label ipv4** command causes MP-BGP negotiation of the address family for AFI=1 and SAFI=4 (IPv4 NLRI with MPLS labels).
- The **split-horizon** command is optional. It prevents a best BGP route from the CSC-CE peer from being re-advertised back to that peer.
- The **as-override** command replaces CSC-CE-2's AS number (64496) with CSC-PE-2's AS number (64511) in the AS_PATH attribute of routes advertised to CSC-CE-2. Without this configuration CSC-CE-2 may reject routes originated by CSC-CE-1 as invalid due to an AS-path loop.
- The **export** command applies a BGP export policy to the session. The configuration of the policy is shown below:

```
*A:csc-pe-2>config>router>policy-options# info
-----
      policy-statement "bgp-vpn-routes"
        entry 10
          from
            protocol bgp-vpn
          exit
          action accept
          exit
        exit
      default-action reject
    exit
-----
*A:csc-pe-2>config>router>policy-options#
```

The effect of the BGP export policy is to re-advertise VPN-IPv4 routes imported into the CSC VPRN (and used for forwarding) to CSC-CE-2.

Step 7. Verify exchange of routes between CSC-PE-1 and CSC-PE-2.

When the preceding steps have been completed properly CSC-PE-1 should now be advertising the labelled-IPv4 route for 192.0.2.1/32 (the system IP address of CSC-CE-1) to CSC-PE-2. This can be checked from the perspective of CSC-PE-1 as shown below:

```
*A:csc-pe-1# show router bgp routes vpn-ipv4 192.0.2.1/32 hunt
=====
BGP Router ID:192.0.2.251      AS:64511      Local AS:64511
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP VPN-IPv4 Routes
=====
RIB In Entries
-----
RIB Out Entries
-----
Network      : 192.0.2.1/32
Nexthop      : 192.0.2.251
Route Dist.  : 64511:1          VPN Label      : 262140
Path Id      : None
To           : 192.0.2.252
Res. Nexthop : n/a
Local Pref.  : 100
Aggregator AS : None          Interface Name : NotAvailable
Atomic Aggr. : Not Atomic     Aggregator    : None
AIGP Metric  : None          MED           : None
Connector    : None
Community    : target:64511:1
Cluster      : No Cluster Members
Originator Id : None          Peer Router Id : 192.0.2.252
Origin       : IGP
AS-Path      : 64496
Neighbor-AS  : 64496
-----
Routes : 1
=====
*A:csc-pe-1#
```

Note that CSC-PE-1 has advertised a label value of 262140 with the prefix.

The following output shows the received route from the perspective of CSC-PE-2:

```
A:csc-pe-2# show router bgp routes vpn-ipv4 192.0.2.1/32 hunt
=====
BGP Router ID:192.0.2.252      AS:64511      Local AS:64511
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
```

Configuration

```
=====
BGP VPN-IPv4 Routes
=====
-----
RIB In Entries
-----
Network      : 192.0.2.1/32
Nexthop      : 192.0.2.251
Route Dist.  : 64511:1          VPN Label      : 262140
Path Id      : None
From         : 192.0.2.251
Res. Nexthop : n/a
Local Pref.  : 100              Interface Name : csc-pe-2-to-csc-pe-1
Aggregator AS : None            Aggregator    : None
Atomic Aggr. : Not Atomic       MED            : None
AIGP Metric  : None
Connector    : None
Community    : target:64511:1
Cluster      : No Cluster Members
Originator Id : None            Peer Router Id : 192.0.2.251
Fwd Class    : None             Priority       : None
Flags        : Used Valid Best IGP
Route Source : Internal
AS-Path      : 64496
Neighbor-AS  : 64496
VPRN Imported : 1
-----
RIB Out Entries
-----
-----
Routes : 1
=====
A:csc-pe-2#
```

Also note the label swap entries that BGP programmed in the line cards of CSC-PE-1 based on the received labelled-IPv4 route from CSC-CE-1 (Label Origin = ExtCarCarVpn) and the advertised VPN-IPv4 route to CSC-PE-2:

```
*A:csc-pe-1# show router bgp inter-as-label
=====
BGP Inter-AS labels
=====
NextHop      Received      Advertised      Label
              Label          Label           Origin
-----
192.168.0.1  262142          262140          ExtCarCarVpn
-----
Total Labels allocated: 1
=====
*A:csc-pe-1#
```


Step 8. Configure CSC-CE-2.

This example assumes that CSC-CE-2 is a PE router with Layer 2 and Layer 3 VPN services that must extend across the CSC VPN service. The configuration of one such Layer 3 VPN service in CSC-CE-2 is shown below:

```
A:csc-ce-2>config>service>vpn# info
-----
route-distinguisher 64496:2
auto-bind mpls
vrf-target target:64496:1
...
no shutdown
-----
A:csc-ce-2>config>service>vpn#
```

For brevity, the above configuration sample omits commands related to SAP IP interfaces, spoke-SDP IP interfaces, PE-CE routing protocols, QoS, IP filters, etc.

The base routing instance of CSC-CE-2 should be configured with the appropriate router-ID and autonomous-system number and the system interface should be given an IPv4 address (usually the same as the router-id). The interface to CSC-PE-2 should then be created and configured. The base router configuration of CSC-CE-2 is shown below:

```
A:csc-ce-2>config>router# info
-----
#-----
echo "IP Configuration"
#-----
interface "int-csc-ce-2-to-csc-pe-2"
address 192.168.2.2/30
port 1/1/2
no shutdown
exit
interface "system"
address 192.0.2.2/32
no shutdown
exit
autonomous-system 64496
router-id 192.0.2.2
-----
A:csc-ce-2>config>router#
```

BGP should be configured as the control plane protocol running on the interface to CSC-PE-2 as shown below:

```
A:csc-ce-2>config>router>bgp# info
-----
group "csc-pe"
family ipv4
peer-as 64511
neighbor 192.168.2.1
```

Configuration

```
        family ipv4
        export "static-to-bgp"
        advertise-label ipv4
        split-horizon
    exit
exit
no shutdown
-----
A:csc-ce-2>config>router>bgp#
```

Note the following about the BGP configuration of CSC-CE-2:

- The peer type is EBGP (**peer-as** is different from the locally configured **autonomous-system**).
- The transport for the EBGP session is IPv4 (the **neighbor** address is an IPv4 address).
- The **advertise-label ipv4** command causes MP-BGP negotiation of the address family for AFI=1 and SAFI=4 (IPv4 NLRI with MPLS labels).
- The **split-horizon** command is optional. It prevents a best BGP route from the CSC-PE peer from being re-advertised back to that peer.
- The **export** command applies a BGP export policy to the session. The configuration of the policy is shown below:

```
A:csc-ce-2>config>router>policy-options# info
-----
    prefix-list "system-ip"
        prefix 192.0.2.2/32 exact
    exit
    policy-statement "static-to-bgp"
        entry 10
            from
                protocol direct
                prefix-list "system-ip"
            exit
            action accept
            exit
        exit
        default-action reject
    exit
-----
A:csc-ce-2>config>router>policy-options#
```

The effect of the BGP export policy is to advertise the system IP address of CSC-CE-2 as a labelled-IPv4 BGP route towards CSC-PE-2.

Step 9. Verify exchange of routes between CSC-PE-2 and CSC-CE-2.

When the preceding steps have been completed properly CSC-PE-2 should now be advertising the labelled-IPv4 route for 192.0.2.1/32 to CSC-CE-2. This can be checked from the perspective of CSC-PE-2 as shown below:

```
A:csc-pe-2# show router 1 bgp routes ipv4 192.0.2.1/32 hunt
=====
BGP Router ID:192.0.2.252      AS:64511      Local AS:64511
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP IPv4 Routes
=====
RIB In Entries
-----
RIB Out Entries
-----
Network      : 192.0.2.1/32
Nexthop      : 192.168.2.1
Path Id      : None
To           : 192.168.2.2
Res. Nexthop : n/a
Local Pref.  : n/a
Aggregator AS : None
Atomic Aggr. : Not Atomic
AIGP Metric  : None
Connector    : None
Community    : target:64511:1
Cluster      : No Cluster Members
Originator Id : None
IPv4 Label   : 262139
Origin       : IGP
AS-Path      : 64511 64511
Neighbor-AS  : 64511
-----
Routes : 1
=====
A:csc-pe-2#
```

Note that CSC-PE-2 has advertised a label value of 262139 with the prefix.

The following output shows the received route from the perspective of CSC-CE-2:

```
A:csc-ce-2# show router bgp routes ipv4 192.0.2.1/32 hunt
=====
BGP Router ID:192.0.2.2      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
```

```
=====
BGP IPv4 Routes
=====
-----
RIB In Entries
-----
Network      : 192.0.2.1/32
Nexthop      : 192.168.2.1
Path Id      : None
From         : 192.168.2.1
Res. Nexthop : 192.168.2.1
Local Pref.  : None
Aggregator AS : None
Atomic Aggr. : Not Atomic
AIGP Metric  : None
Connector    : None
Community    : target:64511:1
Cluster      : No Cluster Members
Originator Id : None
Fwd Class    : None
IPv4 Label    : 262139
Flags        : Used Valid Best IGP
Route Source  : External
AS-Path       : 64511 64511
Neighbor-AS   : 64511
Interface Name : int-csc-ce-2-to-csc-p*
Aggregator     : None
MED            : None
Peer Router Id : 192.0.2.252
Priority        : None
-----
RIB Out Entries
-----
-----
Routes : 1
=====
* indicates that the corresponding row element may have been truncated.
A:csc-ce-2#
```

Also note the label swap entries that BGP programmed in the line cards of CSC-PE-2 based on the received VPN-IPv4 routes from CSC-PE-1 (Label Origin = Internal) and the advertised labelled-IPv4 routes to CSC-CE-2:

```
A:csc-pe-2# show router 1 bgp inter-as-label
=====
BGP Inter-AS labels
=====
NextHop      Received      Advertised      Label
              Label         Label           Origin
-----
192.0.2.251   262140             262139          Internal
192.0.2.251   262142             262138          Internal
-----
Total Labels allocated: 2
=====
A:csc-pe-2#
```

In the above output the first entry for NextHop 192.0.2.251 corresponds to the prefix 192.0.2.1/32; recall from Step 7 that CSC-PE-2 received the VPN-IPv4 route with label value 262140 and it can be seen from this step that it re-advertised the route to CSC-CE-2 with label value 262139.

Step 10. Setup BGP session between CSC-CE-1 AND CSC-CE-2.

The final step in the setup of the CSC solution shown in Figure 1 is the creation of a BGP session between CSC-CE-1 and CSC-CE-2 so that they can exchange routes belonging to VPN services they support. The configuration of this BGP session from the perspective of CSC-CE-1 is shown below:

```
*A:csc-ce-1>config>router>bgp# info
-----
      group "csc-ce"
        peer-as 64496
        neighbor 192.0.2.2
          family vpn-ipv4
        exit
      exit
    no shutdown
-----
*A:csc-ce-1>config>router>bgp#
```

The configuration of the BGP session from the perspective of CSC-CE-2 is very similar, as shown below.

```
A:csc-ce-2>config>router>bgp# info
-----
      group "csc-ce"
        peer-as 64496
        neighbor 192.0.2.1
          family vpn-ipv4
        exit
      exit
    no shutdown
-----
A:csc-ce-2>config>router>bgp#
```

Note the following about the configuration of the BGP session between CSC-CE-1 and CSC-CE-2:

- The peer type is IBGP (**peer-as** is the same as the locally configured **autonomous-system**).
- The transport for the IBGP session is IPv4 (the **neighbor** address is an IPv4 address).
- The **family vpn-ipv4** command causes MP-BGP negotiation of the address family for AFI=1 and SAFI=128.

Step 11. Verify exchange of routes between CSC-CE-1 and CSC-CE-2.

When the preceding steps have been completed properly CSC-PE-2 should now be able to advertise a VPN-IPv4 route for some IP prefix (for example 10.14.30.0/24) to CSC-CE-2. This can be checked from the perspective of CSC-CE-2 as shown below:

```
A:csc-ce-2# show router bgp routes vpn-ipv4 10.14.30.0/24 hunt
=====
BGP Router ID:192.0.2.2          AS:64496          Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP VPN-IPv4 Routes
=====
RIB In Entries
-----
Network      : 10.14.30.0/24
Nexthop      : 192.0.2.1
Route Dist.  : 64496:1          VPN Label      : 262143
Path Id      : None
From         : 192.0.2.1
Res. Nexthop : n/a
Local Pref.  : 100              Interface Name : NotAvailable
Aggregator AS : None           Aggregator     : None
Atomic Aggr. : Not Atomic      MED            : None
AIGP Metric  : None
Connector    : None
Community    : target:64496:1
Cluster      : No Cluster Members
Originator Id : None           Peer Router Id  : 192.0.2.1
Fwd Class    : None           Priority        : None
Flags        : Used Valid Best IGP
Route Source  : Internal
AS-Path       : No As-Path
Neighbor-AS   : N/A
VPRN Imported : 1
-----
RIB Out Entries
-----
Routes : 1
=====
A:csc-ce-2#
```

It is also possible to check that CSC-CE-2 has properly installed the above VPN-IPv4 route into the routing table of the importing VPRN service, as shown below.

```
A:csc-ce-2# show router 1 route-table
=====
Route Table (Service: 1)
=====
Dest Prefix[Flags]                                Type      Proto      Age      Pref
```

```

      Next Hop[Interface Name]
-----
10.14.30.0/24                               Remote  BGP VPN  00h03m33s  170
      192.0.2.1 (tunneled)                               0
-----
No. of Routes: 1
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
=====
A:csc-ce-2#
```

Conclusion

Carrier Supporting Carrier is a scalable and secure solution for using an infrastructure IP VPN to transport traffic between dispersed CSC-CE devices belonging to an ISP or other service provider. Many different topology models are supported by the 7x50. This guide has explored one simplified configuration that can serve as the basis for more complicated setups.

Layer 3 VPN: VPRN Type Spoke

In This Chapter

This section provides information about Layer 3 VPRN CE hub and spoke architecture.

Topics in this section include:

- [Applicability on page 1374](#)
- [Summary on page 1375](#)
- [Overview on page 1376](#)
- [Configuration on page 1379](#)
- [Conclusion on page 1387](#)

Applicability

This example is applicable to the 7950 XRS, 7750 SR and 7450 ESS in mixed mode with a CPM3 or later, and is limited to IOM3-XP line cards or later. It is also supported on 7750 SR c4/12 systems. The configuration was tested on release 12.0.R3.

Summary

This example provides a basic technology overview and configuration examples of a network topology used for a Layer 3 VPRN CE hub and spoke architecture.

Knowledge of the Alcatel-Lucent's Layer 3 VPN concepts is assumed throughout this document.

Overview

Prior to SR OS release 12.0 a **CE hub and spoke** architecture was partially supported. Internal optimization was available for the hub sites connected to the same PE router only. This feature is known as VPRN **type hub**. If, on the other hand, multiple spoke sites were connected to the same PE router, separate VPRN instances had to be created to maintain the split horizon forwarding behavior. This approach was complex, hard to maintain and consumed extra VPRN instances.

Release 12.0.R1 adds new functionality to overcome these limitations. Introducing the VPRN **type spoke** feature allows multiple spoke sites to be kept within the same VPRN instance while at the same time maintaining the split horizon approach such that spoke sites cannot send traffic directly to each other.

The primary goal of the feature is to allow multiple spoke sites to be part of a single VPRN instance without allowing direct communication between the spoke CE sites which are part of that VPRN (of type spoke). The packet flow is demonstrated in [Figure 199](#).

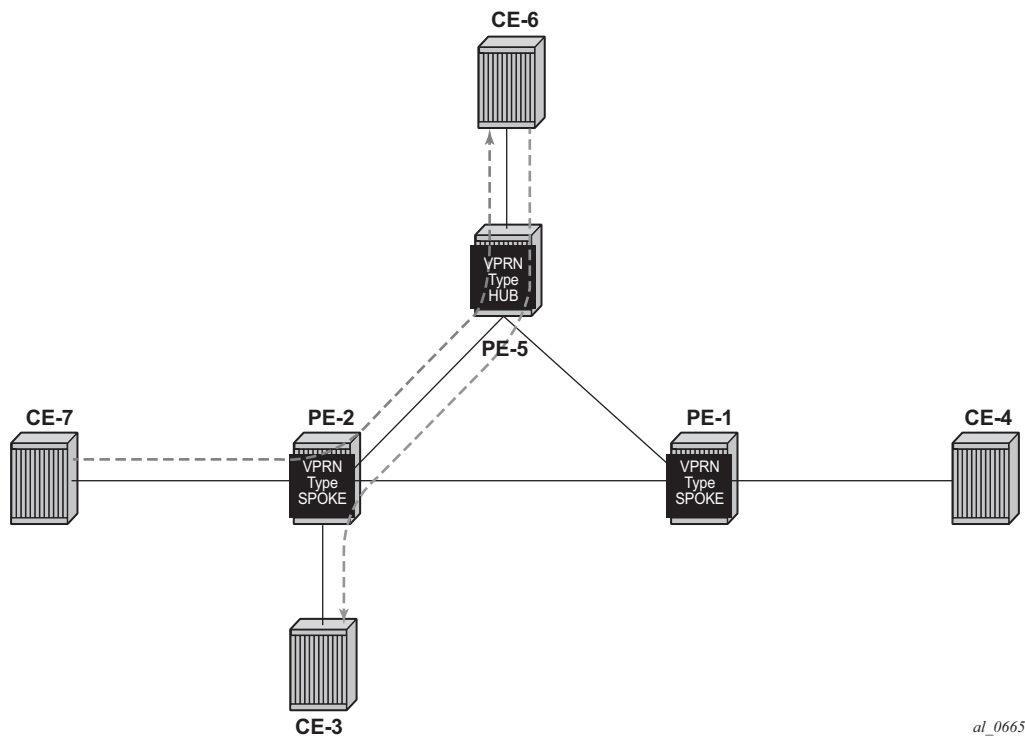


Figure 199: CE Hub and Spoke Data Path

The only way for CE-7 to communicate with CE-3 is via hub site CE-6. The same applies to CE-7 and CE-4 communication. The VPRN on PE-2 is configured as **type spoke** and has IP interfaces using SAPs or spoke SDPs that are considered spoke sites only. No direct communication between any of the spoke CE sites in the network is allowed.

This is achieved using two techniques (Figure 200).

- Use the **type spoke** command under the VPRN context as explained later.
- The extended community configuration using route-target policies (this is not covered in detail in this example).

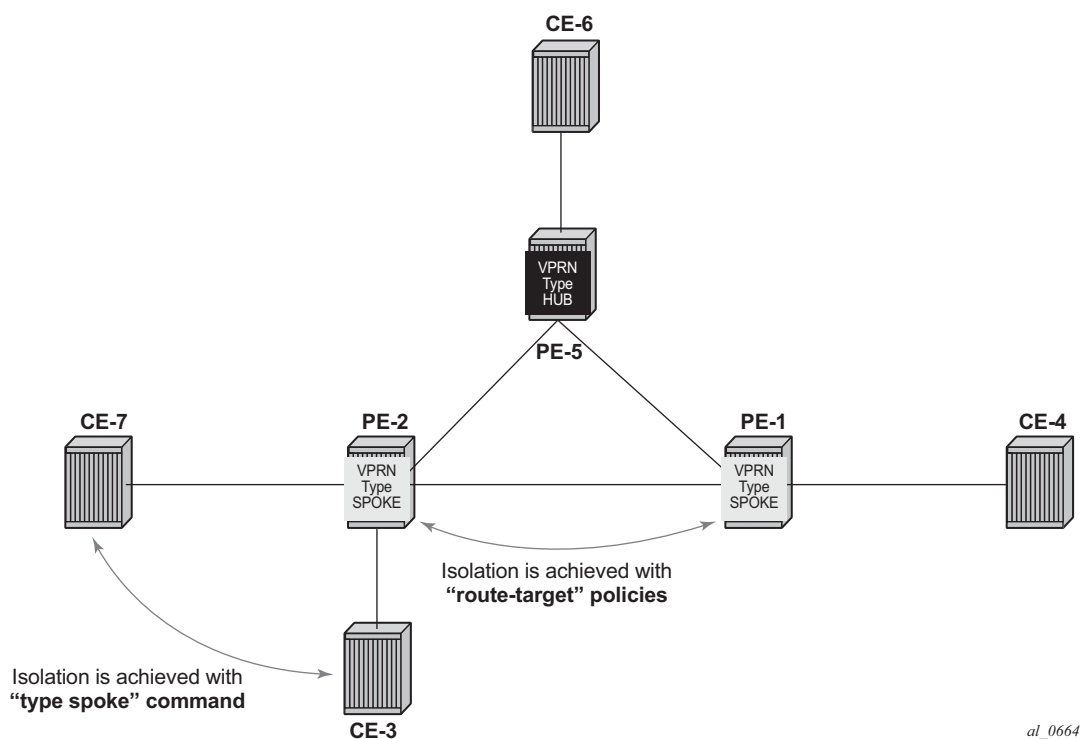


Figure 200: CE Hub and Spoke Control Plane Isolation

When a VPRN on a PE router is configured as **type spoke** then the internal forwarding logic changes as demonstrated in Figure 201.

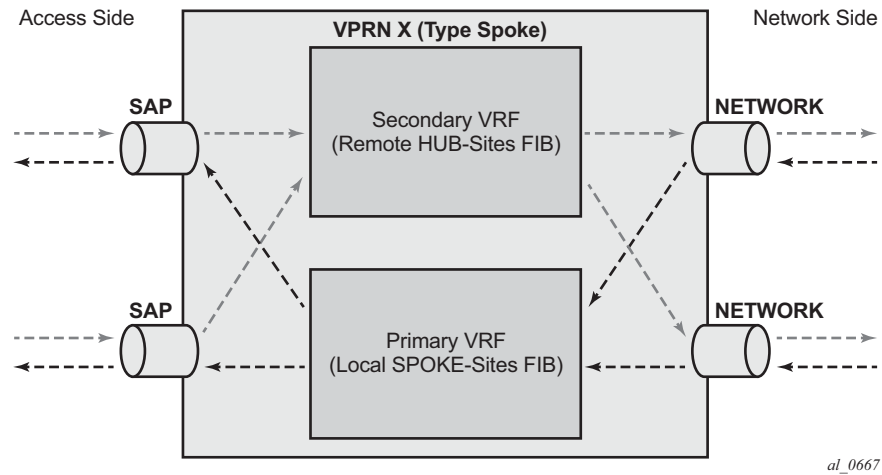


Figure 201: Internal VPRN Logic on a PE Router

- VPRNs of type spoke create a primary and a secondary VRF internally to the VPRN:
 - The primary VRF is used for forwarding traffic from the network interfaces towards the IP interfaces using SAPs or spoke SDPs. This VRF is populated with routes learned from the spoke CE sites connected to the local PE through IP interfaces using SAPs or spoke SDPs.
 - The secondary VRF is used for forwarding traffic from the IP interfaces using SAPs or spoke SDPs towards the network interfaces or other VPRN instances. This VRF is populated with routes learned via MP-BGP from Hub sites.
- VPRNs of type spoke export routes using a specific extended community (for instance spoke-ext-comm) via an export policy to identify them as spoke site originated routes.
 - This community is not hard-coded and has to be configured manually (see configuration example later).
- VPRNs of type spoke import routes (using an import policy) received from other PEs or VPRN instances with a hub specific community only (for example hub-ext-comm). Routes with spoke-ext-comm community are ignored.
 - This community is not hard-coded and has to be configured manually (see configuration example later).
- Multiple VPRNs of type spoke and hub can coexist on the same PE if they use different VPRN instances.
- The configuration of type hub and type spoke is mutually exclusive within one VPRN instance.

Configuration

The physical topology and addressing scheme are presented in [Figure 202](#).

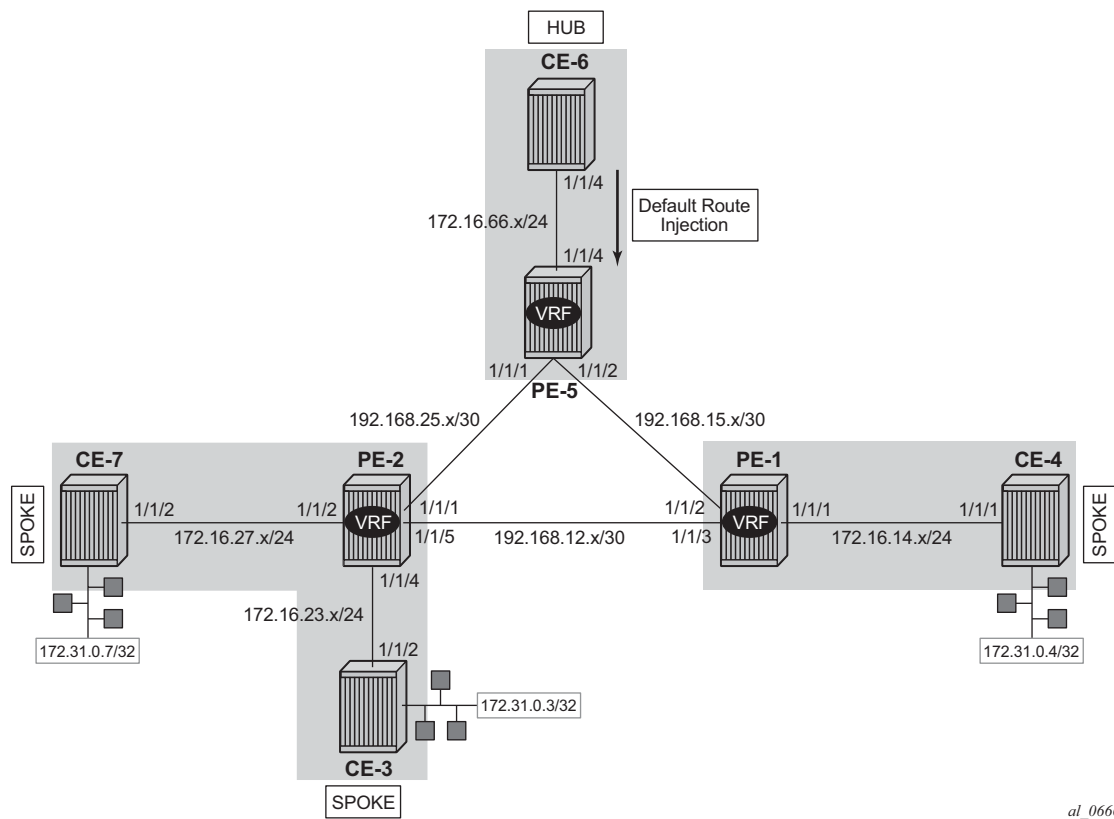


Figure 202: CE Hub and Spoke Topology and Addressing Scheme

The configuration of PE-2 and PE-5 are the main focus of this example. The configuration of PE-1 is similar to that of PE-2.

Hub Site Configuration

Only the essential part of the configuration is provided for the hub site.
PE-5 is configured with VPRN 1 providing OSPF connectivity to customer CE-6.

```
A:PE-5# configure service vprn 1
*A:PE-5>config>service>vprn# info
-----
vrf-import "vrf-import"
vrf-export "vrf-export"
route-distinguisher 1:5
type hub
auto-bind mpls-gre
interface "int-PE-5-CE-6" create
    address 172.16.56.1/24
    sap 1/1/3:100 create
    exit
exit
ospf
    export "export-ospf"
    area 0.0.0.0
        interface "int-PE-5-CE-6"
            interface-type point-to-point
            mtu 1500
            no shutdown
        exit
    exit
exit
no shutdown
```

Vrf-import and export policies are used to manipulate the vrf-target in order to achieve logical isolation between the spoke sites in the network.

```
*A:PE-5>config>router>policy-options# info
-----
community "hub-ext-comm" members "target:64500:11"
community "spoke-ext-comm" members "target:64500:12"
policy-statement "vrf-export"
    default-action accept
    community add "hub-ext-comm"
    exit
exit
policy-statement "vrf-import"
    entry 10
        from
            community "spoke-ext-comm"
        exit
        action accept
        exit
    exit
    default-action reject
exit
```


At the same time CE-6 is configured (not shown) to advertise a default route which is used by all remote spoke CE sites to forward traffic via CE-6.

Hub Site Verification

The routing table (RIB) for VPRN 1 on PE-5 (hub site) lists all reachable networks.

```
*A:PE-5# show router 1 route-table
=====
Route Table (Service: 1)
=====
Dest Prefix[Flags]                                Type      Proto      Age          Pref
  Next Hop[Interface Name]                        Metric
-----
0.0.0.0/0                                           Remote    OSPF        21h28m50s    150
      172.16.56.2                                   1
172.16.14.0/24                                       Remote    BGP VPN     00h15m59s    170
      192.0.2.1 (tunneled)                           0
172.16.23.0/24                                       Remote    BGP VPN     00h15m59s    170
      192.0.2.2 (tunneled)                           0
172.16.27.0/24                                       Remote    BGP VPN     00h15m59s    170
      192.0.2.2 (tunneled)                           0
172.16.56.0/24                                       Local     Local       21h29m07s    0
      int-PE-5-CE-6                                   0
172.31.0.3/32                                       Remote    BGP VPN     00h15m59s    170
      192.0.2.2 (tunneled)                           0
172.31.0.4/32                                       Remote    BGP VPN     00h15m59s    170
      192.0.2.1 (tunneled)                           0
172.31.0.7/32                                       Remote    BGP VPN     00h15m59s    170
      192.0.2.2 (tunneled)                           0
-----
No. of Routes: 8
```

The forwarding table (FIB) for the primary VRF of VPRN 1 is displayed using following command. All remote spoke and hub sites are reachable via this VRF.

```
*A:PE-5# show router 1 fib 1
=====
FIB Display
=====
Prefix                                           Protocol
  NextHop
-----
0.0.0.0/0                                           OSPF
      172.16.56.2 (int-PE-5-CE-6)
172.16.14.0/24                                       BGP_VPN
      192.0.2.1 (VPRN Label:262143 Transport:LDP)
172.16.23.0/24                                       BGP_VPN
      192.0.2.2 (VPRN Label:262142 Transport:LDP)
172.16.27.0/24                                       BGP_VPN
      192.0.2.2 (VPRN Label:262142 Transport:LDP)
172.16.56.0/24                                       LOCAL
      172.16.56.0 (int-PE-5-CE-6)
```

Hub Site Verification

```
172.31.0.3/32                                     BGP_VPN
  192.0.2.2 (VPRN Label:262142 Transport:LDP)
172.31.0.4/32                                     BGP_VPN
  192.0.2.1 (VPRN Label:262143 Transport:LDP)
172.31.0.7/32                                     BGP_VPN
  192.0.2.2 (VPRN Label:262142 Transport:LDP)
-----
Total Entries : 8
-----
=====
```

The forwarding table for the secondary VRF of VPRN 1 is displayed using following command, including the **secondary** keyword. All local hub CE sites are reachable via this VRF.

```
*A:PE-5# show router 1 fib 1 secondary
=====
FIB Display
=====
Prefix                                     Protocol
  NextHop
-----
0.0.0.0/0                                     OSPF
  172.16.56.2 (int-PE-5-CE-6)
172.16.56.0/24                               LOCAL
  172.16.56.0 (int-PE-5-CE-6)
-----
Total Entries : 2
-----
=====
```

Spoke Site Configuration

According to the network topology two spoke VPRNs are present. One VPRN with two CE spoke sites connected is located on PE-2 and another VPRN with one spoke CE site is located on PE-1. The service configuration for PE-2 is shown below with the one for PE-1 being similar.

PE-2 is configured with VPRN 1, which has OSPF connectivity to the customer CE-3 and CE-7. Note the new command **type spoke** which is used to prevent direct CE spoke to CE spoke communications for this VPRN.

```
A:PE-2# configure service vprn 1
A:PE-2>config>service>vprn# info
-----
vrf-import "vrf-import"
vrf-export "vrf-export"
route-distinguisher 1:2
type spoke
auto-bind mpls-gre
interface "int-PE-2-CE-7" create
  address 172.16.27.1/24
  sap 1/1/2:100 create
  exit
exit
interface "int-PE-2-CE-3" create
  address 172.16.23.1/24
  sap 1/1/4:100 create
  exit
exit
ospf
  export "export-ospf"
  area 0.0.0.0
    interface "int-PE-2-CE-7"
      interface-type point-to-point
      mtu 1500
      no shutdown
    exit
    interface "int-PE-2-CE-3"
      interface-type point-to-point
      mtu 1500
      no shutdown
    exit
  exit
exit
no shutdown
```

Vrf-import and export policies are used to build a hub-and-spoke topology in order to achieve a logical isolation between spoke sites connected to different PE routers.

```
*A:PE-2# configure router policy-options
*A:PE-2>config>router>policy-options# info
-----
community "hub-ext-comm" members "target:64500:11"
community "spoke-ext-comm" members "target:64500:12"
policy-statement "vrf-export"
```

Hub Site Verification

```
        default-action accept
        community add "spoke-ext-comm"
    exit
exit
policy-statement "vrf-import"
    entry 10
        from
            community "hub-ext-comm"
        exit
        action accept
        exit
    exit
    default-action reject
exit
```

For connectivity verification purposes CE-3, CE-4 and CE-7 are configured to advertise their internal loopback interfaces via OSPF:

- CE-3 advertises 172.31.0.3/32
- CE-4 advertises 172.31.0.4/32
- CE-7 advertises 172.31.0.7/32

Spoke Site Verification

The RIB for VPRN 1 on PE-2 (spoke VPRN) lists all reachable networks.

The other spoke sites connected to the remote PEs (only CE-4 here, for example: 172.31.0.4/32)) are not present in the routing table.

PE-2's local interface addresses (172.16.23.1/32 and 172.16.27.1/32) are present in the routing table of VPRN 1. From a FIB point of view these are reachable from any spoke VPRN but the spoke CE's router host addresses are not. This fact does not influence the data plane isolation for the customer networks.

```
*A:PE-2# show router 1 route-table
```

```
=====
Route Table (Service: 1)
```

```
=====
Dest Prefix[Flags]                                Type  Proto  Age      Pref
Next Hop[Interface Name]                        Metric
-----
0.0.0.0/0                                           Remote BGP VPN 00h19m04s 170
192.0.2.5 (tunneled)                               0
172.16.23.0/24                                       Local  Local  21h55m36s 0
int-PE-2-CE-3                                       0
172.16.23.1/32                                       Remote Host 21h55m36s 0
int-PE-2-CE-3                                       0
172.16.27.0/24                                       Local  Local  21h55m36s 0
int-PE-2-CE-7                                       0
172.16.27.1/32                                       Remote Host 21h55m36s 0
```

Layer 3 VPN: VPRN Type Spoke

```

int-PE-2-CE-7
172.16.56.0/24 Remote BGP VPN 00h19m04s 170
192.0.2.5 (tunneled) 0
172.31.0.3/32 Remote OSPF 04h46m36s 10
172.16.23.2 100
172.31.0.7/32 Remote OSPF 21h55m21s 10
172.16.27.2 100
-----
No. of Routes: 8

```

The FIB for the primary VRF of VPRN 1 shows all local spoke sites are reachable via this VRF.

```

*A:PE-2# show router 1 fib 1
=====
FIB Display
=====
Prefix NextHop Protocol
-----
172.16.23.0/24 LOCAL
172.16.23.0 (int-PE-2-CE-3)
172.16.23.1/32 HOST
Blackhole
172.16.27.0/24 LOCAL
172.16.27.0 (int-PE-2-CE-7)
172.16.27.1/32 HOST
Blackhole
172.31.0.3/32 OSPF
172.16.23.2 (int-PE-2-CE-3)
172.31.0.7/32 OSPF
172.16.27.2 (int-PE-2-CE-7)
-----
Total Entries : 6
=====

```

The FIB for the secondary VRF of VPRN 1 shows the remote hub site (address 172.16.56.0/24) is reachable via this VRF.

```

*A:PE-2# show router 1 fib 1 secondary
=====
FIB Display
=====
Prefix NextHop Protocol
-----
0.0.0.0/0 BGP_VPN
192.0.2.5 (VPRN Label:262141 Transport:LDP)
172.16.23.1/32 HOST
Blackhole
172.16.27.1/32 HOST
Blackhole
172.16.56.0/24 BGP_VPN
192.0.2.5 (VPRN Label:262141 Transport:LDP)
-----
Total Entries : 4

```

=====

Spoke Sites Connectivity Verification

Without the VPRN spoke type configuration in VPRN 1 on PE-2 CE-3 takes the shortest path to CE-7, which violates the “hub and spoke” design approach explained above.

```
A:CE-3# traceroute 172.31.0.7 no-dns
traceroute to 172.31.0.7, 30 hops max, 40 byte packets
 1 172.16.23.1      3.22 ms  14.3 ms  2.75 ms
 2 172.31.0.7       3.47 ms  3.34 ms  3.42 ms
A:CE-3# traceroute router 100 172.31.0.7 no-dns
```

After enabling the **type spoke** feature, CE-3 takes the longest path via hub CE-6 to reach CE-7, as it should.

```
*A:CE-3# traceroute 172.31.0.7 no-dns
traceroute to 172.31.0.7, 30 hops max, 40 byte packets
 1 172.16.23.1      3.16 ms  2.79 ms  2.79 ms
 2 0.0.0.0          * * *
 3 172.16.56.2      69.7 ms  7.31 ms  10.5 ms
 4 172.16.56.1      32.0 ms  67.5 ms  80.6 ms
 5 172.16.27.1      7.54 ms  7.52 ms  33.6 ms
 6 172.31.0.7       77.4 ms  90.7 ms  12.8 ms
```

Similarly, the long path is taken by CE-3 to reach CE-4.

```
*A:CE-3# traceroute 172.31.0.4 no-dns
traceroute to 172.31.0.4, 30 hops max, 40 byte packets
 1 172.16.23.1      72.6 ms  2.82 ms  2.72 ms
 2 0.0.0.0          * * *
 3 172.16.56.2      10.8 ms  41.7 ms  9.51 ms
 4 172.16.56.1      10.6 ms  20.7 ms  10.8 ms
 5 172.16.14.1      11.8 ms  11.9 ms  11.6 ms
 6 172.31.0.4       20.9 ms  15.6 ms  15.4 ms
```

Conclusion

The VPRN type spoke feature completes the **CE hub and spoke** solution. It brings a new level of simplicity, scalability and flexibility to operators using this VPRN architecture for their customers.

Multicast in a VPN I

In This Chapter

This section provides information about multicast in a VPRN service.

Topics in this section include:

- [Applicability on page 1390](#)
- [Summary on page 1391](#)
- [Overview on page 1394](#)
- [Configuration on page 1397](#)
- [Conclusion on page 1445](#)

Applicability

This section is applicable to all of the 7750 and 7710 SR series and was tested on release 7.0R5. There are no pre-requisites for this configuration. This is supported on 7450 ESS-7 or ESS-12 in mixed-mode since 8.0R1. The 7750 SR-c4 is supported from 8.0R4 and higher.

Summary

Multicast VPN (MVPN) architectures describe a set of VRFs that support the transport of multicast traffic across a provider network.

Draft-rosen-vpn-mcast-08.txt (herein referred to as Draft-Rosen) describes the use of multicast distribution trees (MDT) established between PEs within a given VRF. Each VRF required its own tree. Customer edge routers form Protocol Independent Multicast (PIM) adjacencies with the PE, and PE-PE PIM adjacencies are formed across the multicast tree. PIM signaling and data streams are transported across the MDT. There are a number of limitations with the Draft-Rosen implementation including, but not limited to:

- Draft Rosen requires a set of MDTs per VPN, which requires a PIM state per MDT. There is no option to aggregate MDT across multiple VPNs
- Customer signaling, PE discovery and Data MDT signaling are all PIM-based. There is no mechanism available to decouple these. Thus there is an incongruity between Unicast and multicast VPNs using Draft-Rosen
 - There is no mechanism for using MPLS to encapsulate multicast traffic in the VPN. GRE is the only encapsulation method available in Draft-Rosen.
 - Draft-Rosen multicast trees are signalled using PIM only. MVPN allows the use of mLDP, RSVP P2MP LSPs.
 - PE to PE protocol exchanges for Draft-Rosen is achieved using PIM only. MVPN allows for the use of BGP signaling as per unicast Layer 3 VPNs.

Next Generation MVPN addresses these limitations by extending the idea of the per-VRF tree, by introducing the idea of Provider Multicast Service Interfaces (PMSI). These are equivalent to the default MDTs of Draft-Rosen in that they support control plane traffic (customer multicast signaling), and the data MDTs which carry multicast data traffic streams between PEs within a multicast VRF.

Next Generation MVPN allows the decoupling of the mechanism required to create a multicast VPN, such as PE auto-discovery (which PEs are members of which VPN), PMSI signaling (creation of tunnels between PEs) and customer multicast signaling (multicast signaling —IGMP/PIM — received from customer edge routers). Two types of PMSI exist:

- Inclusive (I-PMSI): contains all the PEs for a given MVPN.
- Selective (S-PMSI): contains only a subset of PEs of a given MVPN.

Knowledge of MPLS-VPN RFC 4364, *BGP/MPLS IP Virtual Private Networks (VPNs)*, architecture and functionality, as well as an understanding of multicast protocols, is assumed throughout.

This chapter provides configuration details required to implement the parts of Next Generation MVPN shown in [Table 10](#).

Table 10: Next Generation MVPN Components

Provider Multicast Domain				Customer Multicast Domain		
I-PMSI	Auto-discovery	C-MCAST	S-PMSI Creation	PE-based RP	Anycast RP on PE	PIM SSM
PIM ASM	PIM	PIM join/leave	PIM SSM with S-PMSI join TLV	X	X	X
PIM ASM	BGP A/D	PIM join/leave	PIM SSM with S-PMSI join TLV			X

The first section of this chapter describes the common configuration required for each PE within the provider multicast domain regardless of the MVPN PE auto-discovery or customer signaling methods. This includes IGP and VPRN service configuration.

Following the common configuration, specific MVPN configuration required for the configuration for the provider multicast domain using PIM Any Source Multicast (ASM) with auto-discovery based on PIM or BGP auto-discovery (A/D), PIM used for the customer multicast signaling and PIM Source Specific Multicast (SSM) used for the S-PMSI creation are described. The customer domain configuration covers the following three cases:

1. PIM ASM with the Rendezvous Point (RP) in the provider PE
2. PIM ASM using anycast RP on the provider RPs
3. PIM SSM

Other possible options, not covered in this section but are discussed in the 7750 SR/7710 SR/7450 ESS OS Routing Protocols Guide:

- The use of PIM SSM for the provider multicast I-PMSI.
- The use of BGP for the customer multicast signaling in the provider multicast domain.
- The provider S-PMSI creation through BGP S-PMSI A/D.
- The use of the customer RP based in the customer CE.

The use of mLDP and RSVP p2mp LSPs for the I/S-PMSI was not available in release 7.0.

The Multicast in a VPRN II example in [Multicast in a VPN II on page 1447](#) introduces features that were not supported in Release 7.0R5. It provides configuration details to implement:

- Multicast LDP (mLDP) and RSVP-TE Point to Multi-point (P2MP) for building customer trees (C-trees) which are using MPLS instead of PIM techniques.
 - MVPN source redundancy
 - MDT AFI/SAFI (to fully interoperate with Cisco networks).
-

References

- IETF
 - BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPNs, draft-ietf-l3vpn-2547bis-mcast-bgp-05.txt
 - Multicast in MPLS/BGP IP VPNs, draft-ietf-l3vpn-2547bis-mcast-07.txt
- 7750 SR/7710 SR/7450 ESS OS Services Guide

Overview

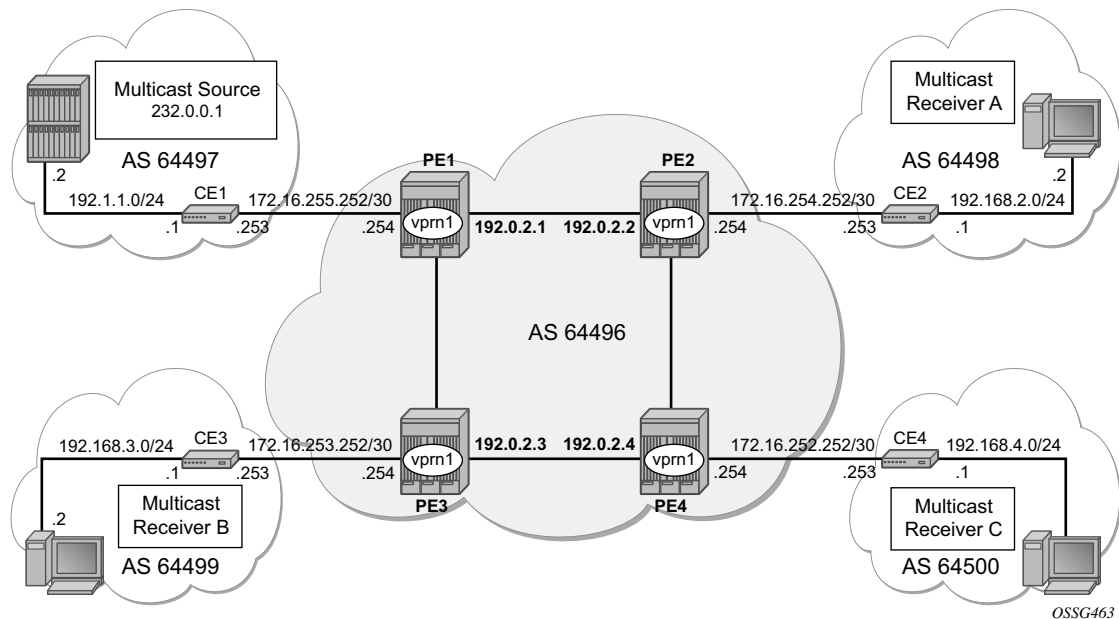


Figure 203: Network Topology

The network topology is displayed in [Figure 203](#). The setup consists of four SR 7750s acting as Provider Edge (PE) routers within a single Autonomous System (AS).

- Full mesh ISIS or OSPF in each AS
- LDP on all interfaces in each AS (RSVP could also be used)
- MP-iBGP sessions between the PE routers in each AS (Route Reflectors (RRs) could also be used).
- Layer 3-VPN on all PEs with identical route targets, in the form AS-no: *vprn-service-id*

Connected to each PE is a single 7750 acting as a Customer Edge (CE) router. CE-1 has a multicast sources connected, and PEs 2 to 4 each have a single receiver connected which will receive the multicast streams from the source. In this document, each receiver is both IGMPv2 and IGMPv3 capable. If the customer domain multicast signaling plane uses Source Specific

Multicasting (SSM), then an IGMPv3 receiver is configured; if Any Source Multicasting (ASM) is used, the receiver is IGMPv2 capable.

If the receiver is IGMPv3 capable, it will issue IGMPv3 reports that will include a list of required source addresses. The receiver will join the 232.0.0.1 multicast group.

If the receiver is only IGMPv2 capable, then it will issue IGMPv2 reports which do not specify a source of the group. In this case a Rendezvous Point is required within the PIM control plane of the multicast VRF which is source-aware. In this case, the receiver will join the 225.0.0.1 multicast group.

When the receiver wishes to become a member of any group, the source address of the group must be known to the CE. As a result the source address must be IP reachable by each CE, so it is advertised by CE-1 to the PEs with attachment circuits in VPRN1 using BGP.

Static routes are then configured on the receiver CEs to achieve IP reachability to the source address of multicast groups. In the case of PIM ASM, any RP that is configured must also be reachable from the CE.

Multicast VPN Overview

Multicast traffic from the source is streamed towards router CE-1. Receivers connected to PE-2, PE-3 and PE-4 are interested in joining this multicast group.

CEs 1 to 4 are PIM enabled routers, which form a PIM adjacency with nearest PE. The PIM adjacencies between PEs across the Provider network are achieved using I-PMSIs. I-PMSIs carry PIM control messages between PEs. Data plane traffic is transported across the I-PMSI until a configured bandwidth threshold is reached. A Selective PMSI is then signaled that carries data plane traffic. This threshold can be as low as 1kb/second and must be explicitly configured along with the S-PMSI multicast group. An S-PMSI per customer group per VPRN is configured. If no S-PMSI and threshold is configured, data traffic will continue to be forwarded across the provider network within the I-PMSI.

Configuration

The configuration is divided into the following sections:

- Provider Common Configuration
 - PE Global Configuration
 - PE VPRN Configuration
- PE VPRN Multicast Configuration
 - Auto-Discovery within Provider Domain using PIM
 - PIM Autodiscovery: Customer Signaling using PIM
- PIM Any Source Multicasting with RP at the provider PE
- PIM Any Source Multicasting with Anycast RP at the provider PE
- PIM Source Specific Multicasting
 - BGP Autodiscovery: PE VPRN Multicast Configuration
 - Data Path Using Selective-PMSIs

Provider Common Configuration

This section describes the common configuration required for each PE within the Provider multicast domain, regardless of the MVPN PE auto-discovery or customer signaling methods. This includes IGP and VPRN service configuration.

The configuration tasks can be summarized as follows:

- PE global configuration. This includes configuration of the Interior Gateway Protocol (IGP) (ISIS or OSPF); configuration of link layer LDP between PEs; configuration of iBGP between PEs, to facilitate VPRN route learning; configuration of PIM.
- VPRN configuration on PEs. This includes configuration of basic VPRN parameters (route-distinguisher, route target communities); configuration of attachment circuits towards CEs; configuration of VRF routing protocol and any policies towards CE.
- VRF PIM and MVPN parameters — I-PMSI
- CE configuration.

PE Global Configuration

Step 1. On each of the PE routers, configure the appropriate router interfaces, OSPF (or ISIS) and link layer LDP. For clarity in the following configuration steps, only the configuration for PE-1 is shown. PE-2, PE-3 and PE-4 are similar.

```
A:PE-1>config router
    interface "int-pe1-pe2"
        address 192.168.1.1/30
        port 1/1/1
    exit
    interface "int-pe1-pe3"
        address 192.168.2.1/30
        port 1/1/2
    exit
    interface "system"
        address 192.0.2.1/32
    exit
    autonomous-system 64496
    ospf
        area 0.0.0.0
            interface "system"
            exit
            interface "int-pe1-pe2"
                interface-type point-to-point
            exit
            interface "int-pe1-pe3"
                interface-type point-to-point
            exit
        exit
    exit
    ldp
        interface-parameters
            interface "int-pe1-pe2"
            exit
            interface "int-pe1-pe3"
            exit
        exit
        targeted-session
        exit
    exit
```

Step 2. Verify that OSPF adjacencies are formed and that LDP peer sessions are formed.

```
A:PE-1# show router ospf neighbor
=====
OSPF Neighbors
=====
Interface-Name          Rtr Id          State    Pri  RetxQ  TTL
-----
int-pe1-pe2             192.0.2.2       Full     1    0      31
int-pe1-pe3             192.0.2.3       Full     1    0      37
-----
No. of Neighbors: 2
=====

A:PE-1 # show router ldp session
=====
LDP Sessions
=====
Peer LDP Id          Adj Type  State      Msg Sent  Msg Recv  Up Time
-----
192.0.2.2:0          Link      Established 8651      8651      0d 06:38:44
192.0.2.3:0          Link      Established 8697      8694      0d 06:40:20
-----
No. of Sessions: 2
=====
A:PE-1 #
```

Step 3. Configure BGP between the PEs for VPRN routing.

```
A:PE-1> configure router
      bgp
        group "internal"
          family vpn-ipv4
          peer-as 64496
          neighbor 192.0.2.2
          exit
          neighbor 192.0.2.3
          exit
          neighbor 192.0.2.4
          exit
        exit
      exit
```

Step 4. Verify that BGP peer relationship is established.

```
A:PE-1# show router bgp summary
=====
BGP Router ID:192.0.2.1          AS:64496          Local AS:64496
=====
BGP Admin State      : Up          BGP Oper State      : Up
Total Peer Groups    : 1           Total Peers          : 3
Total BGP Paths       : 14         Total Path Memory    : 1800
Total IPv4 Remote Rts : 0           Total IPv4 Rem. Active Rts : 0
Total IPv6 Remote Rts : 0           Total IPv6 Rem. Active Rts : 0
Total Suppressed Rts  : 0           Total Hist. Rts      : 0
Total Decay Rts       : 0

Total VPN Peer Groups : 2           Total VPN Peers      : 2
Total VPN Local Rts   : 4
Total VPN-IPv4 Rem. Rts : 3         Total VPN-IPv4 Rem. Act. Rts: 3
Total VPN-IPv6 Rem. Rts : 0         Total VPN-IPv6 Rem. Act. Rts: 0
Total L2-VPN Rem. Rts : 0           Total L2VPN Rem. Act. Rts : 0
Total VPN Supp. Rts   : 0           Total VPN Hist. Rts   : 0
Total VPN Decay Rts   : 0
Total MVPN-IPv4 Rem Rts : 0         Total MVPN-IPv4 Rem Act Rts : 0
=====
BGP Summary
=====
Neighbor
      AS PktRcvd InQ Up/Down  State|Rcv/Act/Sent (Addr Family)
      PktSent OutQ
-----
192.0.2.2
      64496    9429    0 01d02h39m 1/1/4 (VpnIPv4)
      9477    0
192.0.2.3
      64496    9435    0 01d02h39m 1/1/4 (VpnIPv4)
      9494    0
192.0.2.4
      64496    9431    0 01d02h39m 1/1/4 (VpnIPv4)
      9457    0
=====
A:PE-1#
```

Step 5. Enable PIM on all network interfaces, including the system interface. This allows the signaling of PMSIs that transport PIM signaling within each VRF.

Step 6. As each I-PMSI will be signalled using PIM ASM, a rendezvous point (RP) is required within the global PIM configuration. A static RP is used and PE-1 is selected. All PEs must be configured with this RP address.

```
A:PE-1> Configure router
      pim
        interface "system"
        exit
        interface "int-pe1-pe2"
        exit
        interface "int-pe1-pe3"
        exit
      rp
        static
          address 192.0.2.1
          group-prefix 239.255.0.0/16
        exit
        bsr-candidate
        shutdown
      exit
      rp-candidate
      shutdown
      exit
    exit
  exit
```

Step 7. Verify PIM neighbor relationship

```
A:PE-1# show router pim neighbor
=====
PIM Neighbor ipv4
=====
Interface          Nbr DR Prty    Up Time      Expiry Time   Hold Time
  Nbr Address
-----
int-pe1-pe2         1              0d 23:21:04   0d 00:01:15   105
  192.168.1.2
int-pe1-pe3         1              0d 23:22:40   0d 00:01:34   105
  192.168.2.2
-----
Neighbors : 2
=====
A:PE-1#
```

PE VPRN Configuration

A VPRN (VPRN 1) is created on each PE. This will be the multicast VPRN. PE-1 will be the PE containing the attachment circuit towards CE-1. CE-1 will be the CE nearest the source. PE-2, PE-3 and PE-4 will contain attachment circuits towards CE-2, CE-3 and CE-4 respectively. Each CE will have a receiving host attached.

Step 1. Create VPRN 1 on each PE, containing a route-distinguisher and vrf-target of 64496:1. The autonomous system number is 64496. Use auto-bind LDP for next hop tunnel route resolution.

```
A:PE-1>config>service>vprn# info
-----
autonomous-system 64496
route-distinguisher 64496:1
auto-bind ldp
vrf-target target:64496:1
...
-----
A:PE-1>config>service>vprn#
```

Step 2. Create an attachment circuit interface towards the CE.

```
configure service vprn 1
  interface "int-pe1-ce1" create
    address 172.16.255.254/30
    sap 1/1/4:1 create
  exit
exit
```

Step 3. The source address of the multicast stream will need to be reachable by all routers (PEs and CEs) within the VPN. This will be advertised within BGP from the CE to the PE. Create a BGP peering relationship with the CE.

```
configure service vprn 1
  bgp
    group "external"
      type external
      peer-as 64497
      neighbor 172.16.255.253
    exit
  exit
exit
```

Step 4. On CE-1, create a VPRN to support the connection of the source to the CE and to connect the CE to the PE. Two attachment circuits are required, as well as a BGP peering relationship with the PE. This uses a default address family of **ipv4**.
(NB — a pair of IES services could also be used to provide the attachment circuits)

```
*A:CE-1>config service vprn 1
    autonomous-system 64497
    route-distinguisher 64497:1
    interface "int-cel-pe1" create
        address 172.16.255.253/30
        sap 1/1/1:1 create
        exit
    exit
    interface "to-source" create
        address 192.168.1.1/24
        sap 1/1/3:1 create
        exit
    exit
    bgp
        group "external"
            type external
            peer-as 64496
            neighbor 172.16.255.254
            exit
        exit
    exit
    no shutdown
```

```
-----
*A:CE-1>config>service>vprn#
```

Step 5. Verify PE-CE BGP peer relationship on CE-1 and PE-1:

```
*A:CE-1# show router 1 bgp summary
=====
BGP Router ID:192.0.2.5          AS:64497          Local AS:64497
=====
BGP Admin State      : Up          BGP Oper State      : Up
Total Peer Groups    : 1          Total Peers          : 1
Total BGP Paths       : 3          Total Path Memory    : 412
Total IPv4 Remote Rts : 2          Total IPv4 Rem. Active Rts : 2
Total IPv6 Remote Rts : 0          Total IPv6 Rem. Active Rts : 0
Total Supressed Rts   : 0          Total Hist. Rts      : 0
Total Decay Rts       : 0

=====
BGP Summary
=====
Neighbor
      AS PktRcvd InQ Up/Down  State|Rcv/Act/Sent (Addr Family)
      PktSent OutQ
-----
172.16.255.254
          64496  10709    0 22h18m21s 2/2/1 (IPv4)
          10696    0

=====
*A:CE-1#

A:PE-1# show router 1 bgp summary
=====
BGP Router ID:192.0.2.1          AS:64496          Local AS:64496
=====
BGP Admin State      : Up          BGP Oper State      : Up
Total Peer Groups    : 1          Total Peers          : 1
Total BGP Paths       : 4          Total Path Memory    : 520
Total IPv4 Remote Rts : 1          Total IPv4 Rem. Active Rts : 0
Total IPv6 Remote Rts : 0          Total IPv6 Rem. Active Rts : 0
Total Supressed Rts   : 0          Total Hist. Rts      : 0
Total Decay Rts       : 0

=====
BGP Summary
=====
Neighbor
      AS PktRcvd InQ Up/Down  State|Rcv/Act/Sent (Addr Family)
      PktSent OutQ
-----
172.16.255.253
          64497  10699    0 22h22m38s 1/0/2 (IPv4)
          10721    0

=====
A:PE-1#
```


Step 6. In order for the CE connecting to the source to be advertised within BGP, a route policy is required. The subnet containing the multicast source is 192.168.1.0/24, so a prefix-list can be used to define a match, and then used within a route policy to inject into BGP.

```
A:CE-1>config>router>policy-options# info
-----
      prefix-list "source"
        prefix 192.168.1.0/24 exact
      exit
    policy-statement "policy-1"
      entry 10
        from
          prefix-list "source"
        exit
        to
          protocol bgp
        exit
        action accept
        exit
      exit
    exit
-----
A:CE-1>config>router>policy-options#
```

Step 7. Apply policy as an export policy within BGP context

```
A:CE-1# configure service vprn 1 bgp
      export "policy-1"
      group "external"
        type external
        peer-as 64496
        neighbor 172.16.255.254
      exit
    exit
```

This results in the 192.168.1.0/24 subnet being seen in the BGP RIB_OUT on CE-1

```
A:CE-1# show router 1 bgp routes 192.168.1.0/24 hunt
=====
BGP Router ID:192.0.2.5      AS:64497      Local AS:64497
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best
=====
BGP IPv4 Routes
=====
-----
RIB In Entries
-----
-----
RIB Out Entries
```

Provider Common Configuration

```
-----
Network      : 192.168.1.0/24
Nexthop      : 172.16.255.253
To           : 172.16.255.254
Res. Nexthop : n/a
Local Pref.  : n/a
Aggregator AS : None
Atomic Aggr. : Not Atomic
Community    : No Community Members
Cluster      : No Cluster Members
Originator Id : None
Origin       : Incomplete
AS-Path      : 64497
Interface Name : NotAvailable
Aggregator    : None
MED           : None
Peer Router Id : 192.0.2.1
-----
```

```
Routes : 1
```

```
=====
A:CE-1#
```

It is also seen in the PE-1 VRF 1 FIB:

```
A:PE-1# show router 1 route-table
=====
Route Table (Service: 1)
=====
Dest Prefix                                Type  Proto  Age           Pref
  Next Hop[Interface Name]                Metric
-----
172.16.253.252/30                         Remote BGP VPN 00h07m30s    170
      192.0.2.3                           0
172.16.254.252/30                         Remote BGP VPN 01d01h30m    170
      192.0.2.2                           0
172.16.255.252/30                         Local  Local  22h48m52s     0
      int-pe1-ce1                         0
192.168.1.0/24                             Remote BGP    00h09m44s    170
      172.16.255.253                       0
-----
No. of Routes: 4
=====
A:PE-1#
```

This prefix will also be automatically advertised within the BGP VPRN to all other PEs, and will be installed in VRF 1.

For example, on PE-2:

```
*A:PE-2# show router 1 route-table
=====
Route Table (Service: 1)
=====
Dest Prefix                                Type  Proto  Age          Pref
  Next Hop[Interface Name]                Metric
-----
172.16.253.252/30                          Remote BGP VPN 00h09m52s  170
      192.0.2.3                             0
172.16.254.252/30                          Local  Local  01d01h33m   0
      int-pe2-ce2                             0
172.16.255.252/30                          Remote BGP VPN 00h51m26s  170
      192.0.2.1                             0
192.168.1.0/24                             Remote BGP VPN 00h11m48s  170
      192.0.2.1                             0
-----
No. of Routes: 4
=====
*A:PE-2#
```

Each CE containing the Multicast Receivers must be able to reach the source. The following output shows the VPRN configuration of CE-2 containing an interface towards PE-2 and an interface towards Receiver A (RX-A). A static route will suffice and is configured with next hop of the PE-2 PE-CE interface.

```
-----
autonomous-system 64498
route-distinguisher 64498:1
interface "RX-A" create
  address 192.2.1.1/24
  sap 1/1/4:1 create
  exit
exit
interface "int-ce2-pe2" create
  address 172.16.254.253/30
  sap 1/1/1:1 create
  exit
exit
static-route 192.168.1.0/24 next-hop 172.16.254.254
```

PE VPRN Multicast Configuration

This section gives details of the VPRN configuration that allows the support of multicasting.

Sub-sections include:

1. Autodiscovery — This is the mechanism by which each PE advertises the presence of a MVPN to other PEs. This can be achieved using PIM or using BGP. This section covers PIM autodiscovery (autodiscovery using BGP is shown later)
2. Customer Domain signaling — This discusses the mechanism of transporting customer signaling
3. Data Plane connectivity — This is the signaling of S-PMSIs within the provider domain to carry each individual customer multicast stream.

This note discusses the PIM and BGP autodiscovery mechanisms in detail. For each of these, there is an example of customer domain signaling. For completion, a single example of S-PMSI creation is also shown.

Auto-Discovery within Provider Domain Using PIM

Each PE advertises its membership of a multicast VPN using PIM through the configuration of an Inclusive PMSI (I-PMSI). This is a multicast group that is common to each VPRN. The configuration for PE 1 and 2 is shown in the following outputs:

```
*A:PE-1# configure service vprn 1
-----
      mvpn
        provider-tunnel
          inclusive
            pim asm 239.255.255.1
            exit
          exit
        exit
      exit
    no shutdown
...
-----
*A:PE-1#

*A:PE-2# configure service vprn 1
-----
      mvpn
        provider-tunnel
          inclusive
            pim asm 239.255.255.1
            exit
          exit
        exit
      exit
```

```

        exit
        no shutdown
    ...
-----
*A:PE-2#

```

The multicast group address used for the PMSI must be the same on all PEs for this VPRN instance.

Verify that PIM in the Global routing table (GRT) has signalled the I-PMSIs.

For the PE acting as the RP for global PIM:

```

A:PE-1# show router pim group
=====
PIM Groups ipv4
=====
Group Address          Type      Spt Bit Inc Intf      No.Oifs
Source Address         RP
-----
239.255.255.1          (*,G)                3
*                       192.0.2.1
239.255.255.1          (S,G)    spt      system      3
192.0.2.1              192.0.2.1
239.255.255.1          (S,G)    spt      int-pe1-pe2  2
192.0.2.2              192.0.2.1
239.255.255.1          (S,G)    spt      int-pe1-pe3  2
192.0.2.3              192.0.2.1
-----
Groups : 4
=====
A:PE-1#

```

This shows an incoming (S,G) join from all other PEs within the multicast VRF, plus an outgoing (*,G) join to the same PEs.

All other PEs will have the following PIM groups.

```

A:PE-2# show router pim group
=====
PIM Groups ipv4
=====
Group Address          Type      Spt Bit Inc Intf      No.Oifs
Source Address         RP
-----
239.255.255.1          (*,G)                int-pe2-pe1  1
*                       192.0.2.1
239.255.255.1          (S,G)    spt      system      2
192.0.2.2              192.0.2.1
-----
Groups : 2
=====

```

Provider Common Configuration

A:PE-2#

This shows an (S,G) join towards the RP at 192.0.2.1, plus a (*,G) join from RP. These represent the outgoing and incoming PIM interfaces for the VRF.

This results in a series of PIM neighbors through the I-PMSIs within the VRF, which are maintained using PIM hellos.

```
A:PE-1# show router 1 pim neighbor
=====
PIM Neighbor ipv4
=====
Interface          Nbr DR Prty    Up Time      Expiry Time   Hold Time
  Nbr Address
-----
int-pe1-cel        1              1d 02:07:04   0d 00:01:35   105
  172.16.255.253
1-mt-239.255.255.1  1              2d 00:37:32   0d 00:01:23   105
  192.0.2.2
1-mt-239.255.255.1  1              2d 00:37:12   0d 00:01:31   105
  192.0.2.3
-----
Neighbors :3
=====
A:PE-1#
```

PIM Auto-Discovery: Customer Signaling using PIM

Consider now how the signaling plane of the customer domain is dealt with at the provider domain.

The customer domain configuration covers the following three cases:

1. PIM ASM with the RP in the provider PE.
2. PIM ASM using anycast RP on the provider RPs.
3. PIM SSM.

PIM Any Source Multicasting with RP at the Provider PE

As each PE connects to a CE which will be part of the multicast VRF, it is necessary to enable PIM on each interface containing an attachment circuit towards a CE, and to configure the I-PMSI multicast tunnel for the VRF.

There is a requirement for an RP, as customer multicast signaling will be PIM-ASM.

The RP for the customer multicast will be on PE-2. In order to facilitate this, a loopback interface is created (called RP within the VPRN 1 context of PE-2, and will be advertised to all PEs. It must also be a PIM enabled interface.

On PE-2 (the RP):

```
*A:PE-2# configure service vprn 1
*A:PE-2>config>service>vprn# info
-----
autonomous-system 64496
route-distinguisher 64496:1
auto-bind ldp
vrf-target target:64496:1
interface "int-pe2-ce2" create
    address 172.16.254.254/30
    sap 1/1/3:1 create
    exit
exit
interface "rp" create
    address 10.2.3.4/32
    loopback
exit
pim
    interface "int-pe2-ce2"
    exit
    interface "rp"
    exit
    rp
        static
```

```
                group-prefix 224.0.0.0/8
            exit
        exit
    no shutdown
```

The RP must also be configured on each of the PEs and also each CE.

On each of the PEs, the configuration displays as follows:

```
A:PE-1#configure service vprn 1
    pim
        interface "int-pe1-ce1"
        exit
        rp
            static
                address 10.2.3.4
                group-prefix 224.0.0.0/8
            exit
        exit
    no shutdown
```

Customer Edge Router Multicast Configuration

Each CE router will have a PIM neighbor peer relationship with its nearest PE.

The CE router (CE-1) containing the source will have PIM enabled on the interface connected to the source. It will also have a static RP entry, as the incoming sources need to be registered with the RP.

```
A:CE-1# configure service vprn 1
    autonomous-system 64497
    route-distinguisher 64497:1
    interface "int-ce1-pe1" create
        address 172.16.255.253/30
        sap 1/1/1:1 create
        exit
    exit
    interface "to-source" create
        address 192.168.1.1/24
        sap 1/1/3:1 create
        exit
    exit
    pim
        interface "int-ce1-pe1"
        exit
        interface "to-source"
        exit
        rp
            static
                address 10.2.3.4
                group-prefix 224.0.0.0/8
```



```

        exit
    exit
exit
no shutdown

```

The CE containing the receivers will have IGMP enabled on the interface connected to the receivers. Once again, there needs to be an RP configured, as the router needs to issue PIM joins to the RP.

```

A:CE-2# configure service vprn 1
    autonomous-system 64498
    route-distinguisher 64498:1
    interface "RX-A" create
        address 192.2.1.1/24
        sap 1/1/4:1 create
    exit
exit
interface "int-ce2-pe2" create
    address 172.16.254.253/30
    sap 1/1/1:1 create
    exit
exit
static-route 192.168.1.0/24 next-hop 172.16.254.254
static-route 10.0.0.0/8 next-hop 172.16.254.254
igmp
    interface "RX-A"
    exit
exit
pim
    interface "int-ce2-pe2"
    exit
    rp
        static
            address 10.2.3.4
            group-prefix 224.0.0.0/8
        exit
    exit
exit
no shutdown

```

Traffic Flow

The source sends a multicast stream using group address 225.0.0.1 towards CE-1. As the group matches the group address in the static RP configuration, the router sends a register join towards the RP. At this time, no receivers are interested in the group, so there are no entries in the Outgoing Interface List (OIL), and the number of outgoing interfaces (OIFs) is zero.

The PIM status of CE-1 within VPN 1 is as follows:

```
A:CE-1# show router 1 pim group
=====
PIM Groups ipv4
=====
Group Address          Type      Spt Bit Inc Intf      No.Oifs
  Source Address        RP
-----
225.0.0.1              (S,G)                to-source      0
    192.168.1.2        10.2.3.4
-----
Groups : 1
=====
A:CE-1#
```

The receiver A connected to CE-2, wishes to join the group 225.0.0.1, and so sends in an IGMPv2 report towards CE-2. CE-2 recognizes the report, which contains no source.

```
A:CE-2# show router 1 igmp group
=====
IGMP Groups
=====
(*,225.0.0.1)          Up Time : 0d 00:00:33
    Fwd List   : RX-A
-----
(*,G)/(S,G) Entries : 1
=====
A:CE-2#
```

CE-2 is not aware of the source of the group so initiates a (*,G) PIM join towards the RP.

At the RP, a (*,G) join is received

```
A:PE-2# show router 1 pim group 225.0.0.1 type starg detail
=====
PIM Source Group ipv4
=====
Group Address      : 225.0.0.1
Source Address     : *
RP Address         : 10.2.3.4
Flags              :
Type               : (*,G)
MRIB Next Hop      :
MRIB Src Flags     : self
Up Time            : 0d 00:15:41
Keepalive Timer    : Not Running
Resolved By        : rtable-u
```

```

Up JP State      : Joined           Up JP Expiry      : 0d 00:00:18
Up JP Rpt        : Not Joined StarG Up JP Rpt Override : 0d 00:00:00

```

```

Rpf Neighbor      :
Incoming Intf     :
Outgoing Intf List : int-pe2-ce2

```

```

Curr Fwding Rate  : 0.0 kbps
Forwarded Packets : 0                Discarded Packets : 0
Forwarded Octets  : 0                RPF Mismatches    : 0
Spt threshold     : 0 kbps           ECMP opt threshold : 7
Admin bandwidth   : 1 kbps

```

```
-----
Groups : 1

```

The RP can now forward traffic from itself towards CE-2, as the outgoing interface is seen as int-pe2-ce2.

CE-2 is now able to determine the source from the traffic stream, so it initiates a Reverse Path Forwarding (RPF) lookup of the source address in the route table, and issues an (S,G) PIM join towards the source.

The join is propagated across the provider network, from PE-2 towards PE-1 which is the resolved RPF next hop for the source.

```

A:PE-1# show router 1 pim group detail
=====
PIM Source Group ipv4
=====
Group Address      : 225.0.0.1
Source Address     : 192.168.1.2
RP Address         : 172.16.254.254
Flags              : spt                Type              : (S,G)
MRIB Next Hop     : 172.16.255.253
MRIB Src Flags     : remote             Keepalive Timer Exp: 0d 00:01:43
Up Time           : 0d 00:08:47         Resolved By        : rtable-u

Up JP State        : Joined             Up JP Expiry        : 0d 00:00:13
Up JP Rpt          : Not Joined StarG   Up JP Rpt Override  : 0d 00:00:00

Register State     : No Info
Reg From Anycast RP: No

Rpf Neighbor       : 172.16.255.253
Incoming Intf      : int-pe1-ce1
Outgoing Intf List : 1-mt-239.255.255.1

Curr Fwding Rate   : 33.6 kbps
Forwarded Packets  : 52214              Discarded Packets   : 0
Forwarded Octets   : 2192988            RPF Mismatches      : 0
Spt threshold      : 0 kbps             ECMP opt threshold  : 7
Admin bandwidth    : 1 kbps

```

```
-----
Groups : 1

```

Note that the outgoing interface is the I-PMSI.

The join is received by CE-1, which contains the subnet of the source.

CE-1 now recognizes the multicast group as a valid stream. This becomes the root of the shortest path tree for the group.

```
A:CE-1# show router 1 pim group
=====
PIM Groups ipv4
=====
Group Address          Type      Spt Bit Inc Intf      No.Oifs
Source Address         RP
-----
225.0.0.1              (S,G)    spt      to-source  1
192.168.1.2
-----
Groups : 1
=====
A:CE-1#
```

For completion, consider a second receiver B interested in group 225.0.0.1. The IGMP V2 report is translated into a (*,G) PIM join at CE-3 towards the RP.

```
A:CE-3# show router 1 pim group type starg
=====
PIM Groups ipv4
=====
Group Address          Type      Spt Bit Inc Intf      No.Oifs
Source Address         RP
-----
225.0.0.1              (*,G)    int-ce3-pe3  1
*                      10.2.3.4
-----
Groups : 1
```

At the RP (PE-2), there is now a second interface in the OIL,

```
A:PE-2# show router 1 pim group 225.0.0.1 type starg detail
=====
PIM Source Group ipv4
=====
Group Address      : 225.0.0.1
Source Address     : *
RP Address         : 10.2.3.4
Flags              :                               Type           : (*,G)
MRIB Next Hop      :
MRIB Src Flags     : self                       Keepalive Timer      : Not Running
Up Time            : 0d 00:24:38                 Resolved By          : rtable-u

Up JP State        : Joined                       Up JP Expiry         : 0d 00:00:21
Up JP Rpt          : Not Joined StarG             Up JP Rpt Override   : 0d 00:00:00

Rpf Neighbor       :
Incoming Intf      :
Outgoing Intf List : int-pe2-ce2, 1-mt-239.255.255.1

Curr Fwding Rate   : 0.0 kbps
Forwarded Packets  : 0                           Discarded Packets    : 0
Forwarded Octets   : 0                           RPF Mismatches       : 0
Spt threshold      : 0 kbps                       ECMP opt threshold   : 7
Admin bandwidth    : 1 kbps
-----
Groups : 1
```

The second interface is the I-PMSI, which is the multicast tunnel towards all other PEs. At PE-3, the (*,G) join has the I-PMSI as an incoming interface, and the PE-CE interface as the outgoing interface.

```
A:PE-3# show router 1 pim group type starg detail
=====
PIM Source Group ipv4
=====
Group Address      : 225.0.0.1
Source Address     : *
RP Address         : 10.2.3.4
Flags              :                               Type           : (*,G)
MRIB Next Hop      : 192.0.2.2
MRIB Src Flags     : remote                       Keepalive Timer      : Not Running
Up Time            : 0d 00:04:10                 Resolved By          : rtable-u

Up JP State        : Joined                       Up JP Expiry         : 0d 00:00:50
Up JP Rpt          : Not Joined StarG             Up JP Rpt Override   : 0d 00:00:00

Rpf Neighbor       : 192.0.2.2
Incoming Intf      : 1-mt-239.255.255.1
Outgoing Intf List : int-pe3-ce3

Curr Fwding Rate   : 0.0 kbps
Forwarded Packets  : 1                           Discarded Packets    : 0
Forwarded Octets   : 42                           RPF Mismatches       : 0
Spt threshold      : 0 kbps                       ECMP opt threshold   : 7
```

PIM Auto-Discovery: Customer Signaling using PIM

```
Admin bandwidth      : 1 kbps
-----
Groups : 1
=====
A:PE-3#
```

Once again, as the CE receives traffic from the group, it can use the source address in the packet to initiate an (S,G) join towards the source ?it joins the shortest path tree

```
A:CE-3# show router 1 pim group type sg detail
=====
PIM Source Group ipv4
=====
Group Address       : 225.0.0.1
Source Address      : 192.168.1.2
RP Address          : 10.2.3.4
Flags               : spt                               Type                : (S,G)
MRIB Next Hop       : 172.16.253.254
MRIB Src Flags      : remote                           Keepalive Timer Exp: 0d 00:02:44
Up Time             : 0d 00:07:48                       Resolved By         : rtable-u

Up JP State         : Joined                             Up JP Expiry        : 0d 00:00:12
Up JP Rpt           : Not Pruned                         Up JP Rpt Override  : 0d 00:00:00

Register State      : No Info
Reg From Anycast RP: No

Rpf Neighbor        : 172.16.253.254
Incoming Intf       : int-ce3-pe3
Outgoing Intf List  : RX-B

Curr Fwding Rate    : 16.9 kbps
Forwarded Packets   : 23303                             Discarded Packets   : 0
Forwarded Octets    : 978726                             RPF Mismatches      : 0
Spt threshold       : 0 kbps                             ECMP opt threshold  : 7
Admin bandwidth     : 1 kbps
-----
Groups : 1
=====
A:CE-3#
```

PIM Any Source Multicasting with Anycast RP at the Provider PE

The network topology is displayed in [Figure 204](#). The setup consists of 4 x 7750s acting as Provider Edge (PE) routers within a single Autonomous System (AS).

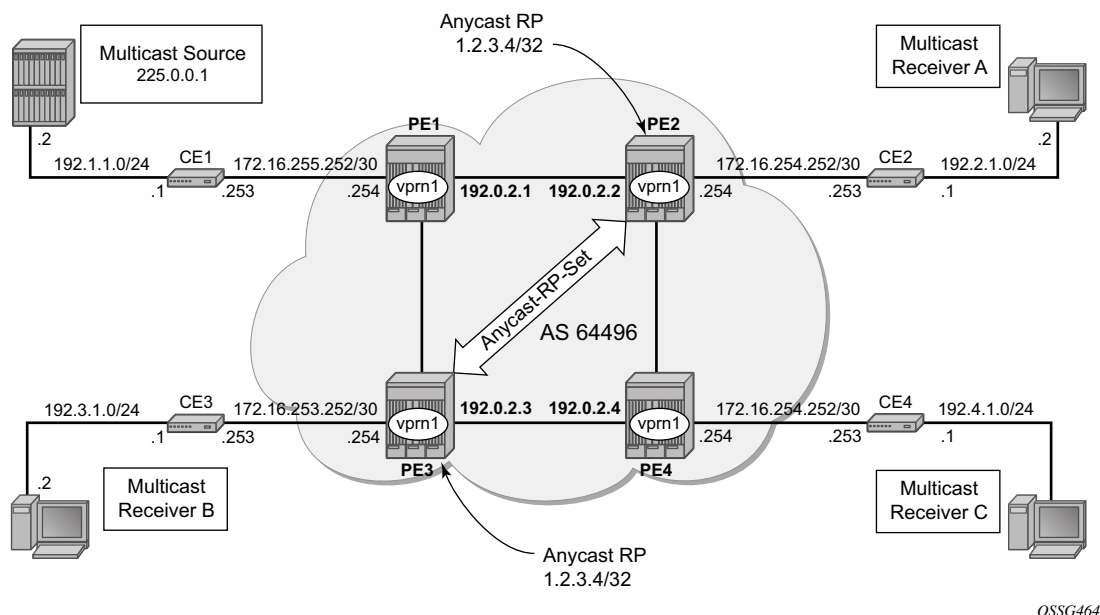


Figure 204: Network Topology for Anycast RP

Connected to each PE is a single 7750 acting as a Customer Edge (CE) router. CE-1 has a single multicast source connected, and PEs 2 to 4 each have a single receiver connected which will receive the multicast stream from the source. In this section, each receiver is IGMPv2 capable, and so will issue IGMPv2 reports. A Rendezvous Point is required by the C-signaling plane to resolve each (*,G) group state into an (S,G) state. In this case, two RPs are chosen to form an Anycast set to resolve each (*,G) group into an (S,G) state.

Multicast traffic from the source group 225.0.0.1 is streamed toward router CE-1. Receivers connected to PE-2, PE-3 and PE-4 are interested in joining this multicast group.

Anycast RP - PE VPRN Configuration

As each PE contains a CE which will be part of the multicast VRF, it is necessary to enable PIM on each interface containing an attachment circuit towards a CE, and to configure the I-PMSI multicast tunnel for the VRF.

As previously stated, there is a requirement for an RP, as Customer Multicast signaling will be PIM-ASM and IGMPv2.

In this case, an Anycast RP will be used. This is configured on PE-2 and PE-3, and an anycast set is created.

As each PE contains a CE which will be part of the multicast VRF, it is necessary to enable PIM on each interface containing an attachment circuit towards a CE, and to configure the I-PMSI multicast tunnel for the VRF.

The following output shows the VPRN configuration for PE-2 containing the RP and anycast RP configuration:

```
A:PE-2#configure service vprn 1
    interface "rp" create
        address 10.2.3.5/32
        loopback
    exit
    interface "lo1" create
        address 10.0.0.2/32
        loopback
    exit
    pim
        interface "int-pe2-ce2"          #Attachment circuit towards CE
    exit
        interface "rp"                  #RP interface
    exit
        interface "lo1"                #loopback interface for inter RP communication
    exit
    rp
        static
            address 10.2.3.5
            group-prefix 225.0.0.0/8
        exit
    exit
    anycast 10.2.3.5                    #Anycast RP IP address
        rp-set-peer 10.0.0.2           #IP address of THIS router
        rp-set-peer 10.0.0.3           #IP address of peer router
    exit
    exit
    exit
    mvpn
        provider-tunnel
            inclusive
            pim asm 239.255.255.1
        exit
    exit
```



```

        exit
    exit
no shutdown

```

Similarly, the VPRN configuration for PE-3 is:

```

A:PE-3#configure service vprn 1
    interface "rp" create
        address 10.2.3.5/32
        loopback
    exit
    interface "lo1" create
        address 10.0.0.3/32
        loopback
    exit
    pim
        interface "int-pe3-ce3" #Attachment circuit towards CE
        exit
        interface "rp"          #RP interface
        exit
        interface "lo1"          #loopback interface for inter RP communication
        exit
        rp
            static
                address 10.2.3.5
                group-prefix 225.0.0.0/8
            exit
            anycast 10.2.3.5      #Anycast RP IP address
            rp-set-peer 10.0.0.2  #IP address of THIS router
            rp-set-peer 10.0.0.3  #IP address of peer router
        exit
    exit
    mvpn
        provider-tunnel
            inclusive
            pim asm 239.255.255.1
        exit
    exit
exit
no shutdown

```

As previously stated, there is a requirement for an RP, as customer multicast signaling will be PIM-ASM and IGMPv2.

In this case, an anycast RP will be used. This is configured on PE-2 and PE-3, and an anycast set is created.

The anycast address will be 10.2.3.5/32 and is created as an interface called **rp** on both PE-2 and PE-3.

An additional loopback interface, called **lo1** is created on each VPRN on PEs containing the anycast address. These are used as source addresses for communication between the routers within the RP set. These addresses will be automatically advertised to all PEs as vpn-ipv4 addresses, and will be installed in the VRF 1 forwarding table of all PEs containing VPRN 1.

Note: All routers containing RP must have their own loopback address included in the RP set as well as all peer routers.

The multicast group address used for the Inclusive PMSI is chosen to be 239.255.255.1 and must be the same on all PEs for this VPRN instance. This is analogous to the MDT within the Draft-Rosen implementation.

Verify that PIM in the global routing table (GRT) has signalled the I-PMSIs.

For the PE acting as the RP for global PIM:

```
A:PE-1# show router pim group
=====
PIM Groups ipv4
=====
Group Address          Type      Spt Bit Inc Intf      No.Oifs
  Source Address      RP
-----
239.255.255.1          (*,G)                3
   *                192.0.2.1
239.255.255.1          (S,G)    spt      system    3
   192.0.2.1        192.0.2.1
239.255.255.1          (S,G)    spt      int-pe1-pe2  2
   192.0.2.2        192.0.2.1
239.255.255.1          (S,G)    spt      int-pe1-pe3  2
   192.0.2.3        192.0.2.1
-----
Groups : 4
=====
A:PE-1#
```

All other PEs will have:

```
=====
PIM Groups ipv4
=====
Group Address          Type      Spt Bit Inc Intf      No.Oifs
  Source Address      RP
-----
239.255.255.1          (*,G)                int-pe2-pe1      1
*                      192.0.2.1
239.255.255.1          (S,G)    spt      system        2
192.0.2.2              192.0.2.1
-----
Groups : 2
=====
```

This shows a (S,G) join towards the RP at 192.0.2.1, plus a (*,G) join from RP. These represent the outgoing and incoming PIM interfaces for the VRF.

This results in a series of PIM neighbors through the I-PMSIs within the VRF, which are maintained using PIM hellos.

```
A:PE-1# show router 1 pim neighbor
=====
PIM Neighbor ipv4
=====
Interface          Nbr DR Prty    Up Time      Expiry Time   Hold Time
  Nbr Address
-----
int-pe1-cel        1              1d 02:07:04   0d 00:01:35   105
  172.16.255.253
1-mt-239.255.255.1 1              2d 00:37:32   0d 00:01:23   105
  192.0.2.2
1-mt-239.255.255.1 1              2d 00:37:12   0d 00:01:31   105
  192.0.2.3
-----
Neighbors : 3
=====
A:PE-1#
```

Verify PIM RP set:

```
*A:PE-2# show router 1 pim anycast
=====
PIM Anycast RP Entries ipv4
=====
Anycast RP                               Anycast RP Peer
-----
10.2.3.5                                10.0.0.2
                                         10.0.0.3
-----
PIM Anycast RP Entries : 2
=====
*A:PE-2#
```

Anycast RP — Customer Edge Router Multicast Configuration

Each CE router will have a PIM neighbor peer relationship with its nearest PE.

The CE router (CE-1) containing the source will have PIM enabled on the interface connected to the source.

```
A:CE-1# configure service vprn 1
    autonomous-system 64497
    route-distinguisher 64497:1
    interface "int-ce1-pe1" create
        address 172.16.255.253/30
        sap 1/1/1:1 create
        exit
    exit
    interface "to-source" create
        address 192.168.1.1/24
        sap 1/1/3:1 create
        exit
    exit
    pim
        interface "int-ce1-pe1"
        exit
        interface "to-source"
        exit
        rp
            static
                address 10.2.3.5
                group-prefix 225.0.0.0/8
            exit
        exit
    no shutdown
```

The CE containing the receivers will have IGMP enabled on the interface connected to the receivers.

```
A:CE-2# configure service vprn 1
      autonomous-system 64498
      route-distinguisher 64498:1
      interface "to-104/2" create
        address 192.168.2.1/24
        sap 1/1/4:1 create
      exit
    exit
  interface "int-ce2-pe2" create
    address 172.16.254.253/30
    sap 1/1/1:1 create
  exit
exit
static-route 0.0.0.0/0 next-hop 172.16.254.254
igmp
  interface "RX-A"
  exit
exit
no shutdown
```

Traffic Flow

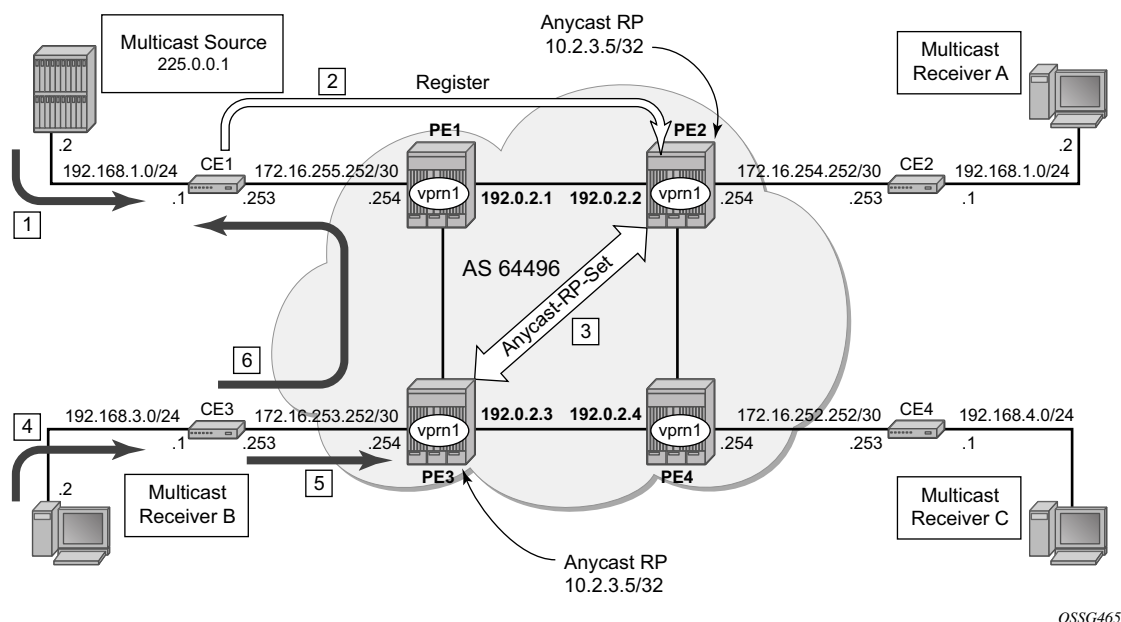


Figure 205: IGMP and PIM Control Messaging Schematic

Figure 205 shows the sequence of IGMP and PIM control messaging.

1. The source multicasts a stream with group address 225.0.0.1 towards CE-1.
2. CE-1 matches the group with the group address prefix in the static RP configuration and sends a register message towards the RP.

```
A:CE-1# show router 1 pim group detail
=====
PIM Source Group ipv4
=====
Group Address      : 225.0.0.1
Source Address     : 192.168.1.2
RP Address         : 10.2.3.5
Flags              :
Type               : (S,G)
MRIB Next Hop     : 192.168.1.2
MRIB Src Flags     : direct
Up Time           : 0d 00:00:10
Keepalive Timer Exp: 0d 00:03:19
Resolved By       : rtable-u

Up JP State        : Not Joined
Up JP Rpt          : Not Joined StarG
Up JP Expiry       : 0d 00:00:00
Up JP Rpt Override : 0d 00:00:00

Register State     : Pruned
Register Stop Exp  : 0d 00:00:38
Reg From Anycast RP: No

Rpf Neighbor       : 192.168.1.2
```

```

Incoming Intf      : to-source
Outgoing Intf List :

Curr Fwding Rate   : 0.0 kbps
Forwarded Packets   : 292
Forwarded Octets    : 12264
Spt threshold       : 0 kbps
Admin bandwidth     : 1 kbps
Discarded Packets   : 0
RPF Mismatches      : 0
ECMP opt threshold  : 7
-----
Groups : 1
=====
A:CE-1#

```

The register message is sent to the nearest RP, the RP with the lowest IGP cost.

As the Register is sent through PE-1, it is PE-1 that determines which RP will receive the message.

```

A:PE-1# show router 1 route-table 10.2.3.5/32
=====
Route Table (Service: 1)
=====
Dest Prefix          Next Hop[Interface Name]      Type   Proto   Age      Pref
-----
10.2.3.5/32          192.0.2.2                     Remote BGP VPN 19h49m49s 170
                                     0
-----
No. of Routes: 1
=====
A:PE-1#

```

The PE which will receive the register is 192.0.2.2 PE-2. The PIM group status on PE-2 is:

```

*A:PE-2# show router 1 pim group
=====
PIM Groups ipv4
=====
Group Address          Type      Spt Bit Inc Intf      No.Oifs
Source Address          RP
-----
225.0.0.1              (S,G)          1-mt-239.255.* 0
192.168.1.2            10.2.3.5
-----
Groups : 1
=====
*A:PE-2#

```

This shows that RP is aware of the (S,G) status of the group 225.1.1.1, and becomes a root of a shared tree for this group. Note that the Outgoing Interface List (OIL) is empty.

3. PE-2 will now send a register message to all other RPs within the anycast set. In this case to PE-3 (which has VPRN 1 containing address 10.0.0.3).

The PIM status of the group 225.0.0.1 on PE-3 is:

```
A:PE-3# show router 1 pim group
=====
PIM Groups ipv4
=====
Group Address          Type      Spt Bit Inc Intf      No.Oifs
  Source Address          RP
-----
225.0.0.1              (S,G)          1-mt-239.255.* 0
  192.168.1.2          10.2.3.4
-----
Groups : 1
=====
* indicates that the corresponding row element may have been truncated.
A:PE-3#
```

Now both PEs within the RP set for VPRN have an (S,G) state for 225.0.0.1.

4. The receiver B, wishes to join the group 225.0.0.1, and so sends in an IGMPv2 report towards CE-3. CE-3 recognizes the report, but has no PIM state for this group.
5. It sends a PIM join towards the RP, in this case the nearest RP will be PE-3.

PE-3 already has (S,G) state for this group, so will forward traffic towards receiver B.

6. CE-3 does a Reverse Path Forwarding (RPF) lookup of the source address in the route table, and issues a PIM join towards the source.

The join is propagated across the provider network, towards PE-1 which is the resolved RPF next hop for the source.

```
A:CE-3# show router 1 pim group type sg detail
=====
PIM Source Group ipv4
=====
Group Address      : 225.0.0.1
Source Address     : 192.168.1.2
RP Address         : 10.2.3.5
Flags              : spt                      Type              : (S,G)
MRIB Next Hop      : 172.16.253.254
MRIB Src Flags     : remote                   Keepalive Timer Exp: 0d 00:02:04
Up Time            : 0d 00:01:28               Resolved By         : rtable-u

Up JP State        : Joined                    Up JP Expiry         : 0d 00:00:32
Up JP Rpt          : Not Pruned                 Up JP Rpt Override  : 0d 00:00:00
```



```

Register State      : No Info
Reg From Anycast RP: No

Rpf Neighbor       : 172.16.253.254
Incoming Intf      : int-ce3-pe3
Outgoing Intf List : RX-B

Curr Fwding Rate    : 16.8 kbps
Forwarded Packets   : 3920
Forwarded Octets    : 164640
Spt threshold       : 0 kbps
Admin bandwidth     : 1 kbps
Discarded Packets   : 0
RPF Mismatches      : 0
ECMP opt threshold  : 7
-----
Groups : 1
=====
A:CE-3#

```

The join is received by CE-1, which contains the subnet of the source.

CE-1 now recognizes the multicast group as a valid stream. This becomes the root of the shortest path tree for the group.

```

A:CE-1# show router 1 pim group
=====
PIM Groups ipv4
=====
Group Address      Type      Spt Bit Inc Intf      No.Oifs
  Source Address      RP
-----
225.0.0.1          (S,G)    spt      to-source    1
  192.168.1.2
-----
Groups : 1
=====
A:CE-1#

```

PIM Source-Specific Multicasting

There is no requirement for an RP, as customer multicast signaling will be PIM-SSM. The Multicast group address used for the PMSI must be the same on all PEs for this VPRN instance.

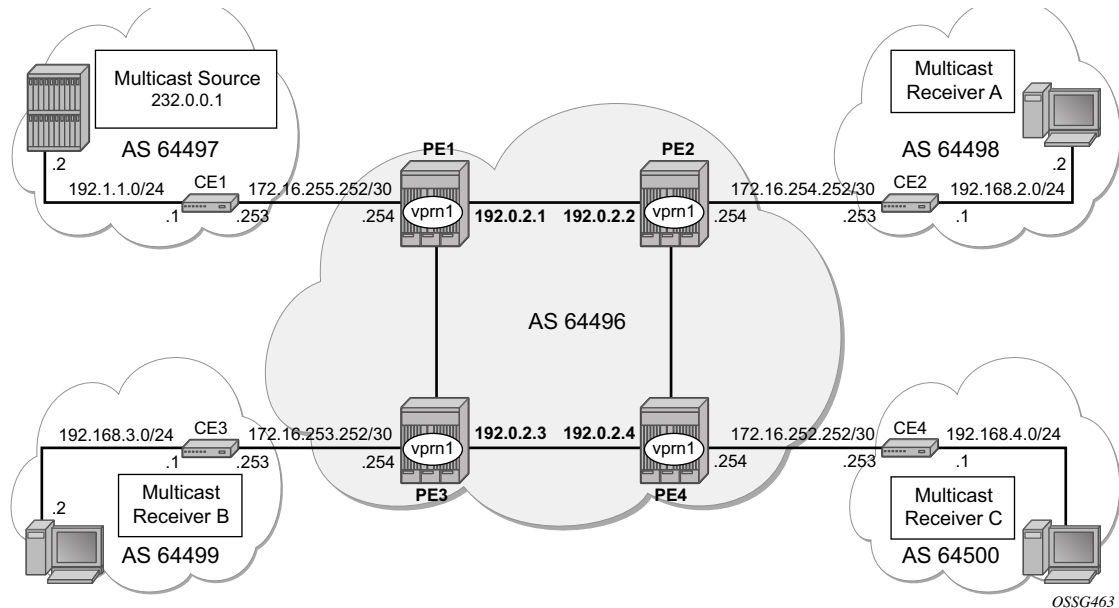


Figure 206: PIM SSM in Customer Signaling Plane

The following output shows the VPRN configuration for PIM and MVPN for PE-1.

```
A:PE-1#configure service vprn 1
  pim
    interface "int-pe1-ce1"
    exit
  mvpn
    provider-tunnel
    inclusive
    pim asm 239.255.255.1
    exit
  exit
exit
no shutdown
```

There is a similar configuration required for each of the other PEs. Verify that PIM in the Global routing table (GRT) has signalled the I-PMSIs.

For the PE acting as the RP for global PIM:

```
A:PE-1# show router pim group
=====
PIM Groups ipv4
=====
Group Address          Type      Spt Bit Inc Intf      No.Oifs
  Source Address      RP
-----
239.255.255.1          (*,G)                3
*                      192.0.2.1
239.255.255.1          (S,G)    spt      system      3
  192.0.2.1            192.0.2.1
239.255.255.1          (S,G)    spt      int-pe1-pe2  2
  192.0.2.2            192.0.2.1
239.255.255.1          (S,G)    spt      int-pe1-pe3  2
  192.0.2.3            192.0.2.1
-----
Groups : 4
=====
A:PE-1#
```

All other PEs will have:

```
A:PE-2# show router pim group
=====
PIM Groups ipv4
=====
Group Address          Type      Spt Bit Inc Intf      No.Oifs
  Source Address      RP
-----
239.255.255.1          (*,G)                int-pe2-pe1  1
*                      192.0.2.1
239.255.255.1          (S,G)    spt      system      2
  192.0.2.2            192.0.2.1
-----
Groups : 2
=====
A:PE-2#
```

This shows a (S,G) join towards the RP at 192.0.2.1, plus a (*,G) join from RP. These represent the outgoing and incoming PIM interfaces for the VRF.

This results in a series of PIM neighbors through the I-PMSIs within the VRF, which are maintained using PIM hellos.

```
A:PE-1# show router 1 pim neighbor
=====
PIM Neighbor ipv4
=====
Interface          Nbr DR Prty    Up Time    Expiry Time    Hold Time
```

PIM Auto-Discovery: Customer Signaling using PIM

```
      Nbr Address
-----
int-pe1-cel          1          1d 02:07:04   0d 00:01:35   105
172.16.255.253
1-mt-239.255.255.1   1          2d 00:37:32   0d 00:01:23   105
192.0.2.2
1-mt-239.255.255.1   1          2d 00:37:12   0d 00:01:31   105
192.0.2.3
-----
Neighbors : 3
=====
A:PE-1#
```

PIM SSM — Customer Edge Router Multicast Configuration

Each CE router will have a PIM neighbor peer relationship with its nearest PE.

The CE router (CE-1) containing the source will have PIM enabled on the interface connected to the source.

```
A:CE-1# configure service vprn 1
      autonomous-system 64497
      route-distinguisher 64497:1
      interface "int-cel-pe1" create
        address 172.16.255.253/30
        sap 1/1/1:1 create
        exit
      exit
      interface "to-source" create
        address 192.168.1.1/24
        sap 1/1/3:1 create
        exit
      exit
      pim
        interface "int-cel-pe1"
        exit
        interface "to-source"
        exit
      no shutdown
```

The CE containing the receivers will have IGMP enabled on the interface connected to the receivers and PIM on the interface facing the PE.

```
A:CE-2# configure service vprn 1
      autonomous-system 64498
      route-distinguisher 64498:1
      interface "RX-A" create
        address 192.2.1.1/24
        sap 1/1/4:1 create
        exit
      exit
      interface "int-ce2-pe2" create
        address 172.16.254.253/30
        sap 1/1/1:1 create
        exit
      exit
      static-route 192.168.1.0/24 next-hop 172.16.254.254
      igmp
        interface "RX-A"
        exit
      exit
      pim
        interface "int-ce2-pe2"
        exit
      no shutdown
```

Traffic Flow

The source multicasts a stream towards CE-1. As there is no receiver interested in the group at this time, there are no outgoing interfaces, so the Outgoing Interface List (OIL) is empty.

```
A:CE-1# show router 1 pim group
=====
PIM Groups ipv4
=====
Group Address          Type      Spt Bit Inc Intf      No.Oifs
Source Address          RP
-----
232.0.0.1              (S,G)          to-104/1      0
192.168.1.2
-----
Groups : 1
=====
A:CE-1#
```

The receiver A, wishes to join the group 232.0.0.1, and so sends in an IGMPv3 report towards CE-2. CE-2 recognizes the report, which contains the source 192.168.1.2 in the INCLUDE filter list.

```
A:CE-2# show router 1 igmp group
=====
IGMP Groups
=====
(192.168.1.2,232.0.0.1)      Up Time : 0d 00:00:33
Fwd List : to-104/2
-----
(*,G)/(S,G) Entries : 1
=====
A:CE-2#
```

CE-2 does a RPF lookup of the source address in the route table, and issues a PIM join towards the source.

The join is propagated across the provider network, towards PE-1 which is the resolved RPF next hop for the source.

```
A:PE-1# show router 1 pim group detail
=====
PIM Source Group ipv4
=====
Group Address      : 232.0.0.1
Source Address     : 192.168.1.2
RP Address         : 172.16.254.254
Flags              : spt                      Type              : (S,G)
MRIB Next Hop      : 172.16.255.253
MRIB Src Flags     : remote                  Keepalive Timer Exp: 0d 00:01:43
Up Time            : 0d 00:08:47              Resolved By         : rtable-u

Up JP State        : Joined                   Up JP Expiry        : 0d 00:00:13
Up JP Rpt          : Not Joined StarG         Up JP Rpt Override  : 0d 00:00:00
```

```

Register State      : No Info
Reg From Anycast RP: No

Rpf Neighbor        : 172.16.255.253
Incoming Intf       : int-pe1-ce1
Outgoing Intf List  : 1-mt-239.255.255.1

Curr Fwding Rate    : 33.6 kbps
Forwarded Packets    : 52214
Forwarded Octets     : 2192988
Spt threshold       : 0 kbps
Admin bandwidth     : 1 kbps
Discarded Packets    : 0
RPF Mismatches       : 0
ECMP opt threshold  : 7
-----
Groups : 1

```

Note that the outgoing interface is the I-PMSI.

The join is received by CE-1, which contains the subnet of the source.

CE-1 now recognizes the multicast group as a valid stream. This becomes the root of the shortest path tree for the group.

```

A:CE-1# show router 1 pim group
=====
PIM Groups ipv4
=====
Group Address          Type      Spt Bit Inc Intf      No.Oifs
Source Address         RP
-----
232.0.0.1              (S,G)   spt      to-source    1
192.168.1.2
-----
Groups : 1
=====
A:CE-1#

```

PE BGP Auto-Discovery

Discovery of multicast-enabled Virtual Private Networks (MVPNs) can also be achieved using BGP. To this end, any PE that is a member of a Multicast VPN will advertise this using a BGP Multi-protocol Reachable Next-Hop Router Layer Information (NRLI) update that is sent to all PEs within the AS. This update will contain an Intra-AS I-PMSI auto-discovery Route type, also known as an Intra-AD. These use an address family, mvpn-ipv4, so each PE must be configured to originate and accept such updates.

This is achieved in the global routing table within the BGP context.

```
A:PE-1#configure router bgp
      group "internal"
        family vpn-ipv4 mvpn-ipv4
        peer-as 64496
        neighbor 192.0.2.2
        exit
        neighbor 192.0.2.3
        exit
        neighbor 192.0.2.4
        exit
      exit
```

This allows each BGP speaker to advertise its capabilities within a BGP Open message.

When the peers become established, the address family capabilities should look like the following:

```
A:PE-1# show router bgp summary
=====
BGP Router ID:192.0.2.1      AS:64496      Local AS:64496
=====
BGP Admin State      : Up      BGP Oper State      : Up
Total Peer Groups    : 1      Total Peers          : 3
Total BGP Paths      : 15     Total Path Memory    : 1932
Total IPv4 Remote Rts : 0      Total IPv4 Rem. Active Rts : 0
Total IPv6 Remote Rts : 0      Total IPv6 Rem. Active Rts : 0
Total Suppressed Rts  : 0      Total Hist. Rts      : 0
Total Decay Rts       : 0

Total VPN Peer Groups : 1      Total VPN Peers      : 1
Total VPN Local Rts   : 2
Total VPN-IPv4 Rem. Rts : 7    Total VPN-IPv4 Rem. Act. Rts: 6
Total VPN-IPv6 Rem. Rts : 0    Total VPN-IPv6 Rem. Act. Rts: 0
Total L2-VPN Rem. Rts  : 0      Total L2VPN Rem. Act. Rts : 0
Total VPN Supp. Rts    : 0      Total VPN Hist. Rts    : 0
Total VPN Decay Rts    : 0
Total MVPN-IPv4 Rem Rts : 3    Total MVPN-IPv4 Rem Act Rts : 3
=====
BGP Summary
=====
Neighbor
```

AS	PktRcvd	InQ	Up/Down	State	Rcv/Act/Sent	(Addr Family)

```

PktSent OutQ
```

192.0.2.2	64496	26629	0	07d01h59m	3/3/1 (VpnIPv4)
		26737	0		1/1/1 (MvpnIpv4)
192.0.2.3	64496	26647	0	07d01h59m	3/2/1 (VpnIPv4)
		26728	0		1/1/1 (MvpnIpv4)
192.0.2.4	64496	26632	0	07d01h59m	1/1/1 (VpnIPv4)
		26685	0		1/1/1 (MvpnIpv4)
=====					
A:PE-1#					

BGP Auto-Discovery — PE VPRN Multicast Configuration

As each PE contains a CE which will be part of the multicast VRF, it is necessary to enable PIM on each interface containing an attachment circuit towards a CE, and to configure the I-PMSI multicast tunnel for the VRF.

In order for the BGP routes to be accepted into the VRF, a route-target community is required (vrf-target). This is configured in the **configure service vprn 1 mvpn** context and, in this case, is set to the same value as the unicast vrf-target, the vrf-target community as the **configure service vprn 1 vrf-target** context.

On each PE, A VPRN instance is configured as follows

```
A:PE-2# configure service vprn 1
      autonomous-system 64496
      route-distinguisher 64496:1
      auto-bind ldp
      vrf-target target:64496:1
      interface "int-pe2-ce2" create
        address 172.16.254.254/30
        sap 1/1/3:1 create
      exit
    exit
  pim
    interface "int-pe2-ce2"
    exit
  mvpn
    auto-discovery
    provider-tunnel
      inclusive
        pim asm 239.255.255.1
      exit
    exit
  exit
  vrf-target unicast
  exit
exit
no shutdown
```

The multicast group address used for the PMSI must be the same on all PEs for this VPRN instance.

The presence of auto-discovery will initiate BGP updates between the PEs that contain an MVPN, such as Intra-AD MVPN routes, are generated and advertised to each peer

```
A:PE-1# show router bgp routes mvpn-ipv4
=====
BGP Router ID:192.0.2.1      AS:64496      Local AS:64496
=====
Legend -
Status codes  : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best
=====
```

```

BGP MVPN-IPv4 Routes
=====
Flag   RouteType      OriginatorIP      LocalPref  MED
      RD          SourceAS          VPNLabel
      Nexthop      SourceIP
      As-Path      GroupIP
-----
u*>i   Intra-Ad        192.0.2.2         100        0
      64496:1        -                  -
      192.0.2.2      -
      No As-Path     -
u*>i   Intra-Ad        192.0.2.3         100        0
      64496:1        -                  -
      192.0.2.3      -
      No As-Path     -
u*>i   Intra-Ad        192.0.2.4         100        0
      64496:1        -                  -
      192.0.2.4      -
      No As-Path     -
-----
Routes : 3
=====
*A:PE-1#

```

This shows that PE-1 has received an Intra-AD route from each of the other PEs, each of which has multicast VPRN 1 configured.

Examining one of the Intra-AD routes from PE-2 shows that the route-target community matches the unicast VRF-target (64496:1), and also that the PMSI tree has a multicast group address of 239.255.255.1, which matches the I-PMSI group configuration on PE-1.

```

A:PE-1# show router bgp routes mvpn-ipv4 type intra-ad originator-ip 192.0.2.2 detail
=====
BGP Router ID:192.0.2.1      AS:64496      Local AS:64496
=====
Legend -
Status codes  : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best
=====
BGP MVPN-IPv4 Routes
=====
Route Type      : Intra-Ad
Route Dist.     : 64496:1
Originator IP   : 192.0.2.2
Nextthop       : 192.0.2.2
From            : 192.0.2.2
Res. Nextthop   : 0.0.0.0
Local Pref.     : 100
Aggregator AS   : None
Atomic Aggr.    : Not Atomic
Community       : target:64496:1
Cluster         : No Cluster Members
Originator Id   : None
Flags           : Used Valid Best IGP
AS-Path         : No As-Path
Peer Router Id  : 192.0.2.2
Interface Name  : NotAvailable
Aggregator      : None
MED             : 0
-----
PMSI Tunnel Attribute :
Tunnel-type      : PIM-SM Tree
MPLS Label       : 0x0
Sender           : 192.0.2.2
Flags            : Local Info not Required
P-Group          : 239.255.255.1
-----
Routes : 1
=====
A:PE-1#

```

Verify that PIM in the global routing table (GRT) has signalled the I-PMSIs.

For the PE acting as the RP for global PIM:

```

A:PE-1# show router pim group
=====
PIM Groups ipv4
=====
Group Address      Type      Spt Bit Inc Intf      No.Oifs
Source Address      RP
-----
239.255.255.1      (*,G)
*                  192.0.2.1      3
239.255.255.1      (S,G)
192.0.2.1          192.0.2.1      spt      system      3
239.255.255.1      (S,G)
192.0.2.2          192.0.2.1      spt      int-pe1-pe2  2
239.255.255.1      (S,G)
192.0.2.3          192.0.2.1      spt      int-pe1-pe3  2
-----
Groups : 4
=====

```

```
A:PE-1#
```

This shows an incoming (S,G) join from all other PEs within the multicast VRF, plus an outgoing (*,G) join to the same PEs.

All other PEs will have the following PIM groups

```
A:PE-2# show router pim group
=====
PIM Groups ipv4
=====
Group Address          Type      Spt Bit Inc Intf      No.Oifs
  Source Address          RP
-----
239.255.255.1          (*,G)                int-pe2-pe1      1
*                        192.0.2.1
239.255.255.1          (S,G)    spt      system          2
  192.0.2.2            192.0.2.1
-----
Groups : 2
=====
A:PE-2#
```

This shows a (S,G) join towards the RP at 192.0.2.1, plus a (*,G) join from RP. These represent the outgoing and incoming PIM interfaces for the VRF.

This results in a series of PIM neighbors through the I-PMSIs within the VRF. As the neighbors were discovered using BGP (rather than with PIM as per Draft-Rosen), there are no PIM hellos exchanged.

```
A:PE-1# show router 1 pim neighbor
=====
PIM Neighbor ipv4
=====
Interface              Nbr DR Prty    Up Time          Expiry Time      Hold Time
  Nbr Address
-----
int-pe1-cel            1                1d 01:18:59      0d 00:01:24      105
  172.16.255.253
1-mt-239.255.255.1      1                1d 01:18:22      never              65535
  192.0.2.2
1-mt-239.255.255.1      1                1d 01:18:59      never              65535
  192.0.2.3
1-mt-239.255.255.1      1                1d 01:18:59      never              65535
  192.0.2.4
-----
Neighbors : 4
=====
*A:PE-1#
```

BGP Auto-Discovery — Customer Signaling Domain

As the customer signaling is independent from the provider PE discovery mechanism, all of the customer signaling techniques described when using PIM for auto-discovery within provider domain are also applicable when using BGP for auto-discovery, namely

- PIM Any Source Multicasting with RP at the provider PE
- PIM Any Source Multicasting with Anycast RP at the provider PE
- PIM Source Specific Multicasting

Data Path Using Selective PMSI

When a configurable data threshold for a multicast group has been exceeded, multicast traffic across the Provider network can be switched to a selective PMSI (S-PMSI).

This has to be configured as a separate group and must contain a threshold which, if exceeded, will see a new PMSI signalled by the PE nearest the source, and traffic switched onto the S-PMSI.

```
*A:PE-1# configure service vprn 1
      mvpn
        provider-tunnel
          inclusive
            pim asm 239.255.255.1
          exit
        exit
      selective
        data-threshold 232.0.0.0/8 1
        pim-ssm 232.255.1.0/24
      exit
    exit
  exit
no shutdown
```

This shows that when the traffic threshold for multicast groups covered by the range 232.0.0.0/8 exceeds 1kbps between a pair of PEs, then an S-PMSI is signalled between the PEs. This is a separate multicast tunnel over which traffic in the given group now flows.

```
*A:PE-1# show router 1 pim s-psmi detail
=====
PIM Selective provider tunnels
=====
Md Source Address   : 192.0.2.1           Md Group Address   : 232.255.1.14
Number of VPN SGs   : 1                   Uptime             : 0d 00:00:12
MT IfIndex          : 16395

VPN Group Address    : 232.0.0.1           VPN Source Address : 192.168.1.2
State                : TX Joined           Mdt Threshold      : 1
Join Timer           : 0d 00:00:47         Holddown Timer     : 0d 00:00:47
=====
PIM Selective provider tunnels Interfaces : 1
=====
*A:PE-1#
```

In this example, the (S,G) group is (192.168.1.2, 232.0.0.1). As the data rate has exceeded the configured MDT threshold of 1kbps, a new provider tunnel with a group address of 232.255.1.14 has been signalled and now carries the multicast stream.

Data Path Using Selective PMSI

The TX Joined state indicates that the S-PMSI has been sourced at this PE — PE-1.

Comparing this to PE-3, where a receiver is connected through a CE indicates that it has received a join to connect the S-PMSI.

```
*A:PE-3# show router 1 pim s-pmsi detail
=====
PIM Selective provider tunnels
=====
Md Source Address   : 192.0.2.1           Md Group Address   : 232.255.1.14
Number of VPN SGs   : 1                   Uptime             : 0d 00:05:13
MT IfIndex          : 24576                Egress Fwding Rate : 52.8 kbps

VPN Group Address   : 232.0.0.1           VPN Source Address : 192.168.1.2
State               : RX Joined
Expiry Timer        : 0d 00:02:31
=====
PIM Selective provider tunnels Interfaces : 1
...
=====
*A:PE-3#
```


Conclusion

This note provides configuration on how to configure multicast within a VPRN with next generation multicast VPN techniques. Specifically, discovery of multicast VPNs using PIM and BGP auto-discovery mechanisms are described with a number of ASM and SSM signaling techniques within the customer domain.

Multicast in a VPN II

In This Chapter

This section provides information about multicast in a VPRN service.

Topics in this section include:

- [Applicability on page 1448](#)
- [Summary on page 1449](#)
- [Overview on page 1451](#)
- [Configuration on page 1453](#)
- [Conclusion on page 1502](#)

Applicability

This section is applicable to all 7750 SR platforms and to 7450 ESS platforms when configured in mixed-mode. It was tested on release 9.0R5. The features are supported with IOM2, IOM3-XP and IMMs, and need chassis mode C or higher. There are no other pre-requisites for this configuration.

Summary

Multicast VPN (MVPN) or Next Generation IP Multicast in an IP-VPN (NG-MVPNs) architectures describe a set of VRFs (Virtual Routing and Forwarding) or VPRNs (Virtual Private Routed Networks) that support the transport of multicast traffic across a provider network. MVPNs are defined in draft-ietf-l3vpn-2547bis-mcast-10.txt, *Multicast in MPLS/BGP IP VPNs*, and in draft-ietf-l3vpn-2547bis-mcast-bgp-08.txt, *BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPN*.

Initial MVPN deployments were originally based on Draft-Rosen (draft-rosen-vpn-mcast-08.txt) which described the protocols and procedures required to support an IP Multicast VPN. There are a number of limitations with the Draft-Rosen implementation including, but not limited to:

- Draft-Rosen requires a set of multicast distribution trees (MDTs) per VPN, which requires a PIM state per MDT. There is no option to aggregate MDT across multiple VPNs
- Customer signaling, initially PE discovery and Data MDT signaling, were all PIM-based as there was no mechanism available to decouple these. Now, PE discovery is supported using a BGP MDT Address Family Identifier/Subsequent Address Family Identifier (AFI/SAFI), however, the data MDT still needs PIM.
- There is no mechanism for using MPLS to encapsulate multicast traffic in the VPN. GRE is the only encapsulation method available in Draft-Rosen.
- Draft-Rosen multicast trees are signaled using PIM only. MVPN allows the use of mLDP and RSVP P2MP LSPs.
- PE to PE protocol exchanges for Draft-Rosen is achieved using PIM only. MVPN allows for the use of BGP signaling as per unicast Layer 3 VPNs.

NG-MVPN addresses these limitations by extending the idea of the per-VRF tree by introducing the idea of Provider Multicast Service Interfaces (PMSIs). These are equivalent to the default MDTs of Draft-Rosen. NG-MVPN allows the decoupling of the mechanisms required to create a multicast VPN, such as PE auto-discovery (which PEs are members of which VPN), PMSI signaling (creation of tunnels between PEs) and customer multicast signaling (multicast signaling —IGMP/PIM— received from customer edge routers). Two types of PMSI exist:

- Inclusive (I-PMSI) — Contains all the PEs for a given MVPN, I-PMSI is the default multicast data path between all PEs of the same VPN.
- Selective (S-PMSI) — Contains only a subset of PEs of a given MVPN, used to optimize multicast stream distribution to only the PEs with active receivers for those streams.

The [Multicast in a VPN](#) section on [page 1389](#) contains the VPN configuration required for the provider multicast domain using PIM Any Source Multicast (ASM) with auto-discovery based on PIM or BGP auto-discovery (A/D), PIM used for the customer multicast signaling and PIM Source Specific Multicast (SSM) used for the S-PMSI creation. The customer domain configuration covers the following cases:

Summary

- PIM ASM with the Rendezvous Point (RP) in the provider PE
- PIM ASM using anycast RP on the provider RPs
- PIM SSM

This section introduces some of the features that were not supported at the time of writing of the Multicast in a VPN I (Release 7.0). It provides configuration details to implement:

- Multicast LDP (mLDP) and RSVP-TE Point to Multipoint (P2MP) for building customer trees (C-trees) which are using MPLS instead of PIM techniques.
- MVPN source redundancy.
- MDT AFI/SAFI (to fully interoperate with Cisco networks).

Note that PIM SSM is the only case addressed in this example, other PIM customer domain configurations are out of the scope, for more information refer to [Multicast in a VPN I on page 1389](#)

Overview

The network topology is displayed in [Figure 207](#). The setup consists of four 7750 SRs acting as Provider Edge (PE) routers within a single Autonomous System (AS).

- Full mesh ISIS in the AS (OSPF could be used instead)
- LDP on all interfaces in each AS (RSVP could be used instead)
- MP-iBGP sessions between the PE routers in the AS (Route Reflectors (RRs) could also be used).
- Layer 3 VPN on all PEs with identical route targets, in the form AS-number: vpnr-service-id

Connected to each PE is a single 7750 SR acting as a Customer Edge (CE) router. CE-1 has a multicast source connected, and PEs 2 to 4 each have a single receiver connected which will receive the multicast streams from the source. In this setup, each receiver is IGMPv3 capable. If the receiver is IGMPv3 capable, it will issue IGMPv3 reports that may include a list of required source addresses.

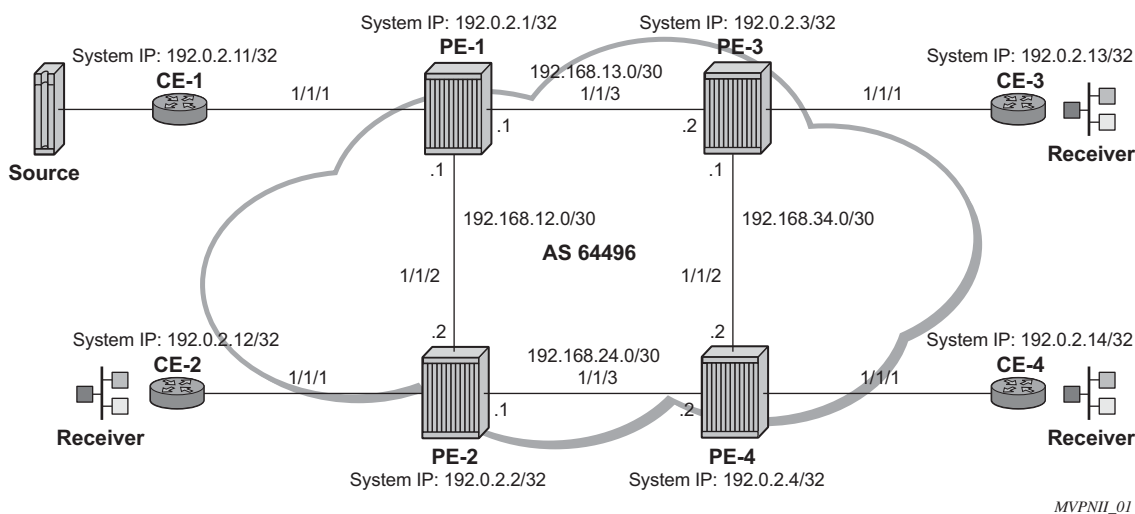


Figure 207: Network Topology

When the receiver wishes to become a member of any group, the source address of the group must be known to the CE. As a result the source address must be IP reachable by each CE, so it is advertised by CE-1 to the PEs with attachment circuits in VPRN using BGP. Static routes are then configured on the receiver CEs to achieve IP reachability to the source address of the multicast group.

Multicast traffic from the source is streamed towards router CE-1. Receivers connected to PE-2, PE-3 and PE-4 are interested in joining this multicast group.

CEs 1 to 4 are PIM enabled routers, which form a PIM adjacency with nearest PE. Between the PEs across the provider network there are no PIM adjacencies since BGP auto-discovery and BGP signalling are used. Selective PMSI using mLDP or RSVP P2MP are out of the scope of this section. Selective PMSI using PIM SSM is supported but cannot be used when I-PMSI is mLDP or RSVP with R9.0. I-PMSI and S-PMSI must use same tunnelling technology, either PIM/GRE or mLDP or RSVP P2MP.

Configuration

The configuration is divided into the following sections:

- Provider Common Configuration
 - PE Global Configuration
- PE VPRN Configuration and PE VPRN Multicast Configuration for NG-MVPN
 - PMSI using mLDP
 - PMSI using RSVP-TE
 - UMH (Upstream Multicast Hop)
- PE VPRN Configuration and PE VPRN Multicast Configuration for Draft-Rosen using MDT AFI SAFI
 - Auto discovery using BGP MDT AFI SAFI as per Draft-Rosen version 9 with MDT using PIM SSM

Provider Common Configuration

PE Global Configuration

This section describes the common configuration required for each PE within the provider multicast domain, regardless of the MVPN PE auto-discovery or customer signaling methods. This includes Interior Gateway Protocol (IGP) and VPRN service configuration.

The configuration tasks can be summarized as follows:

- PE global configuration.

This includes configuration of the IGP (ISIS will be used); configuration of link layer LDP between PEs (LDP will be used here as the method to interconnect VPRNs); configuration of iBGP between PEs to facilitate VPRN route learning.
- VPRN configuration on the PEs.

This includes configuration of basic VPRN parameters (route-distinguisher, route target communities), configuration of attachment circuits towards CEs, configuration of VRF routing protocol and any routing policies.
- PIM within the VRF and MVPN parameters — I-PMSI
- CE configuration.

Step 1. Configure the interfaces, the IGP (ISIS) in all PE nodes (where ISIS redistributes route reachability to all routers) and LDP in the interfaces (link layer LDP). To facilitate the ISIS configuration, all routers are Level2-Level1 capable within the same ISIS area-id, so there is only a single topology area in the network (all routers share the same topology). The configuration for PE-2 is displayed below.

```
PE-2>configure router
  interface "int-PE-2-PE-1"
    address 192.168.12.2/30
    port 1/1/2:1
  exit
  interface "int-PE-2-PE-4"
    address 192.168.24.1/30
    port 1/1/3:1
  exit
  interface "system"
    address 192.0.2.2/32
  exit
  autonomous-system 64496
  isis
    area-id 49.0001
    traffic-engineering
    interface "system"
      passive
    exit
    interface "int-PE-2-PE-1"
      interface-type point-to-point
    exit
    interface "int-PE-2-PE-4"
      interface-type point-to-point
    exit
    no shutdown
  exit
  ldp
    interface-parameters
      interface "int-PE-2-PE-1"
        exit
      interface "int-PE-2-PE-4"
        exit
    exit
    targeted-session
  exit
```

The configuration for the rest of nodes is similar. The IP addresses can be derived from [Figure 207](#).

Step 2. Verify that ISIS adjacencies and LDP peer sessions are formed.

```
*A:PE-1# show router isis adjacency
=====
ISIS Adjacency
=====
System ID                Usage State Hold Interface                MT Enab
-----
PE-2                     L1L2  Up    30   int-PE-1-PE-2                No
PE-3                     L1L2  Up    24   int-PE-1-PE-3                No
-----
Adjacencies : 2
=====
*A:PE-1# show router ldp session
=====
LDP Sessions
=====
Peer LDP Id              Adj Type   State           Msg Sent  Msg Recv  Up Time
-----
192.0.2.2:0              Both       Established     235       237       0d 00:08:57
192.0.2.3:0              Both       Established     237       245       0d 00:08:58
-----
No. of Sessions: 2
```

Step 3. Configure iBGP full mesh between the PEs for VPRN routing (Route Reflectors could also be an option).

```
*A:PE-1>config>router>bgp# info
-----
min-route-advertisement 1
rapid-withdrawal
rapid-update mvpn-ipv4 mdt-safi
group "internal"
  family vpn-ipv4 mvpn-ipv4 mdt-safi
  type internal
  neighbor 192.0.2.2
  exit
  neighbor 192.0.2.3
  exit
  neighbor 192.0.2.4
  exit
exit
no shutdown
-----
```

Note that the families configured under the group **internal** are vpn-ipv4, mvpn-ipv4 and mdt-safi, since the three families are referenced in this chapter.

Note that the mdt-safi parameter is not needed for NG-MVPN (mLDP/RSVP scenarios) and is only required for Draft-Rosen with MDT AFI SAFI.

Rapid withdrawal (configured on all PEs) disables the Minimum Route Advertisement Interval (MRAI) interval on sending BGP withdrawals. Rapid update (configured for MVPN-IPv4 and MDT AFI/SAFI address families) disables the MRAI interval on sending BGP update messages for the address family MVPN-IPv4 and MDT AFI/SAFI).

Step 4. Verify that BGP peer relationships are established.

```
*A:PE-1# show router bgp summary
=====
BGP Router ID:192.0.2.1      AS:64496      Local AS:64496
=====
BGP Admin State      : Up      BGP Oper State      : Up
Total Peer Groups    : 1      Total Peers          : 3
* Truncated info
=====
BGP Summary
=====
Neighbor
      AS PktRcvd InQ  Up/Down  State|Rcv/Act/Sent (Addr Family)
      PktSent OutQ
-----
192.0.2.2
      64496      28    0 00h12m04s 4/4/3 (VpnIPv4)
      29      0      1/1/1 (MvpnIpv4)
      0/0/0 (MdtSafi)
192.0.2.3
      64496      27    0 00h12m11s 2/2/3 (VpnIPv4)
```

		30	0		1/1/1 (MvpnIpv4)
					0/0/0 (MdtSafi)
192.0.2.4					
	64496	30	0	00h12m04s	2/2/3 (VpnIPv4)
		29	0		2/2/1 (MvpnIpv4)
					0/0/0 (MdtSafi)

PE VPRN Configuration and PE VPRN Multicast Configuration

A VPRN is created on each PE per service (the different services using mLDP, RSVP-TE and AFI/SAFI with PIM), these are the multicast VPRNs. PE-1 is the PE containing the attachment circuit towards CE-1. CE-1 is the CE nearest to the source. PE-2, PE-3 and PE-4 contain attachment circuits towards CE-2, CE-3 and CE-4 respectively. Each CE has a receiving host attached.

PMSI using mLDP

Figure 208 shows the details of the topology for VPRN 1.

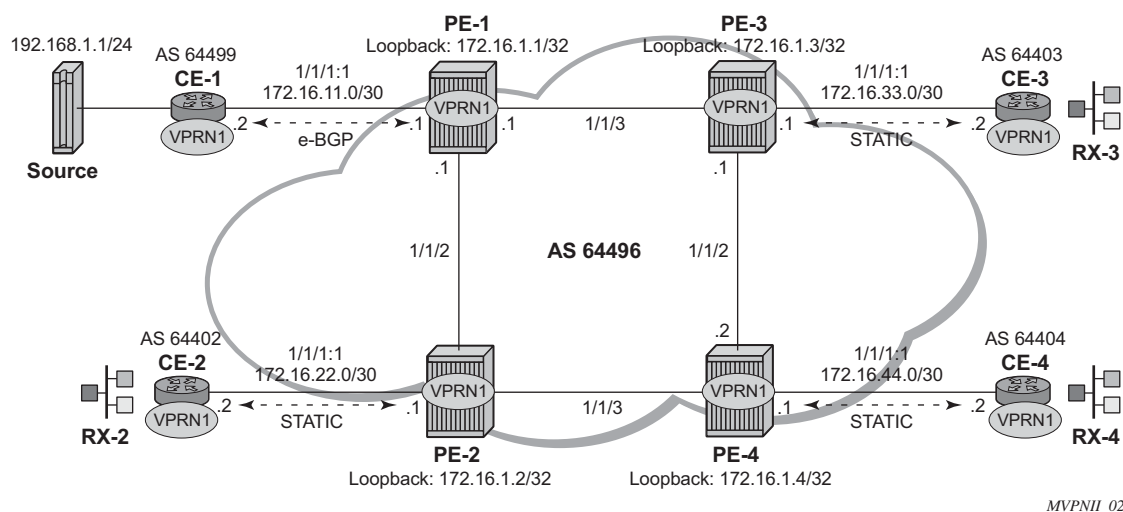


Figure 208: VPRN 1 Topology used for mLDP

Unicast

Step 1. Create VPRN 1 on each PE, containing a route-distinguisher of 64496:10X (where X= number of PE) and vrf-target of 64496:100. The autonomous system number is 64496. For the next hop tunnel route resolution to connect the VPRNs between PEs, manually configured spoke SDPs are created (note that other methods like auto-bind LDP, RSVP-TE or auto-bind MPLS could also be used). LDP was already enabled in the overview section.

```
*A:PE-1>config>service# info
-----
customer 1 create
    description "Default customer"
exit
sdp 12 mpls create
    far-end 192.0.2.2
    ldp
    keep-alive
    shutdown
exit
no shutdown
exit
sdp 13 mpls create
    far-end 192.0.2.3
    ldp
    keep-alive
    shutdown
exit
no shutdown
exit
sdp 14 mpls create
    far-end 192.0.2.4
    ldp
    keep-alive
    shutdown
exit
no shutdown
exit
vprn 1 customer 1 create
    description "mLDP"
    autonomous-system 64496
    route-distinguisher 64496:101
    vrf-target target:64496:100

    spoke-sdp 12 create
        no shutdown
    exit
    spoke-sdp 13 create
        no shutdown
    exit
    spoke-sdp 14 create
        no shutdown
    exit
```

Step 2. Create an attachment circuit interface towards the CE and a loopback (the loopback is not mandatory but it is configured to aid troubleshooting the routers).

```
PE-1>configure service vprn 1
    interface "loopback" create
        address 172.16.1.1/32
        loopback
    exit
    interface "int-PE-1-CE-1" create
        address 172.16.11.1/30
        sap 1/1/1:1 create
    exit
exit
```

Step 3. The source address of the multicast stream will need to be reachable by all routers (PEs and CEs) within the VPN. This will be advertised within BGP from CE-1 to PE-1. Create a BGP peering relationship with the CE.

```
PE-1>configure service vprn 1
    bgp
        group "external"
            type external
            peer-as 64499
            neighbor 172.16.11.2
        exit
    exit
    no shutdown
exit
```

Step 4. On CE-1, create a VPRN to support the connection of the source to CE-1 and to connect CE-1 to PE-1. Two attachment circuits are required as well as a BGP peering relationship with the PE. This uses a default BGP address family of ipv4.

```
A:CE-1>config>service>vprn# info
-----
autonomous-system 64499
route-distinguisher 64499:1
interface "int-CE-1-PE-1" create
    *Truncated info
interface "source" create
    *Truncated info
exit

bgp
    export "source"
    group "external"
        type external
        peer-as 64496
        neighbor 172.16.11.1
    exit
exit
no shutdown
exit
no shutdown
-----
```


Step 5. In order for the subnet on the CE connecting to the source to be advertised within BGP, a route policy is required. The subnet containing the multicast source is 192.168.1.0/24, so a prefix-list can be used to define a match, and then used within a route policy to inject into BGP. In this example, only the host (/32) of the source is advertised.

```
A:CE-1>config>router# info
* Truncated info
  policy-options
  begin
  prefix-list "source"
    prefix 192.168.1.1/32 exact
  exit
  policy-statement "source"
    entry 10
      from
        prefix-list "source"
      exit
      to
        protocol bgp
      exit
      action accept
      exit
    exit
  exit
  commit
exit
-----
```

Step 6. Check that the route is seen in PE-1:

```
*A:PE-1>config>service# show router 1 route-table
=====
Route Table (Service: 1)
=====
Dest Prefix[Flags]                                Type   Proto   Age           Pref
Next Hop[Interface Name]                        Metric
-----
* Truncated info
192.168.1.1/32                                     Remote BGP       00h04m15s    170
    172.16.11.2                                     0
-----
No. of Routes: 10
Flags: L = LFA nexthop available    B = BGP backup route available
      n = Number of times nexthop is repeated
=====
```

This prefix will also be automatically advertised within the BGP VPRN to all other PEs, and will be installed in VRF 1.

For example, on PE-4, the source 192.168.1.1/32 is received via BGP VPN with a next-hop of PE-1 (192.0.2.1):

```
*A:PE-4>config>service>vprn# show router 1 route-table
=====
```

Provider Common Configuration

```
Route Table (Service: 1)
=====
Dest Prefix[Flags]                Type   Proto   Age      Pref
      Next Hop[Interface Name]                Metric
-----
* Truncated info
192.168.1.1/32                    Remote BGP VPN 00h00m05s 170
      192.0.2.1 (tunneled)                        0
-----
No. of Routes: 10
Flags: L = LFA nexthop available    B = BGP backup route available
      n = Number of times nexthop is repeated
=====
```

Each CE containing a multicast receiver must be able to reach the source. As an example on CE-2, a static route will suffice and is configured with next hop of the PE-2 PE-CE interface.

```
A:CE-2# configure service vprn 1
...
static-route 192.168.1.1/32 next-hop 172.16.22.1
```

After **Steps 1 to 6**, all required unicast routing is provisioned. The following sections show the configuration of the multicast in the VPRN.

Auto-Discovery and mLDP PMSI Establishment

The MP-BGP based auto-discovery is implemented with a new address family defined in RFC 4760 MP_REACH_NRLI/MP_UNREACH_NRLI attributes, with AFI 1 (IPv4) or 2 (IPv6) SAFI 5 (temporary value assigned by IANA). This is the mechanism by which each PE advertises the presence of an MVPN to other PEs. This can be achieved using PIM (like in Draft-Rosen) or using BGP. With the default parameter, BGP is automatically chosen because the PMSIs are mLDP and PIM is not an option in this case. Any PE that is a member of an MVPN will advertise to the other PEs using a BGP Multi-protocol Reachable Next-Hop Router Layer Information (NRLI) update that is sent to all PEs within the AS. This update will contain an Intra-AS I-PMSI auto-discovery Route type, also known as an Intra-AD. These use an address family mvpn-ipv4, so each PE must be configured to originate and accept such updates (note this was done earlier when configuring the families).

At this step (auto-discovery), the information about the PMSI is exchanged but the PMSI is not instantiated.

As each PE contains a CE which will be part of the multicast VRF, it is necessary to enable PIM on each interface containing the attachment circuit towards a CE, and to configure the I-PMSI multicast tunnel for the VRF. Note that S-PMSIs are not supported for mLDP with the 9.0R5 software release. In order for the BGP routes to be accepted into the VRF, a route-target community is required (vrf-target). This is configured in the **configure service vprn 1 mvpn** context and, in this case is set to the same value as the unicast vrf-target (the vrf-target community as the **configure service vprn 1 vrf-target** context).

On each PE, a VPRN instance is configured as follows:

```
A:PE-4# configure service vprn 1
...?
pim
    interface "loopback"
    exit
    interface "int-PE-4-CE-4"
    exit
    rp
        static
        exit
        bsr-candidate
            shutdown
        exit
        rp-candidate
            shutdown
        exit
    exit
    no shutdown
exit
mvpn
    auto-discovery default
    c-mcast-signaling bgp
    exit
    provider-tunnel
        inclusive
```

Provider Common Configuration

```
        mldp
        no shutdown
    exit
    exit
    exit
    vrf-target unicast
    exit
exit
```

When PIM SSM is used, the configuration always shows RP static with no RP entries (this is enabled by default when PIM is provisioned). In order for the BGP routes to be accepted into the VRF, a route-target community is required (vrf-target). Although it is not mandatory for the mvpn target to be equal to the unicast target, the recommendation is to use **vrf-target unicast** to avoid configuration mistakes and extra complexity.

The status of VPRN 1 on PE-1 is shown with the following output:

```
*A:PE-1# show router 1 mvpn
=====
MVPN 1 configuration data
=====
signaling           : Bgp           auto-discovery      : Default
UMH Selection       : Highest-Ip    intersite-shared     : Enabled
vrf-import          : N/A
vrf-export          : N/A
vrf-target          : unicast
C-Mcast Import RT   : target:192.0.2.1:2

ipmsi               : ldp
i-pmsi P2MP AdmSt   : Up

s-pmsi              : none
data-delay-interval: 3 seconds
enable-asm-mdt      : N/A
=====
```

The following shows a debug of an Intra-AD BGP update message received by PE-1 that was sent by PE-2. The message contains the PMSI tunnel type to be used (LDP P2MP LSP), LSP identification (root node, opaque value) and the type of BGP update (Type: Intra-AD Len: 12 RD: 64496:102 Orig: 192.0.2.2):

```
A 4 2011/10/06 01:25:42.81 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.2
"Peer 1: 192.0.2.2: UPDATE
Peer 1: 192.0.2.2 - Received BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 91
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 8 Len: 4 Community:
        no-export
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:64496:100
```

```

Flag: 0xc0 Type: 22 Len: 22 PMSI:
  Tunnel-type LDP P2MP LSP (2)
  Flags [Leaf not required]
  MPLS Label 0
  Root-Node 192.0.2.2, LSP-ID 0x2001
Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
  Address Family MVPN_IPV4
  NextHop len 4 NextHop 192.0.2.2
  Type: Intra-AD Len: 12 RD: 64496:102 Orig: 192.0.2.2
"

```

The set up has four PEs, so every PE should see each others peer Intra-AD route; the output below shows the routes received in PE-1:

```

*A:PE-1# show router bgp routes mvpn-ipv4 type intra-ad
=====
BGP Router ID:192.0.2.1      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP MVPN-IPv4 Routes
=====
Flag RouteType      OriginatorIP      LocalPref  MED
      RD            SourceAS          VPNLabel
      Nexthop       SourceIP
      As-Path       GroupIP
-----
u*>i Intra-Ad         192.0.2.2         100        0
      64496:102      -                 -
      192.0.2.2      -                 -
      No As-Path     -                 -
u*>i Intra-Ad         192.0.2.3         100        0
      64496:103      -                 -
      192.0.2.3      -                 -
      No As-Path     -                 -
u*>i Intra-Ad         192.0.2.4         100        0
      64496:104      -                 -
      192.0.2.4      -                 -
      No As-Path     -                 -
-----
Routes : 3
=====

```

The detailed output of the Intra-AD received from PE-2 shows the Tunnel-Type LDP P2MP LSP (LSP-ID is 8193), the originator id (192.0.2.2), and the route-distinguisher (64496:102):

```

*A:PE-1# show router bgp routes mvpn-ipv4 type intra-ad detail
=====
BGP Router ID:192.0.2.1      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP MVPN-IPv4 Routes

```

Provider Common Configuration

```
=====
Route Type      : Intra-Ad
Route Dist.     : 64496:102
Originator IP   : 192.0.2.2
Nextthop        : 192.0.2.2
From            : 192.0.2.2
Res. Nextthop   : 0.0.0.0
Local Pref.     : 100
Aggregator AS   : None
Atomic Aggr.    : Not Atomic
Community       : no-export target:64496:100
Cluster         : No Cluster Members
Originator Id   : None
Flags           : Used Valid Best IGP
Route Source    : Internal
AS-Path         : No As-Path
VPRN Imported   : 1
Interface Name  : NotAvailable
Aggregator      : None
MED             : 0
Peer Router Id  : 192.0.2.2
-----
PMSI Tunnel Attribute :
Tunnel-type       : LDP P2MP LSP
MPLS Label        : 0
Root-Node         : 192.0.2.2
Flags             : Leaf not required
LSP-ID            : 8193
-----
```

Because of the receiver-driven nature of mLDP, mLDP P2MP LSPs are setup unsolicited from the leaf PEs towards the head-end PE. The leaf PEs discover the head-end PE via I-PMSI/S-PMSI A-D routes. The Tunnel Identifier carried in the PMSI attribute is used as the P2MP Forwarding Equivalence Class (FEC) Element. The Tunnel Identifier consists of the head-end PE's address, along with a Generic LSP Identifier value. The Generic LSP Identifier value is automatically generated by the head-end PE. The previous show command displays the PMSI information with the detail of the Root-Node (192.0.2.2) and the LSP-ID (8193). The PMSI was created after receiving the A-D message from PE-2, where the following excerpt from the previous debug shows the same information (x2001 in HEX is equal to 8193 in decimal).

```
Flag: 0xc0 Type: 22 Len: 22 PMSI:
  Tunnel-type LDP P2MP LSP (2)
  Flags [Leaf not required]
  MPLS Label 0
  Root-Node 192.0.2.2, LSP-ID 0x2001
```

Once the mLDP P2MP LSPs are created, the I-PMSI is instantiated in the core:

```
*A:PE-1# show router 1 pim neighbor
=====
PIM Neighbor ipv4
=====
Interface      Nbr DR Prty    Up Time      Expiry Time  Hold Time
  Nbr Address
-----
int-PE-1-CE-1  1              0d 01:21:30  0d 00:01:16  105
  172.16.11.2
mpls-if-73729  1              0d 01:21:03  never         65535
  192.0.2.2
mpls-if-73730  1              0d 01:20:47  never         65535
```

```

192.0.2.3
mpls-if-73731          1          0d 01:20:32  never          65535
192.0.2.4
-----
Neighbors : 4
=====

*A:PE-1# show router 1 pim tunnel-interface
=====
PIM Interfaces ipv4
=====
Interface              Adm  Opr  DR Prty          Hello Intvl  Mcast Send
DR
-----
mpls-if-73728          Up   Up   N/A           N/A          N/A
192.0.2.1
mpls-if-73729          Up   Up   N/A           N/A          N/A
192.0.2.2
mpls-if-73730          Up   Up   N/A           N/A          N/A
192.0.2.3
mpls-if-73731          Up   Up   N/A           N/A          N/A
192.0.2.4
-----
Interfaces : 4
=====

```

Every PE has created an I-PMSI to the other PEs. Checking the mLDP P2MP LSPs that are originated, transit, or destination to PE-1:

```

*A:PE-1# show router ldp bindings fec-type p2mp active p2mp-id 8193 root 192.0.2.1
=====
LDP P2MP Bindings (Active)
=====
P2MP-Id      RootAddr
Interface    Op          IngLbl      EgrLbl  EgrIntf/    EgrNextHop
              Op          LspId
-----
8193          192.0.2.1
73728          Push          --          262137  1/1/2:1      192.168.12.2
8193          192.0.2.1
73728          Push          --          262137  1/1/3:1      192.168.13.2
8193          192.0.2.2
73729          Pop           262137      --      --           --
8193          192.0.2.2
73729          Swap          262137      262136  1/1/3:1      192.168.13.2
8193          192.0.2.3
73730          Swap          262142      262136  1/1/2:1      192.168.12.2
8193          192.0.2.3
73730          Pop           262142      --      --           --
8193          192.0.2.4
73731          Pop           262136      --      --           --
-----
No. of P2MP Active Bindings: 7
=====

```

The two first entries in the output show the P2MP LSP where PE-1 is the root headend (Push). The other two entries (Swap and Pop) correspond with transit and leaf for the P2MP LSPs originated

by the other PEs. The command shows a P2MP-ID (8193) with an interface 73728 (matches with the **show router 1 pim tunnel interface** being the PIM interface created from PE-1) with two egress interfaces pointing to PE-2 and PE-3.

A similar command executed on PE-2 shows:

```
*A:PE-2# show router ldp bindings fec-type p2mp p2mp-id 8193 root 192.0.2.1 active
=====
LDP P2MP Bindings (Active)
=====
P2MP-Id      RootAddr
Interface    Op          IngLbl      EgrLbl      EgrIntf/    EgrNextHop
                               LspId
-----
8193          192.0.2.1
73729        Pop          262137      --          --          --
8193          192.0.2.1
73729        Swap         262137      262135      1/1/3:1     192.168.24.2
* Truncated info
-----
No. of P2MP Active Bindings: 7
=====
```

On PE-2, the first entry shows that PE-2 is a leaf of the P2MP LSP tree created by PE-1 (ingress label is 262137 which was the egress label to reach PE-2 and is popped). However, the second entry shows that PE-2 is transit for the P2MP LSP going to PE-4 (ingress label 262137, egress label 262135 next hop PE-4).

The same command on PE-4 shows:

```
*A:PE-4# show router ldp bindings active fec-type p2mp p2mp-id 8193 root 192.0.2.1
=====
LDP P2MP Bindings (Active)
=====
P2MP-Id      RootAddr
Interface    Op          IngLbl      EgrLbl      EgrIntf/    EgrNextHop
                               LspId
-----
8193          192.0.2.1
73731        Pop          262135      --          --          --
* Truncated info
-----
No. of P2MP Active Bindings: 5
=====
```

In the first entry the root is PE-1 and the action is Pop, being the ingress label 262135, showing that this is another leaf for the P2MP LSP started on PE-1.

To complete the information, checking on PE-3, the first entry there is a Pop where the root is PE-1, and the ingress label is 262137:

```
*A:PE-3# show router ldp bindings active fec-type p2mp p2mp-id 8193 root 192.0.2.1
```



```

=====
LDP P2MP Bindings (Active)
=====
P2MP-Id      RootAddr
Interface     Op          IngLbl    EgrLbl  EgrIntf/  EgrNextHop
                               LspId
-----
8193          192.0.2.1
73729         Pop          262137    --      --        --
* Truncated info
-----
No. of P2MP Active Bindings: 5
=====

```

As a summary, each root PE has a P2MP LSP with three leaves (the other PEs) and they are also transit points to the P2MP LSPs created in the other PEs. As an additional check, an OAM ping can show the different leaves that a P2MP LSP has:

```

*A:PE-1# oam p2mp-lsp-ping ldp 8193 sender-addr 192.0.2.1 detail
P2MP identifier 8193: 88 bytes MPLS payload
=====
Leaf Information
=====
From          RTT          Return Code
-----
192.0.2.2     =3.03ms      EgressRtr(3)
192.0.2.4     =5.17ms      EgressRtr(3)
192.0.2.3     =33.1ms      EgressRtr(3)
=====
Total Leafs responded = 3
      round-trip min/avg/max   = 3.03 / 13.8 / 33.1 ms

Responses based on return code:
      EgressRtr(3)=3

```

An easy way to see the path that the LDP P2MP LSP follows for a specific leaf is the following command (such as LDP trace from PE-1 to PE-4):

```

*A:PE-1# oam ldp-treetrace prefix 192.0.2.4/32

ldp-treetrace for Prefix 192.0.2.4/32:

      192.168.24.2, ttl = 2 dst = 127.1.0.255 rc = EgressRtr status = Done
Hops:      192.168.12.2

ldp-treetrace discovery state: Done
ldp-treetrace discovery status: ' OK '
Total number of discovered paths: 1
Total number of failed traces: 0

```

The command shows that on PE-4 there is an active leaf of the P2MP LSP, and that there is an intermediate hop on PE-2.

Traffic Flow

The receiver RX-4, connected to CE-4, wishes to join the group 232.1.1.1 with source 192.168.1.1 and so sends an IGMPv3 report towards CE-4. CE-4 recognizes the report and sends a PIM join towards the source, hence it reaches PE-1 where the source is connected to through CE-1. The output below shows the debug seen on PE-4, where the PIM join is received from CE-4 and a BGP update Source Join is sent to all PEs (note that only the update sent to PE-1 is shown).

```
11 2011/10/13 15:41:30.83 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimJPProcessSG
pimJPProcessSG: (S,G)-> (192.168.1.1,232.1.1.1) type <S,G>, i/f int-PE-4-CE-4, u
pNbr 172.16.44.1 isJoin 1 isRpt 0 holdTime 210"

12 2011/10/13 15:41:30.83 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimJPPrintFsmEvent
PIM JP Downstream: State NoInfo Event RxJoin, (S,G) (192.168.1.1,232.1.1.1) grou
pType <S,G>"

13 2011/10/13 15:41:30.83 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimJPPrintFsmEvent
PIM JP Upstream: State NotJoined Event JoinDesiredTrue, (S,G) (192.168.1.1,232.1
.1.1) groupType <S,G>"

14 2011/10/13 15:41:30.83 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimSGUpJoinDesiredTrue
No upstream interface. pSG (192.168.1.1,232.1.1.1) rpfType 3"

15 2011/10/13 15:41:30.83 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimSGUpJoinDesiredTrue
pim 2 sg_type 2 refetch route type SPMSI pendingFetchMask 0x8"

16 2011/10/13 15:41:30.83 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimSGUpJoinDesiredTrue
No upstream interface. pSG 0x5578f87c, (192.168.1.1,232.1.1.1) rpfType 3"

17 2011/10/13 15:41:30.83 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimJPPrintFsmEvent
PIM JP Upstream: State Joined Event MribChange, (S,G) (192.168.1.1,232.1.1.1) gr
oupType <S,G>"

18 2011/10/13 15:41:30.83 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimSGUpStateJMribChange
pSG 0x5578f87c, (192.168.1.1,232.1.1.1), type <S,G> oldMribNhopIp 0.0.0.0 oldRpf
NbrIp 0.0.0.0, oldRpfType NONE oldRpfif 0 rptMribNhopIp 0.0.0.0, rptRpfNbrIp 0.0
.0.0 rtmReason 32"

19 2011/10/13 15:41:30.83 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimSGUpStateJMribChange
pSG 0x5578f87c, (192.168.1.1,232.1.1.1), type <S,G> newMribNhopIp 192.0.2.1 newR
pNbrIp 192.0.2.1 newRpfType REMOTE newRpfif 73731"

20 2011/10/13 15:41:30.83 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.1
"Peer 1: 192.0.2.1: UPDATE
Peer 1: 192.0.2.1 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 69
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
```

```

Flag: 0x40 Type: 2 Len: 0 AS Path:
Flag: 0x80 Type: 4 Len: 4 MED: 0
Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:192.0.2.1:2
Flag: 0x90 Type: 14 Len: 33 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.4
    Type: Source-Join Len:22 RD: 64496:101 SrcAS: 64496 Src: 192.168.1.1 Grp
: 232.1.1.1
"

```

The following debug shows that PE-1 receives the BGP update Source Join with source 192.168.1.1 and group 232.1.1.1 and sends a PIM join towards CE-1:

```

57 2011/10/14 02:06:58.22 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.4
"Peer 1: 192.0.2.4: UPDATE
Peer 1: 192.0.2.4 - Received BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 69
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:192.0.2.1:2
    Flag: 0x90 Type: 14 Len: 33 Multiprotocol Reachable NLRI:
        Address Family MVPN_IPV4
        NextHop len 4 NextHop 192.0.2.4
        Type: Source-Join Len:22 RD: 64496:101 SrcAS: 64496 Src: 192.168.1.1 Grp
: 232.1.1.1
"

58 2011/10/14 02:06:58.22 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimJPPProcessSG
pimJPPProcessSG: (S,G)-> (192.168.1.1,232.1.1.1) type <S,G>, i/f mpls-if-73728, u
pNbr 192.0.2.1 isJoin 1 isRpt 0 holdTime 65535"

59 2011/10/14 02:06:58.22 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimJPPrintFsmEvent
PIM JP Downstream: State NoInfo Event RxJoin, (S,G) (192.168.1.1,232.1.1.1) grou
pType <S,G>"

60 2011/10/14 02:06:58.22 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimJPPrintFsmEvent
PIM JP Upstream: State NotJoined Event JoinDesiredTrue, (S,G) (192.168.1.1,232.1
.1.1) groupType <S,G>"

61 2011/10/14 02:06:58.22 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimSGUpJoinDesiredTrue
pim 2 sg_type 2 refetch route type SPMSI pendingFetchMask 0x8"

62 2011/10/14 02:06:58.22 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimSendJoinPrunePdu
pimSendJoinPrunePdu: if 3, adj 172.16.11.2"

63 2011/10/14 02:06:58.22 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]

```

Provider Common Configuration

```
"PIM[Instance 2 vprn1]: pimSGEncodeGroupSet
pimEncodeGroupSet: encoding groupset for group 232.1.1.1, numJoinedSrcs 1, numPrunedSrcs 0"
```

```
64 2011/10/14 02:06:58.22 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
```

```
"PIM[Instance 2 vprn1]: pimSGEncodeGroupSet
pimEncodeGroupSet: Encoding Join for source 192.168.1.1"
```

```
65 2011/10/14 02:06:58.22 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
```

```
"PIM[Instance 2 vprn1]: pimSGEncodeGroupSet
pimEncodeGroupSet: num joined srcs 1, num pruned srcs 0"
```

```
66 2011/10/14 02:06:58.22 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
```

```
"PIM[Instance 2 vprn1]: pimSendJoinPrunePdu
pimSendJoinPrunePdu2: sending JP PDU with 1 groups."
```

The BGP update source join received by PE-1 is displayed with the command:

```
*A:PE-1# show router bgp routes mvpn-ipv4 type source-join
=====
BGP Router ID:192.0.2.1          AS:64496          Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP MVPN-IPv4 Routes
=====
Flag RouteType OriginatorIP LocalPref MED
RD SourceAS VPNLabel
Nexthop SourceIP
As-Path GroupIP
-----
u*>i Source-Join - 100 0
64496:101 64496 -
192.0.2.4 192.168.1.1
No As-Path 232.1.1.1
-----
Routes : 1
=====
```

To verify the traffic: on PE-1 there is a group 232.1.1.1 with source 192.168.1.1, the Reverse Path Forwarding (RPF) is CE-1, the multicast traffic is flowing from CE-1 to PE-1 using int-PE-1-CE-1 and the outgoing interface is using the PMSI mLDP mpls-if-73728.

```
*A:PE-1# show router 1 pim group detail
=====
PIM Source Group ipv4
=====
Group Address      : 232.1.1.1
Source Address     : 192.168.1.1
RP Address         : 0
Flags              :
MRIB Next Hop      : 172.16.11.2
MRIB Src Flags     : remote
Up Time            : 0d 00:12:04
Type               : (S,G)
Keepalive Timer    : Not Running
Resolved By        : rtable-u
```

```

Up JP State      : Joined          Up JP Expiry      : 0d 00:00:56
Up JP Rpt        : Not Joined StarG Up JP Rpt Override : 0d 00:00:00

```

```

Register State   : No Info
Reg From Anycast RP: No

```

```

Rpf Neighbor     : 172.16.11.2
Incoming Intf    : int-PE-1-CE-1
Outgoing Intf List : mpls-if-73728

```

```

Curr Fwding Rate : 0.0 kbps
Forwarded Packets : 0
Forwarded Octets  : 0
Spt threshold     : 0 kbps
Admin bandwidth   : 1 kbps
Discarded Packets : 0
RPF Mismatches    : 0
ECMP opt threshold : 7

```

```

-----
Groups : 1
=====

```

On PE-4, the same (S,G) arrives in the incoming interface mpls-if-73731 and the outgoing interface is int-PE-4-CE-4.

```
*A:PE-4# show router 1 pim group detail
```

```
=====
PIM Source Group ipv4
=====
```

```

Group Address      : 232.1.1.1
Source Address     : 192.168.1.1
RP Address         : 0
Flags              :
Type               : (S,G)
MRIB Next Hop     : 192.0.2.1
MRIB Src Flags     : remote
Up Time           : 0d 00:15:44
Keepalive Timer    : Not Running
Resolved By       : rtable-u

```

```

Up JP State      : Joined          Up JP Expiry      : 0d 00:00:16
Up JP Rpt        : Not Joined StarG Up JP Rpt Override : 0d 00:00:00

```

```

Register State   : No Info
Reg From Anycast RP: No

```

```

Rpf Neighbor     : 192.0.2.1
Incoming Intf    : mpls-if-73731
Outgoing Intf List : int-PE-4-CE-4

```

```

Curr Fwding Rate : 0.0 kbps
Forwarded Packets : 0
Forwarded Octets  : 0
Spt threshold     : 0 kbps
Admin bandwidth   : 1 kbps
Discarded Packets : 0
RPF Mismatches    : 0
ECMP opt threshold : 7

```

When the receiver is not interested in the channel group any more, the receiver RX-4 sends an IGMPv3 leave, PE-4 sends a PIM prune translated to a BGP MP_UNREACH NLRI to all PEs. Note that, as mentioned before, rapid withdrawals are sent without waiting for the mrai (note that for simplicity, only one BGP update is shown in the output debug).

Provider Common Configuration

```
33 2011/10/13 16:20:17.74 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimJPPProcessSG
pimJPPProcessSG: (S,G)-> (192.168.1.1,232.1.1.1) type <S,G>, i/f int-PE-4-CE-4, u
pNbr 172.16.44.1 isJoin 0 isRpt 0 holdTime 210"

34 2011/10/13 16:20:17.74 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimJPPrintFsmEvent
PIM JP Downstream: State Joined Event RxPrune, (S,G) (192.168.1.1,232.1.1.1) gro
upType <S,G>"

35 2011/10/13 16:20:17.74 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimJPPrintFsmEvent
PIM JP Downstream: State PrunePending Event PrunePendTimerExp, (S,G) (192.168.1.
1,232.1.1.1) groupType <S,G>"

36 2011/10/13 16:20:17.74 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 2 vprn1]
"PIM[Instance 2 vprn1]: pimJPPrintFsmEvent
PIM JP Upstream: State Joined Event JoinDesiredFalse, (S,G) (192.168.1.1,232.1.1
.1) groupType <S,G>"

37 2011/10/13 16:20:17.74 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.2
"Peer 1: 192.0.2.2: UPDATE
Peer 1: 192.0.2.2 - Send BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 31
    Flag: 0x90 Type: 15 Len: 27 Multiprotocol Unreachable NLRI:
        Address Family MVPN_IPV4
        Type: Source-Join Len:22 RD: 64496:101 SrcAS: 64496 Src: 192.168.1.1 Grp
: 232.1.1.1
"
```

PMSI using RSVP-TE

Figure 209 shows the details of the topology for VPRN 2.

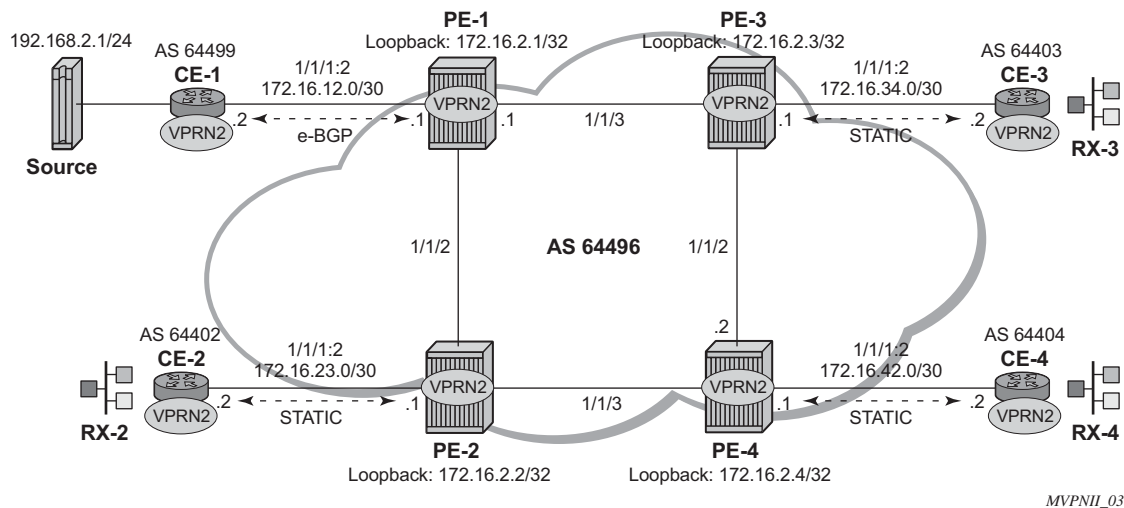


Figure 209: VPRN 2 Topology used for RSVP-TE P2MP

Unicast

For the sake of simplicity, check **Steps 1 to 6** in [PMSI using mLDP on page 1458](#) for VPRN 2 creation information. The same steps are repeated for RSVP using for provisioning the details that appear in [Figure 209](#). The result is the configuration in all the PEs, taking as an example PE-1:

```
*A:PE-1>config>service-vprn# info
-----
description "P2MP RSVP"
autonomous-system 64496
route-distinguisher 64496:201
vrf-target target:64496:200
interface "loopback" create
    address 172.16.2.1/32
    loopback
exit
interface "int-PE-1-CE-1" create
    address 172.16.12.1/30
    sap 1/1/1:2 create
    exit
exit
bgp
    group "external"
```

Provider Common Configuration

```
        type external
        peer-as 64499
        neighbor 172.16.12.2
        exit
    exit
    no shutdown
exit

spoke-sdp 12 create
    no shutdown
exit
spoke-sdp 13 create
    no shutdown
exit
spoke-sdp 14 create
    no shutdown
exit
no shutdown
```

In addition to the unicast, because RSVP is the signalling protocol to establish the P2MP LSPs, RSVP is configured on the interfaces. In addition, to use P2MP RSVP an LSP template is needed. The template defines the characteristics of the LSP to be created, for example, make-before-break, bandwidth, administrative groups, cspf, specific paths, etc. A basic template is used here. TE parameters specified in the template are commonly used in each RSVP PATH message for each of the branches of the P2MP RSVP LSP. The template is used in the mvpn context within the VPRN configuration (see [Auto-Discovery and RSVP PMSI Establishment on page 1477](#)). The resignal timer for P2MP is configured to the minimum value of sixty minutes (60 — 10080 minutes):

```
*A:PE-1>config>router>mpls# info
```

```
    p2mp-resignal-timer 60
    interface "system"
    exit
    interface "int-PE-1-PE-2"
    exit
    interface "int-PE-1-PE-3"
    exit
    path "empty"
        no shutdown
    exit
    lsp-template "vrf2" p2mp
        default-path "empty"
        cspf
        fast-reroute facility
    exit
    no shutdown
exit
no shutdown
```

Auto-Discovery and RSVP PMSI Establishment

The MP-BGP based auto-discovery is implemented with a new address family defined in RFC 4760 MP_REACH_NRLI/MP_UNREACH_NRLI attributes, with AFI 1 (IPv4) or 2 (IPv6) SAFI 5 (temporary value assigned by IANA). This is the mechanism by which each PE advertises the presence of an MVPN to other PEs. This can be achieved using PIM (like in Draft-Rosen) or using BGP. With the default parameter, BGP is automatically chosen because the PMSIs are RSVP and PIM is not an option in this case. Any PE that is a member of an MVPN will advertise to the other PEs using a BGP Multi-protocol Reachable Next-Hop Router Layer Information (NRLI) update that is sent to all PEs within the AS. This update will contain an Intra-AS I-PMSI auto-discovery Route type, also known as an Intra-AD. These use an address family mvpn-ipv4, so each PE must be configured to originate and accept such updates (note this was done earlier when configuring the families).

At this step (auto-discovery), the information about the PMSI is exchanged but the PMSI is not instantiated.

As each PE contains a CE which will be part of the multicast VRF, it is necessary to enable PIM on each interface containing the attachment circuit towards a CE, and to configure the I-PMSI multicast tunnel for the VRF. Note that S-PMSIs are not supported for RSVP with the 9.0R5 software release. In order for the BGP routes to be accepted into the VRF a route-target community is required (vrf-target). Although it is not mandatory for the mVPN target to be equal to the unicast target, the recommendation is to use vrf-target unicast to avoid configuration mistakes and extra complexity.

On each PE, a VPRN instance is configured as follows:

```
*A:PE-1>config>service>vprn# info
-----
      pim
      interface "loopback"
      exit
      interface "int-PE-1-CE-1"
      exit
      rp
        static
        exit
        bsr-candidate
        shutdown
        exit
        rp-candidate
        shutdown
        exit
      exit
      no shutdown
    exit
  mvpn
    auto-discovery default
    c-mcast-signaling bgp
    provider-tunnel
      inclusive
      rsvp
```

Provider Common Configuration

```
                lsp-template vrf2
                no shutdown
            exit
        exit
    exit
    vrf-target unicast
    exit
exit
```

The status of VPRN 2 on PE-1 is shown with the following output:

```
*A:PE-1>config>service>vprn# show router 2 mvpn
=====
MVPN 2 configuration data
=====
signaling          : Bgp                auto-discovery    : Default
UMH Selection      : Highest-Ip         intersite-shared   : Enabled
vrf-import         : N/A
vrf-export         : N/A
vrf-target         : unicast
C-Mcast Import RT  : target:192.0.2.1:3

ipmsi              : rsvp vrf2
i-pmsi P2MP AdmSt  : Up

s-pmsi             : none
data-delay-interval: 3 seconds
enable-asm-mdt     : N/A
=====
```

The following shows a debug of an Intra-AD BGP update message received by PE-1 that was sent by PE-4. The message contains the PMSI tunnel-type to be used (RSVP P2MP LSP), the P2MP LSP ID (encoded as extended Tunnel ID and P2MP-ID carried in the RSVP Session object), and the type of BGP update (Type: Intra-AD Len: 12 RD: 64496:204 Orig: 192.0.2.4):

```
41 2011/10/07 00:49:09.46 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.4
"Peer 1: 192.0.2.4: UPDATE
Peer 1: 192.0.2.4 - Received BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 86
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 8 Len: 4 Community:
        no-export
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:64496:200
    Flag: 0xc0 Type: 22 Len: 17 PMSI:
        Tunnel-type RSVP-TE P2MP LSP (1)
        Flags [Leaf not required]
        MPLS Label 0
        P2MP-ID 0x2, Tunnel-ID: 61441, Extended-Tunnel-ID 192.0.2.4
    Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
        Address Family MVPN_IPV4
        NextHop len 4 NextHop 192.0.2.4
```

```

Type: Intra-AD Len: 12 RD: 64496:204 Orig: 192.0.2.4
"

```

The set up has four PEs, so every PE should see the others peer Intra-AD route; the output below shows the routes received in PE-1:

```

*A:PE-1# show router bgp routes mvpn-ipv4 type intra-ad
=====
BGP Router ID:192.0.2.1      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP MVPN-IPv4 Routes
=====
Flag  RouteType      OriginatorIP      LocalPref  MED
      RD          SourceAS          VPNLabel
      Nexthop      SourceIP
      As-Path      GroupIP
-----
u*>i  Intra-Ad        192.0.2.2        100        0
      64496:202    -                -
      192.0.2.2    -                -
      No As-Path   -                -
u*>i  Intra-Ad        192.0.2.3        100        0
      64496:203    -                -
      192.0.2.3    -                -
      No As-Path   -                -
u*>i  Intra-Ad        192.0.2.4        100        0
      64496:204    -                -
      192.0.2.4    -                -
      No As-Path   -                -
-----
Routes : 3
=====

```

The detailed output of the Intra-AD received from PE-4 shows the Tunnel-Type RSVP-TE P2MP LSP (P2MP-ID is 2), the originator id (192.0.2.4), and the route-distinguisher (64496:204):

```

*A:PE-1# show router bgp routes mvpn-ipv4 type intra-ad originator-ip 192.0.2.4 detail
=====
BGP Router ID:192.0.2.1      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP MVPN-IPv4 Routes
=====
Route Type      : Intra-Ad
Route Dist.     : 64496:204
Originator IP   : 192.0.2.4
Nexthop         : 192.0.2.4
From            : 192.0.2.4
Res. Nexthop    : 0.0.0.0

```

Provider Common Configuration

```
Local Pref.      : 100
Aggregator AS    : None
Atomic Aggr.     : Not Atomic
Community        : no-export target:64496:200
Cluster          : No Cluster Members
Originator Id    : None
Flags            : Used Valid Best IGP
Route Source     : Internal
AS-Path          : No As-Path
VPRN Imported    : 2
Interface Name   : NotAvailable
Aggregator       : None
MED              : 0
Peer Router Id   : 192.0.2.4

-----
PMSI Tunnel Attribute :
Tunnel-type          : RSVP-TE P2MP LSP
MPLS Label           : 0
P2MP-ID              : 2
Extended-Tunne*:     : 192.0.2.4
Flags                : Leaf not required
Tunnel-ID             : 61441
-----
```

For the I-PMSI, the headend PE firstly discovers all the leaf PEs via I-PMSI A-D routes, it then signals the P2MP LSP to all the leaf PEs using RSVP-TE (subsequently adding or removing S2L (source to leaf) paths as PEs are added or removed from the MVPN). Note that RSVP-TE S-PMSIs are not supported in the tested release 9.0R5.

As in the mLDP case, the demarcation of the domains are in the PE. The PE router participates in both the customer multicast domain and the provider's multicast domain. The customer's CEs are limited to a multicast adjacency with the multicast instance on the PE created to support that specific customer's IP-VPN. This way, customers are isolated from the provider's core multicast domain and other customer multicast domains while the provider's core P routers only participate in the provider's multicast domain and are isolated from all customers's multicast domains. C-trees to P-tunnels bindings are also discovered using BGP routes, instead of PIM join TLVs. MVPN c-multicast routing information is exchanged between PEs by using c-multicast routes that are carried using MCAST-VPN NLRIs.

Once the RSVP-TE P2MP LSPs are created, the I-PMSI is instantiated in the core:

```
*A:PE-1# show router 2 pim neighbor
=====
PIM Neighbor ipv4
=====
Interface          Nbr DR Prty    Up Time      Expiry Time   Hold Time
Nbr Address
-----
int-PE-1-CE-1      1              0d 00:42:20   0d 00:01:34   105
172.16.12.2
mpls-if-73741      1              0d 00:42:20   never          65535
192.0.2.2
mpls-if-73742      1              0d 00:42:20   never          65535
192.0.2.3
mpls-if-73743      1              0d 00:42:20   never          65535
192.0.2.4
-----
Neighbors : 4
=====
```

```
*A:PE-1# show router 2 pim tunnel-interface
=====
PIM Interfaces ipv4
=====
```

Interface	Adm	Opr	DR Prty	Hello Intvl	Mcast Send
DR					

mpls-if-73740	Up	Up	N/A	N/A	N/A
192.0.2.1					
mpls-if-73741	Up	Up	N/A	N/A	N/A
192.0.2.2					
mpls-if-73742	Up	Up	N/A	N/A	N/A
192.0.2.3					
mpls-if-73743	Up	Up	N/A	N/A	N/A
192.0.2.4					

```
Interfaces : 4
```

The following command displays the PMSIs created on a PE, taking PE-3 as an example:

```
*A:PE-3# tools dump router 2 mvpn provider-tunnels
=====
MVPN 2 Inclusive Provider Tunnels Originating
=====
```

ipmsi (RSVP)	P2MP-ID	Tunl-ID	Ext-Tunl-ID

vrf2-2-73732	2	61440	192.0.2.3

```
=====
MVPN 2 Selective Provider Tunnels Originating
=====
```

spmsi (RSVP)	P2MP-ID	Tunl-ID	Ext-Tunl-ID

No Tunnels Found			

```
=====
MVPN 2 Inclusive Provider Tunnels Terminating
=====
```

ipmsi (RSVP)	P2MP-ID	Tunl-ID	Ext-Tunl-ID

mpls-if-73737	2	61452	192.0.2.1
mpls-if-73734	2	61440	192.0.2.2
mpls-if-73735	2	61441	192.0.2.4

```
=====
MVPN 2 Selective Provider Tunnels Terminating
=====
```

spmsi (RSVP)	P2MP-ID	Tunl-ID	Ext-Tunl-ID

No Tunnels Found			

Every PE has created an I-PMSI to the other PEs. As an example, PE-1 has established an LSP with name vrf2-2-73740 and PE-2, PE-3 and PE-4 as leaves. Note that the S2L path is empty as the template did not have any S2L path configured for simplicity.

```
*A:PE-1# show router mpls p2mp-lsp detail
=====
MPLS P2MP LSPs (Originating) (Detail)
=====
-----
Type : Originating
-----
LSP Name       : vrf2-2-73740
LSP Type       : P2mpAutoLsp
From           : 192.0.2.1
Adm State      : Up
LSP Up Time    : 0d 00:48:03
Transitions    : 1
Retry Limit    : 0
Signaling      : RSVP
Hop Limit      : 255
Adaptive       : Enabled
FastReroute    : Enabled
FR Method      : Facility
FR Bandwidth   : 0 Mbps
FR Object      : Enabled
CSPF           : Enabled
Metric         : Disabled
Include Grps   :
None
Least Fill    : Disabled

LSP Tunnel ID  : 61452
Oper State     : Up
LSP Down Time  : 0d 00:00:00
Path Changes   : 2
Retry Timer    : 30 sec
Resv. Style    : SE
Negotiated MTU : n/a
ClassType      : 0
Oper FR        : Enabled
FR Hop Limit   : 16
FR Node Protect : Disabled
ADSPEC         : Disabled
Use TE metric  : Disabled
Exclude Grps   :
None

Auto BW        : Disabled
LdpOverRsvp    : Disabled
IGP Shortcut   : Disabled
BGPTransTun    : Disabled
Oper Metric    : Disabled
Prop Adm Grp   : Disabled

VprnAutoBind   : Disabled
BGP Shortcut   : Disabled

CSPFFirstLoose : Disabled

P2MPInstance: 2
S2l-Name       : empty
S2l-Name       : empty
S2l-Name       : empty

P2MP-Inst-type : Primary
To             : 192.0.2.2
To             : 192.0.2.3
To             : 192.0.2.4
=====
```

Checking the RSVP-TE P2MP LSPs that are originated, transit, or destination to PE-1, the show command allows filtering by type, in this case showing the originated LSPs only:

```
*A:PE-1# show router mpls p2mp-info type originate
=====
MPLS P2MP LSPs (Originate)
=====
-----
S2L vrf2-2-73740::empty
-----
Source IP Address : 192.0.2.1
P2MP ID           : 2
Tunnel ID         : 61452
Lsp ID            : 38402
```

```

S2L Name           : vrf2-2-73740::empty   To           : 192.0.2.2
Out Interface      : 1/1/2:1               Out Label    : 262138
Num. of S2ls      : 2
-----
S2L vrf2-2-73740::empty
-----
Source IP Address  : 192.0.2.1             Tunnel ID    : 61452
P2MP ID           : 2                     Lsp ID      : 38402
S2L Name          : vrf2-2-73740::empty   To           : 192.0.2.3
Out Interface      : 1/1/3:1               Out Label    : 262128
Num. of S2ls      : 1
-----
S2L vrf2-2-73740::empty
-----
Source IP Address  : 192.0.2.1             Tunnel ID    : 61452
P2MP ID           : 2                     Lsp ID      : 38402
S2L Name          : vrf2-2-73740::empty   To           : 192.0.2.4
Out Interface      : 1/1/2:1               Out Label    : 262138
Num. of S2ls      : 2
-----
P2MP Cross-connect instances : 3
=====

```

Following the path of the S2L from PE-1 to PE-4 (third entry S2L vrf2-2-73740), the outgoing interface is 1/1/2 that connects PE-1 to PE-2, so the LSP goes to PE-4 via PE-2.

```

*A:PE-2# show router mpls p2mp-info type transit
=====
MPLS P2MP LSPs (Transit)
=====
-----
S2L vrf2-2-73740::empty
-----
Source IP Address  : 192.0.2.1             Tunnel ID    : 61452
P2MP ID           : 2                     Lsp ID      : 38402
S2L Name          : vrf2-2-73740::empty   To           : 192.0.2.4
Out Interface      : 1/1/3:1               Out Label    : 262136
Num. of S2ls      : 1
-----
S2L vrf2-2-73732::empty
-----
Source IP Address  : 192.0.2.4             Tunnel ID    : 61441
P2MP ID           : 2                     Lsp ID      : 17924
S2L Name          : vrf2-2-73732::empty   To           : 192.0.2.1
Out Interface      : 1/1/2:1               Out Label    : 262133
Num. of S2ls      : 1
-----
P2MP Cross-connect instances : 2
=====

```

As transit, PE-2 shows that there is an LSP coming from PE-1 (vrf2-2-73740) and the outgoing interface is 1/1/3 that connects PE-2 with PE-4.

Using the same command with a different filter on PE-4, 3 P2MP LSPs are terminated, one from each remote PE (PE-1, PE-2 and PE-3). On PE-4, an S2L vrf2-2-73740 from 192.0.2.1 and P2MP ID = 2 is traced.

```
*A:PE-4# show router mpls p2mp-info type terminate
=====
MPLS P2MP LSPs (Terminate)
=====
-----
S2L vrf2-2-73740::empty
-----
Source IP Address   : 192.0.2.1           Tunnel ID   : 61452
P2MP ID             : 2                   Lsp ID      : 38402
S2L Name            : vrf2-2-73740::empty To         : 192.0.2.4
In Interface        : 1/1/3:1           In Label    : 262136
Num. of S2ls        : 1
-----
S2L vrf2-2-73732::empty
-----
Source IP Address   : 192.0.2.2           Tunnel ID   : 61440
P2MP ID             : 2                   Lsp ID      : 8196
S2L Name            : vrf2-2-73732::empty To         : 192.0.2.4
In Interface        : 1/1/3:1           In Label    : 262132
Num. of S2ls        : 1
-----
S2L vrf2-2-73732::empty
-----
Source IP Address   : 192.0.2.3           Tunnel ID   : 61440
P2MP ID             : 2                   Lsp ID      : 59396
S2L Name            : vrf2-2-73732::empty To         : 192.0.2.4
In Interface        : 1/1/2:1           In Label    : 262142
Num. of S2ls        : 2
-----
P2MP Cross-connect instances : 3
=====
```

The following output shows P2MP LSP on PE-1 with more detail:

```
*A:PE-1# show router mpls p2mp-lsp "vrf2-2-73740" p2mp-instance "2" s2l "empty" detail
=====
MPLS LSP vrf2-2-73740 S2L empty (Detail)
=====
Legend :
  @ - Detour Available           # - Detour In Use
  b - Bandwidth Protected        n - Node Protected
  S - Strict                     L - Loose
  s - Soft Preemption
=====
-----
LSP vrf2-2-73740 S2L empty
-----
LSP Name      : vrf2-2-73740           S2l LSP ID   : 38402
P2MP ID       : 2                     S2l Grp Id   : 1
Adm State     : Up                    Oper State    : Up
S2l State     : Active                :
S2L Name      : empty                 To           : 192.0.2.2
S2l Admin     : Up                    S2l Oper     : Up
```



```

OutInterface: 1/1/2:1
S2L Up Time : 0d 01:22:50
RetryAttempt: 0
S2L Trans : 2
Failure Code: noError
ExplicitHops:
    No Hops Specified
Actual Hops :
    192.168.12.1(192.0.2.1) @
    -> 192.168.12.2(192.0.2.2)
ComputedHops:
    192.168.12.1(S) -> 192.168.12.2(S)
LastResignal: n/a

```

```

Out Label : 262138
S2L Dn Time : 0d 00:00:00
NextRetryIn : 0 sec
CSPF Queries: 2
Failure Node: n/a

```

```

Record Label : N/A
Record Label : 262138

```

```

-----
LSP vrf2-2-73740 S2L empty
-----

```

```

LSP Name : vrf2-2-73740
P2MP ID : 2
Adm State : Up
S2L State: : Active
S2L Name : empty
S2L Admin : Up
OutInterface: 1/1/3:1
S2L Up Time : 0d 01:22:50
RetryAttempt: 0
S2L Trans : 2
Failure Code: noError
ExplicitHops:
    No Hops Specified
Actual Hops :
    192.168.13.1(192.0.2.1) @
    -> 192.168.13.2(192.0.2.3)
ComputedHops:
    192.168.13.1(S) -> 192.168.13.2(S)
LastResignal: n/a

```

```

S2L LSP ID : 38402
S2L Grp Id : 2
Oper State : Up
:
To : 192.0.2.3
S2L Oper : Up
Out Label : 262128
S2L Dn Time : 0d 00:00:00
NextRetryIn : 0 sec
CSPF Queries: 2
Failure Node: n/a

```

```

Record Label : N/A
Record Label : 262128

```

```

-----
LSP vrf2-2-73740 S2L empty
-----

```

```

LSP Name : vrf2-2-73740
P2MP ID : 2
Adm State : Up
S2L State: : Active
S2L Name : empty
S2L Admin : Up
OutInterface: 1/1/2:1
S2L Up Time : 0d 01:22:54
RetryAttempt: 0
S2L Trans : 3
Failure Code: noError
ExplicitHops:
    No Hops Specified
Actual Hops :
    192.168.12.1(192.0.2.1) @
    -> 192.168.12.2(192.0.2.2) @
    -> 192.168.24.2(192.0.2.4)
ComputedHops:
    192.168.12.1(S) -> 192.168.12.2(S) -> 192.168.24.2(S)
LastResignal: n/a

```

```

S2L LSP ID : 38402
S2L Grp Id : 3
Oper State : Up
:
To : 192.0.2.4
S2L Oper : Up
Out Label : 262138
S2L Dn Time : 0d 00:00:00
NextRetryIn : 0 sec
CSPF Queries: 3
Failure Node: n/a

```

```

Record Label : N/A
Record Label : 262138
Record Label : 262136

```

```

=====

```

Provider Common Configuration

The last entry, vrf2-2-73740, provides the details of the S2L traced earlier, displaying the different hops (PE-1, PE-2, and PE-3), the fast reroute protection (link protection is supported only) and the labels used (262138 from PE-1 to PE-2, 262136 from PE-2 to PE-4). Note that on PE-1, although only one has been shown, both links PE-1 to PE-3 and PE-1 to PE-2 are fast reroute protected.

If any of the protected links between PE-1 and PE-2 or PE-3 are broken, fast reroute will be initiated. The protected bypass hops are displayed with the following command:

```
*A:PE-1# show router mpls bypass-tunnel protected-lsp p2mp detail
=====
MPLS Bypass Tunnels (Detail)
=====
-----
bypass-link192.168.12.2
-----
To                : 192.168.24.1          State           : Up
Out I/F           : 1/1/3:1              Out Label        : 262132
Up Time           : 0d 00:08:19           Active Time       : n/a
Reserved BW       : 0 Kbps                 Protected LSP Count : 3
Type              : P2mp
Setup Priority     : 7                      Hold Priority      : 0
Class Type        : 0
Actual Hops       :
    192.168.13.1(S)  -> 192.168.13.2(S)    -> 192.168.34.2(S)
    -> 192.168.24.1(S)

Protected LSPs -
LSP Name          : vrf2-2-73740::empty
From              : 192.0.2.1              To              : 192.0.2.2
Avoid Node/Hop    : 192.168.12.2           Downstream Label : 262138
Bandwidth         : 0 Kbps

LSP Name          : vrf2-2-73740::empty
From              : 192.0.2.1              To              : 192.0.2.4
Avoid Node/Hop    : 192.168.12.2           Downstream Label : 262138
Bandwidth         : 0 Kbps

LSP Name          : vrf2-2-73732::empty
From              : 192.0.2.3              To              : 192.0.2.2
Avoid Node/Hop    : 192.168.12.2           Downstream Label : 262137
Bandwidth         : 0 Kbps
-----
bypass-link192.168.13.2
-----
To                : 192.168.34.1          State           : Up
Out I/F           : 1/1/2:1              Out Label        : 262131
Up Time           : 0d 00:08:10           Active Time       : n/a
Reserved BW       : 0 Kbps                 Protected LSP Count : 2
Type              : P2mp
Setup Priority     : 7                      Hold Priority      : 0
Class Type        : 0
Actual Hops       :
    192.168.12.1(S)  -> 192.168.12.2(S)    -> 192.168.24.2(S)
    -> 192.168.34.1(S)

Protected LSPs -
LSP Name          : vrf2-2-73740::empty
```

```

From          : 192.0.2.1          To          : 192.0.2.3
Avoid Node/Hop : 192.168.13.2      Downstream Label : 262128
Bandwidth     : 0 Kbps

```

```

LSP Name      : vrf2-2-73732::empty
From          : 192.0.2.2          To          : 192.0.2.3
Avoid Node/Hop : 192.168.13.2      Downstream Label : 262127
Bandwidth     : 0 Kbps

```

```

=====

```

Traffic Flow

The receiver RX-4, connected to CE-4, wishes to join the group 232.2.2.2 with source 192.168.2.1 and so sends an IGMPv3 report towards CE-4. CE-4 recognizes the report and sends a PIM join towards the source, hence it reaches PE-1 where the source is connected to through CE-1. The output below shows the debug seen on PE-4, where the PIM join is received from CE-4 and a BGP update Source Join is sent to all PEs (note that only the update sent to PE-1 is shown).

```
40 2011/10/13 19:02:20.42 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
"PIM[Instance 3 vprn2]: pimJPProcessSG
pimJPProcessSG: (S,G)-> (192.168.2.1,232.2.2.2) type <S,G>, i/f int-PE-4-CE-4, u
pNbr 172.16.42.1 isJoin 1 isRpt 0 holdTime 210"
```

```
41 2011/10/13 19:02:20.42 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
"PIM[Instance 3 vprn2]: pimJPPrintFsmEvent
PIM JP Downstream: State NoInfo Event RxJoin, (S,G) (192.168.2.1,232.2.2.2) grou
pType <S,G>"
```

```
42 2011/10/13 19:02:20.42 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
"PIM[Instance 3 vprn2]: pimJPPrintFsmEvent
PIM JP Upstream: State NotJoined Event JoinDesiredTrue, (S,G) (192.168.2.1,232.2
.2.2) groupType <S,G>"
```

```
43 2011/10/13 19:02:20.42 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
"PIM[Instance 3 vprn2]: pimSGUpJoinDesiredTrue
No upstream interface. pSG (192.168.2.1,232.2.2.2) rpfType 3"
```

```
44 2011/10/13 19:02:20.42 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
"PIM[Instance 3 vprn2]: pimSGUpJoinDesiredTrue
pim 3 sg_type 2 refetch route type SPMSI pendingFetchMask 0x8"
```

```
45 2011/10/13 19:02:20.42 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
"PIM[Instance 3 vprn2]: pimSGUpJoinDesiredTrue
No upstream interface. pSG 0x5578f87c, (192.168.2.1,232.2.2.2) rpfType 3"
```

```
46 2011/10/13 19:02:20.42 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
"PIM[Instance 3 vprn2]: pimJPPrintFsmEvent
PIM JP Upstream: State Joined Event MribChange, (S,G) (192.168.2.1,232.2.2.2) gr
oupType <S,G>"
```

```
47 2011/10/13 19:02:20.42 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
"PIM[Instance 3 vprn2]: pimSGUpStateJMribChange
pSG 0x5578f87c, (192.168.2.1,232.2.2.2), type <S,G> oldMribNhopIp 0.0.0.0 oldRpf
NbrIp 0.0.0.0, oldRpfType NONE oldRpfIf 0 rptMribNhopIp 0.0.0.0, rptRpfNbrIp 0.0
.0.0 rtmReason 32"
```

```
48 2011/10/13 19:02:20.41 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
"PIM[Instance 3 vprn2]: pimSGUpStateJMribChange
pSG 0x5578f87c, (192.168.2.1,232.2.2.2), type <S,G> newMribNhopIp 192.0.2.1 newR
pNbrIp 192.0.2.1 newRpfType REMOTE newRpfIf 73737"
```

```
49 2011/10/13 19:02:20.41 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.1
"Peer 1: 192.0.2.1: UPDATE
Peer 1: 192.0.2.1 - Send BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 69
    Flag: 0x40 Type: 1 Len: 1 Origin: 0"
```

```

Flag: 0x40 Type: 2 Len: 0 AS Path:
Flag: 0x80 Type: 4 Len: 4 MED: 0
Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:192.0.2.1:3
Flag: 0x90 Type: 14 Len: 33 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.4
    Type: Source-Join Len:22 RD: 64496:201 SrcAS: 64496 Src: 192.168.2.1 Grp
: 232.2.2.2
"

```

The following debug shows that PE-1 receives the BGP update Source Join with source 192.168.2.1 and group 232.2.2.2 and sends a PIM join towards CE-1:

```

67 2011/10/14 05:11:35.06 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.4
"Peer 1: 192.0.2.4: UPDATE
Peer 1: 192.0.2.4 - Received BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 69
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:192.0.2.1:3
    Flag: 0x90 Type: 14 Len: 33 Multiprotocol Reachable NLRI:
        Address Family MVPN_IPV4
        NextHop len 4 NextHop 192.0.2.4
        Type: Source-Join Len:22 RD: 64496:201 SrcAS: 64496 Src: 192.168.2.1 Grp
: 232.2.2.2
"

68 2011/10/14 05:11:35.05 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
"PIM[Instance 3 vprn2]: pimJPPProcessSG
pimJPPProcessSG: (S,G)-> (192.168.2.1,232.2.2.2) type <S,G>, i/f mpls-if-73740, u
pNbr 192.0.2.1 isJoin 1 isRpt 0 holdTime 65535"

69 2011/10/14 05:11:35.06 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
"PIM[Instance 3 vprn2]: pimJPPrintFsmEvent
PIM JP Downstream: State NoInfo Event RxJoin, (S,G) (192.168.2.1,232.2.2.2) grou
pType <S,G>"

70 2011/10/14 05:11:35.06 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
"PIM[Instance 3 vprn2]: pimJPPrintFsmEvent
PIM JP Upstream: State NotJoined Event JoinDesiredTrue, (S,G) (192.168.2.1,232.2
.2.2) groupType <S,G>"

71 2011/10/14 05:11:35.06 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
"PIM[Instance 3 vprn2]: pimSGUpJoinDesiredTrue
pim 3 sg_type 2 refetch route type SPMSI pendingFetchMask 0x8"

72 2011/10/14 05:11:35.06 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
"PIM[Instance 3 vprn2]: pimSendJoinPrunePdu
pimSendJoinPrunePdu: if 3, adj 172.16.12.2"

73 2011/10/14 05:11:35.06 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]

```

Provider Common Configuration

```
"PIM[Instance 3 vprn2]: pimSGEncodeGroupSet
pimEncodeGroupSet: encoding groupset for group 232.2.2.2, numJoinedSrcs 1, numPrunedSrcs 0"
```

```
74 2011/10/14 05:11:35.05 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
```

```
"PIM[Instance 3 vprn2]: pimSGEncodeGroupSet
pimEncodeGroupSet: Encoding Join for source 192.168.2.1"
```

```
75 2011/10/14 05:11:35.05 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
```

```
"PIM[Instance 3 vprn2]: pimSGEncodeGroupSet
pimEncodeGroupSet: num joined srcs 1, num pruned srcs 0"
```

```
76 2011/10/14 05:11:35.05 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
```

```
"PIM[Instance 3 vprn2]: pimSendJoinPrunePdu
pimSendJoinPrunePdu2: sending JP PDU with 1 groups."
```

```
77 2011/10/14 05:12:34.76 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
```

```
"PIM[Instance 3 vprn2]: pimJPPrintFsmEvent
PIM JP Upstream: State Joined Event JTimerExp, (S,G) (192.168.2.1,232.2.2.2) groupType <S,G>"
```

```
78 2011/10/14 05:12:34.76 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
```

```
"PIM[Instance 3 vprn2]: pimSendJoinPrunePdu
pimSendJoinPrunePdu: if 3, adj 172.16.12.2"
```

```
79 2011/10/14 05:12:34.76 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
```

```
"PIM[Instance 3 vprn2]: pimSGEncodeGroupSet
pimEncodeGroupSet: encoding groupset for group 232.2.2.2, numJoinedSrcs 1, numPrunedSrcs 0"
```

```
80 2011/10/14 05:12:34.76 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
```

```
"PIM[Instance 3 vprn2]: pimSGEncodeGroupSet
pimEncodeGroupSet: Encoding Join for source 192.168.2.1"
```

```
81 2011/10/14 05:12:34.77 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
```

```
"PIM[Instance 3 vprn2]: pimSGEncodeGroupSet
pimEncodeGroupSet: num joined srcs 1, num pruned srcs 0"
```

```
82 2011/10/14 05:12:34.77 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
```

```
"PIM[Instance 3 vprn2]: pimSendJoinPrunePdu
pimSendJoinPrunePdu2: sending JP PDU with 1 groups."
```

The BGP update source join received by PE-1 is displayed with the following command:

```
*A:PE-1# show router bgp routes mvpn-ipv4 type source-join
=====
BGP Router ID:192.0.2.1      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP MVPN-IPv4 Routes
=====
Flag  RouteType      OriginatorIP      LocalPref  MED
      RD          SourceAS          VPNLabel
      Nexthop      SourceIP
      As-Path      GroupIP
-----
u*>i  Source-Join      -                  100        0
      64496:201      64496
      192.0.2.4      192.168.2.1
      No As-Path     232.2.2.2
-----
Routes : 1
=====
```

To verify the traffic: on PE-1 there is a group 232.2.2.2 with source 192.168.2.1, the RPF is CE-1, and the multicast traffic is flowing from CE-1 to PE-1 using int-PE-1-CE-1 and the outgoing interface is using the PMSI RSVP mpls-if-73740.

```
*A:PE-1# show router 2 pim group detail
=====
PIM Source Group ipv4
=====
Group Address      : 232.2.2.2
Source Address     : 192.168.2.1
RP Address         : 0
Flags              :
Type               : (S,G)
MRIB Next Hop     : 172.16.12.2
MRIB Src Flags     : remote
Up Time           : 0d 00:10:48
Keepalive Timer    : Not Running
Resolved By       : rtable-u

Up JP State        : Joined
Up JP Rpt          : Not Joined StarG
Up JP Expiry       : 0d 00:00:12
Up JP Rpt Override : 0d 00:00:00

Register State     : No Info
Reg From Anycast RP: No

Rpf Neighbor       : 172.16.12.2
Incoming Intf      : int-PE-1-CE-1
Outgoing Intf List : mpls-if-73740

Curr Fwding Rate   : 0.0 kbps
Forwarded Packets  : 0
Discarded Packets  : 0
Forwarded Octets   : 0
RPF Mismatches     : 0
Spt threshold      : 0 kbps
ECMP opt threshold : 7
```

Provider Common Configuration

```
Admin bandwidth      : 1 kbps
```

```
-----  
Groups : 1  
=====
```

On PE-4, the same (S,G) arrives in the incoming interface mpls-if-73737 and the outgoing interface is int-PE-4-CE-4.

```
*A:PE-4# show router 2 pim group detail
```

```
=====
```

```
PIM Source Group ipv4
```

```
=====
```

Group Address	: 232.2.2.2		
Source Address	: 192.168.2.1		
RP Address	: 0		
Flags	:	Type	: (S,G)
MRIB Next Hop	: 192.0.2.1		
MRIB Src Flags	: remote	Keepalive Timer	: Not Running
Up Time	: 0d 00:12:40	Resolved By	: rtable-u
Up JP State	: Joined	Up JP Expiry	: 0d 00:00:19
Up JP Rpt	: Not Joined StarG	Up JP Rpt Override	: 0d 00:00:00

```
Register State      : No Info  
Reg From Anycast RP: No
```

```
Rpf Neighbor        : 192.0.2.1  
Incoming Intf       : mpls-if-73737  
Outgoing Intf List  : int-PE-4-CE-4
```

```
Curr Fwding Rate    : 0.0 kbps  
Forwarded Packets   : 0  
Forwarded Octets    : 0  
Spt threshold       : 0 kbps  
Admin bandwidth     : 1 kbps
```

Discarded Packets	: 0
RPF Mismatches	: 0
ECMP opt threshold	: 7

```
-----  
Groups : 1  
=====
```

When the receiver is not interested in the channel group any more, the receiver RX-4 sends an IGMPv3 leave, PE-4 sends a PIM prune translated to a BGP MP_UNREACH NLRI to all PEs. Note that, as mentioned before, rapid withdrawals are sent without waiting for the mrai (note that for simplicity, only one BGP update is shown in the output debug).

```
92 2011/10/13 19:16:36.67 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]  
"PIM[Instance 3 vprn2]: pimJPProcessSG  
pimJPProcessSG: (S,G)-> (192.168.2.1,232.2.2.2) type <S,G>, i/f int-PE-4-CE-4, u  
pNbr 172.16.42.1 isJoin 0 isRpt 0 holdTime 210"
```

```
93 2011/10/13 19:16:36.67 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]  
"PIM[Instance 3 vprn2]: pimJPPrintFsmEvent  
PIM JP Downstream: State Joined Event RxPrune, (S,G) (192.168.2.1,232.2.2.2) gro  
upType <S,G>"
```

```
94 2011/10/13 19:16:36.67 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
```



```

"PIM[Instance 3 vprn2]: pimJPrintFsmEvent
PIM JP Downstream: State PrunePending Event PrunePendTimerExp, (S,G) (192.168.2.
1,232.2.2.2) groupType <S,G>"

95 2011/10/13 19:16:36.67 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
"PIM[Instance 3 vprn2]: pimJPrintFsmEvent
PIM JP Upstream: State Joined Event JoinDesiredFalse, (S,G) (192.168.2.1,232.2.2
.2) groupType <S,G>"

96 2011/10/13 19:16:36.67 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.1
"Peer 1: 192.0.2.1: UPDATE
Peer 1: 192.0.2.1 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 31
  Flag: 0x90 Type: 15 Len: 27 Multiprotocol Unreachable NLRI:
    Address Family MVPN_IPV4
    Type: Source-Join Len:22 RD: 64496:201 SrcAS: 64496 Src: 192.168.2.1 Grp
: 232.2.2.2
"

```

MVPN Source Redundancy

So far, the multicast traffic has been streamed towards router CE-1 from a single source. For redundancy purposes, the source can be redundant (two sources attached to different CEs that connect to a pair of PEs). To simulate the redundancy, CE-1 has been connected to both PE-1 and PE-2, using VPRN 2, and ECMP (Equal Cost Multi Path) is configured with the value of 2 in all PEs. With this configuration, any PE is able to reach the source through PE-1 and PE-2. The (S,G) is the same as the one used in P2MP RSVP TE (192.168.2.1, 232.2.2.2). [Figure 210](#) shows the VPRN 2 topology with the source redundancy.

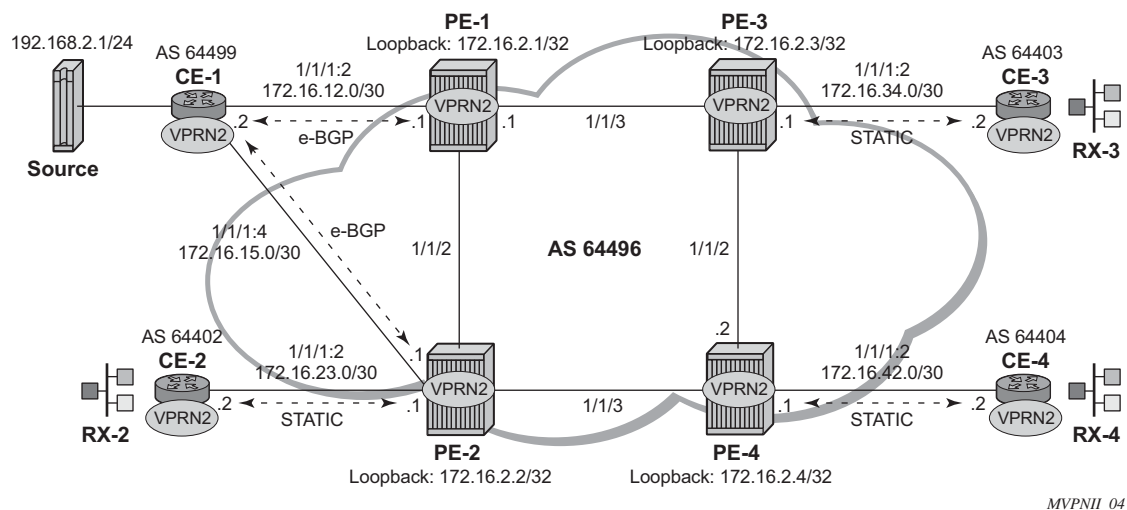


Figure 210: VPRN 2 Topology used for MVPN Source Redundancy

Provider Common Configuration

The configuration change with respect to the previous section (P2MP RSVP-TE PMSIs) is an additional interface created in both CE-1 and PE-2 (int-CE-1-PE-2 on CE-1 and int-PE-2-CE-1 on PE-2), the addition of these interfaces to pim and also the creation an e-BGP session between the two routers. The following is the configuration on PE-2 (CE-1 configuration changes are not displayed for brevity).

```
*A:PE-2>config>service>vprn# info
-----
description "P2MP RSVP"
ecmp 2
autonomous-system 64496
route-distinguisher 64496:202
vrf-target target:64496:200
interface "loopback" create
    address 172.16.2.2/32
    loopback
exit
interface "int-PE-2-CE-2" create
    address 172.16.23.1/30
    sap 1/1/1:2 create
    exit
exit
interface "int-PE-2-CE-1" create
    address 172.16.16.1/30
    sap 1/1/4:2 create
    exit
exit
bgp
    group "external"
        type external
        peer-as 64499
        neighbor 172.16.16.2
        exit
    exit
    no shutdown
exit
pim
    interface "loopback"
    exit
    interface "int-PE-2-CE-2"
    exit
    interface "int-PE-2-CE-1"
    exit
    rp
        static
        exit
        bsr-candidate
            shutdown
        exit
        rp-candidate
            shutdown
        exit
    exit
    no shutdown
exit
mvpn
    auto-discovery default
    c-mcast-signaling bgp
```

```

        provider-tunnel
            inclusive
            rsvp
                lsp-template vrf2
                no shutdown
            exit
        exit
    exit
    vrf-target unicast
    exit
exit
spoke-sdp 12 create
    no shutdown
exit
spoke-sdp 23 create
    no shutdown
exit
spoke-sdp 24 create
    no shutdown
exit
no shutdown

```

Checking the routes on PE-4, the source is reachable through PE-1 and PE-2 as ECMP is set to 2. Note that if the configuration of the VPRN is provisioned with **autobind-mpls** instead of static spoke-SDPs, the command **ignore-nh-metric** is also needed.

```

*A:PE-4# show router 2 route-table
=====
Route Table (Service: 2)
=====
Dest Prefix[Flags]                                Type   Proto   Age      Pref
  Next Hop[Interface Name]                        Metric
-----
* Truncated info

192.168.2.1/32                                     Remote BGP VPN 00h19m55s 170
    192.0.2.1 (tunneled)                           0
192.168.2.1/32                                     Remote BGP VPN 00h19m55s 170
    192.0.2.2 (tunneled)                           0
-----
No. of Routes: 11
Flags: L = LFA nexthop available    B = BGP backup route available
      n = Number of times nexthop is repeated
=====

```

When PE-4 receives a c-join/prune, PE-4 needs to find the **upstream multicast hop** for the (S,G). This is the Upstream Multihop Selection (UMH) and is configurable. The values are highest-ip, hash-based and tunnel-status.

```

*A:PE-4>config>service>vprn>mvpn# umh-selection
- no umh-selection
- umh-selection {highest-ip|hash-based|tunnel-status}

```

Provider Common Configuration

The default is highest-ip, which is the selection of the highest /32 IP addresses (in this set up, PE-2 is preferred versus PE-1). A BGP c-join is sent with the route target equal to the VRF import extended community distributed by PE-2 for the subnet of the source (see PE-4 debug show below).

```
* Truncated info
12 2011/10/14 08:42:53.87 UTC MINOR: DEBUG #2001 vprn2 PIM[Instance 3 vprn2]
"PIM[Instance 3 vprn2]: pimSGUpStateJMribChange
pSG 0x5578f87c, (192.168.2.1,232.2.2.2), type <S,G> newMribNhopIp 192.0.2.2 newR
pfNbrIp 192.0.2.2 newRpftype REMOTE newRpfiif 73741"

13 2011/10/14 08:42:53.87 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.2
"Peer 1: 192.0.2.2: UPDATE
Peer 1: 192.0.2.2 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 69
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:192.0.2.2:3
  Flag: 0x90 Type: 14 Len: 33 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.4
    Type: Source-Join Len:22 RD: 64496:202 SrcAS: 64496 Src: 192.168.2.1 Grp
: 232.2.2.2
"
```

The second option is hash-based, where the UMH is selected (both PEs are potentially possible UMHs) after hashing the stream's source and group addresses. For this example, PE-2 is also preferred.

The third option, tunnel-status, is based on the status of the P2MP RSVP tunnel (not available in mLDP or PIM). The roots PE-1 and PE-2 are sending BFD messages to the leaf PE-4 (in fact this is UFD, unidirectional forwarding detection). PE-4's c-join for the (S,G) is sent to both PE-1 and PE-2, and in return the traffic is forwarded from both PE-1 and PE-2 for the c-group onto the I-PMSI; hence PE-4 receives two copies of the c-(S,G) stream. By configuration, the stream from the primary PE-1 is selected by PE-4 to be forwarded to receiver RX-4. If BFD messages are no longer received over the primary P2MP LSP, then stream from the standby PE-2 is selected and forwarded to the receiver.

The configuration on PE-1 and PE-2 is similar and is as follows (only PE-2 is shown):

```
PE-2>configure service vprn 2
      mvpn
        auto-discovery default
        c-mcast-signaling bgp
        umh-selection tunnel-status
        provider-tunnel
          inclusive
          rsvp
            lsp-template vrf2
            enable-bfd-root 100
```

```

        no shutdown
    exit
    exit
    exit
    vrf-target unicast
    exit
exit

```

PE-1 and PE-2 are root. On PE-4 BFD is configured as leaf and the primary PE (PE-1) and backup PE (PE-2) are also provisioned:

```

PE-4>configure service vprn 2
    mvpn
        auto-discovery default
        c-mcast-signaling bgp
        umh-selection tunnel-status
        umh-pe-backup
            umh-pe 192.0.2.1 standby 192.0.2.2
        exit
        provider-tunnel
            inclusive
            rsvp
                lsp-template vrf2
                enable-bfd-leaf
                no shutdown
            exit
        exit
    exit
    vrf-target unicast
    exit
exit

```

This BFD (UFD) configuration on the root establishes a session with the leaf where BFD packets are only transmitted:

```

*A:PE-1>config>service>vprn# show router 2 bfd session
=====
BFD Session
=====

```

Interface	State	Tx Intvl	Rx Intvl	Multipl
Remote Address	Protocols	Tx Pkts	Rx Pkts	Type
mpls-if-73747	Up (3)	100	0	3
127.0.0.0	pim	9955	0	central

```

-----
No. of BFD sessions: 1

```

On PE-4, two BFD sessions are received, one from each root (note that BFD packets are only received):

```

*A:PE-4>config>service>vprn# show router 2 bfd session
=====
BFD Session
=====

```

Interface Remote Address	State Protocols	Tx Intvl Tx Pkts	Rx Intvl Rx Pkts	Multipl Type
-----	-----	-----	-----	-----
mpls-if-73747	Up (3)	0	100	3
192.0.2.1	pim	0	681	central
mpls-if-73748	Up (3)	0	100	3
192.0.2.2	pim	0	470	central
-----	-----	-----	-----	-----
No. of BFD sessions: 2				
=====				

PE-4 delivers the multicast traffic from the primary configured UMH, PE-1. If, as an example of a failure condition, PE-1 goes down (reboot), PE-4 will switch to the PE-2 P2MP LSP.

MDT AFI SAFI

In the Draft-Rosen up to version 6, the default MDT is PIM Spare Mode only, and there is no auto-discovery mechanism available. From SR-OS Release 7.0 and beyond, it is possible to configure PIM SSM with auto-discovery, using AFI 1 and SAFI 5. The Draft-Rosen version 7 allows use of MDT PIM SM or SSM, and auto-discovery based on AFI 1 and SAFI 66 to distribute the default MDT. Draft-Rosen version 9 adds a new MDT NLRI. The SR-OS has added the capability and support of MDT SAFI as specified in draft-nalawade-idr-mdt-safi-03 and draft-roten-vpn-mcast-15.

MDT SAFI is used to discover PEs in a specific MVPN, so that PIM SSM can be used for default MDT. The basic idea is the same as MVPN BGP auto-discovery, but it uses a different BGP SAFI. BGP messages in which AFI=1 and SAFI=66 are "MDT-SAFI" messages. The NLRI format is 8-byte-RD:IPv4-address followed by the MDT group address. The IPv4 address identifies the PE that originated this route and the RD identifies a VRF in that PE. The group address must be an IPv4 multicast group address and is used to build the P-tunnels.

All PEs attached to a given MVPN must specify the same group-address. MDT-SAFI routes do not carry RTs and the group address is used to associate a received MDT-SAFI route with a VRF.

MDT SAFI can only be used when the implicit provider tunnel is PIM GRE based with a specific IPv4 group address.

Note: For additional information on the use of PIM PMSIs, refer to [Multicast in a VPN I on page 1389](#).

[Figure 211](#) shows the topology of VPRN 3.

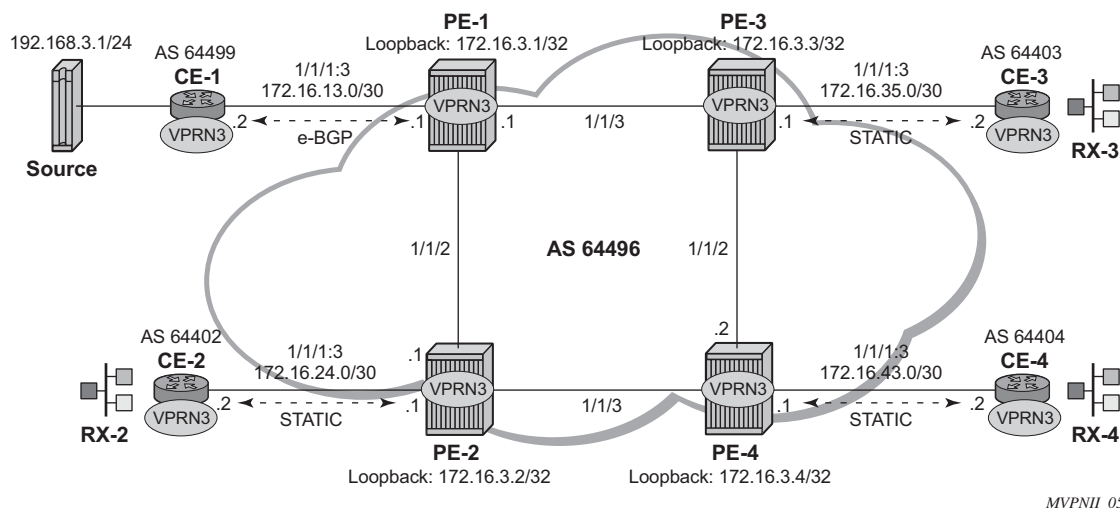


Figure 211: VPRN 3 Topology used for MVPN Source Redundancy

In this scenario, there is no MPLS based PMSI, there is PIM in the core for the control plane and the data traffic is GRE encapsulated. PIM needs to be configured in the base router on interface system and on the other interfaces pointing to other PEs. PIM is used for c-signaling. In addition, auto-discovery is provisioned to use mdt-safi and a PIM SSM inclusive PMSI with address 239.1.1.1 as the default MDT. The configuration, in all PEs, is like the following on PE-4:

```
*A:PE-4>config router
    pim
        interface "system"
        exit
        interface "int-PE-4-PE-2"
        exit
        interface "int-PE-4-PE-3"
        exit
    rp
        static
        exit
        bsr-candidate
        shutdown
        exit
        rp-candidate
        shutdown
        exit
    exit
    no shutdown
exit

*A:PE-4>config service vprn 3
    description "PIM SSM"
    autonomous-system 64496
```

Provider Common Configuration

```
route-distinguisher 64496:304
vrf-target target:64496:300
interface "loopback" create
    address 172.16.3.4/32
    loopback
exit
interface "int-PE-4-CE-4" create
    address 172.16.43.1/30
    sap 1/1/1:3 create
    exit
exit
pim
    interface "loopback"
    exit
    interface "int-PE-4-CE-4"
    exit
    rp
        static
        exit
        bsr-candidate
            shutdown
        exit
        rp-candidate
            shutdown
        exit
    exit
    no shutdown
exit
mvpn
    auto-discovery mdt-safi
    provider-tunnel
        inclusive
        pim ssm 239.1.1.1
        exit
    exit
    vrf-target unicast
    exit
exit
spoke-sdp 14 create
    no shutdown
exit
spoke-sdp 24 create
    no shutdown
exit
spoke-sdp 34 create
    no shutdown
exit
no shutdown
```


The following debug output shows a BGP update with MDT AFI SAFI:

```
2 2011/10/14 16:48:14.58 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.1
"Peer 1: 192.0.2.1: UPDATE
Peer 1: 192.0.2.1 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 62
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:64496:300
  Flag: 0x90 Type: 14 Len: 26 Multiprotocol Reachable NLRI:
    Address Family MDT-SAFI
    NextHop len 4 NextHop 192.0.2.4
    [MDT-SAFI] Addr 192.0.2.4, Group 239.1.1.1, RD 64496:304
```

Conclusion

This chapter provides information to configure multicast within a VPRN with next generation multicast VPN techniques. Specifically, the use of MPLS I-PMSIs (mLDP and P2MP RSVP-TE), MVPN source redundancy and the complete set of features needed to interoperate with Draft-Rosen in live deployments are covered.

Multicast VPN: Core Diversity

In This Chapter

This section provides information about multicast VPN core diversity.

Topics in this section include:

- [Applicability on page 1504](#)
- [Overview on page 1505](#)
- [Configuration on page 1511](#)
- [Conclusion on page 1531](#)

Applicability

This chapter is applicable to 7450 ESS-7/12 with IOM3-XP or IMM, 7750 SR-7/12, 7750 c4/c12, and 7950 XRS. Multicast VPN Core Diversity can be configured on FP2- or FP3-based IOMs or IMM. Chassis mode D is required.

The configuration was tested on release 13.0.R4, using Rosen Multicast Virtual Private Network (MVPN) techniques. Default Multicast Distribution Trees (MDTs) for each Virtual Private Routed Network (VPRN) are signaled using Protocol Independent Multicasting (PIM) and auto-discovery uses Border Gateway Protocol Multicast Distribution Tree Sub-Address Family Indicator (BGP MDT-SAFI) network layer routing.

Overview

This chapter describes a service provider core network used by multiple content providers to deliver multicast services to multiple customers using Rosen MVPN. If the same set of PEs is used to deliver the MVPN, the MDTs will all be routed across the same paths between the set of PEs. Because each MDT is signaled using PIM, and the source of all MDTs is the system address of the PE, the path to this source is the same.

Each remote PE then sends a PIM join toward this PE with its source address set to the system address. For multiple VPNs between the same set of PEs, the MDT will follow the same path.

If there is a requirement to deliver content from each content provider across different MVPNs that use diversely routed MDTs, multiple IGP instances can be used: up to three different instances of IGP, OSPF, or ISIS can exist. In this chapter, two instances of OSPF are used to create incongruent topologies providing isolation between the MDTs of two different MVPNs: a default OSPF instance and OSPF instance 1. A separate /32 loopback address can be used as the MDT source address that is advertised in the non-default IGP, which can also be used as the BGP next hop for labeled IPv4 routes representing the customer source addresses.

Knowledge of Multi-Protocol BGP (MP-BGP) and RFC 4364 (BGP/MPLS IP VPNs) is assumed throughout this chapter, as well as the original RFC 6037 (Cisco Systems' Solution for Multicast in BGP/MPLS IP VPNs).

All PEs within an MVPN create a default MDT with their own system address as the source. Auto-discovery of PEs within a Rosen MVPN is achieved using the BGP route type of Multicast Data Tree Subsequent Address Family Identifier (MDT-SAFI). Each PE originates an MDT-SAFI route update per MVPN. This route advertises the presence of the MVPN on a specific PE.

Each MDT-SAFI update contains attributes, including the following:

- a. Route distinguisher
- b. Route target extended community
- c. MDT source address (usually the system address)
- d. Group address of MDT

Upon receipt of an MDT-SAFI route update, each remote PE accepts or rejects the route based on the Route Target extended community value. If the route is accepted, a remote PE sends a PIM (S,G) join to this local PE. The (S,G) values are taken from the MDT-SAFI. The set of MDTs extend the c-multicast data tree across the MVPN and form PIM adjacencies between PEs within the MVPN. The neighbor address across the set of PIM-enabled tunnels is the default MDT source address, usually the system address.

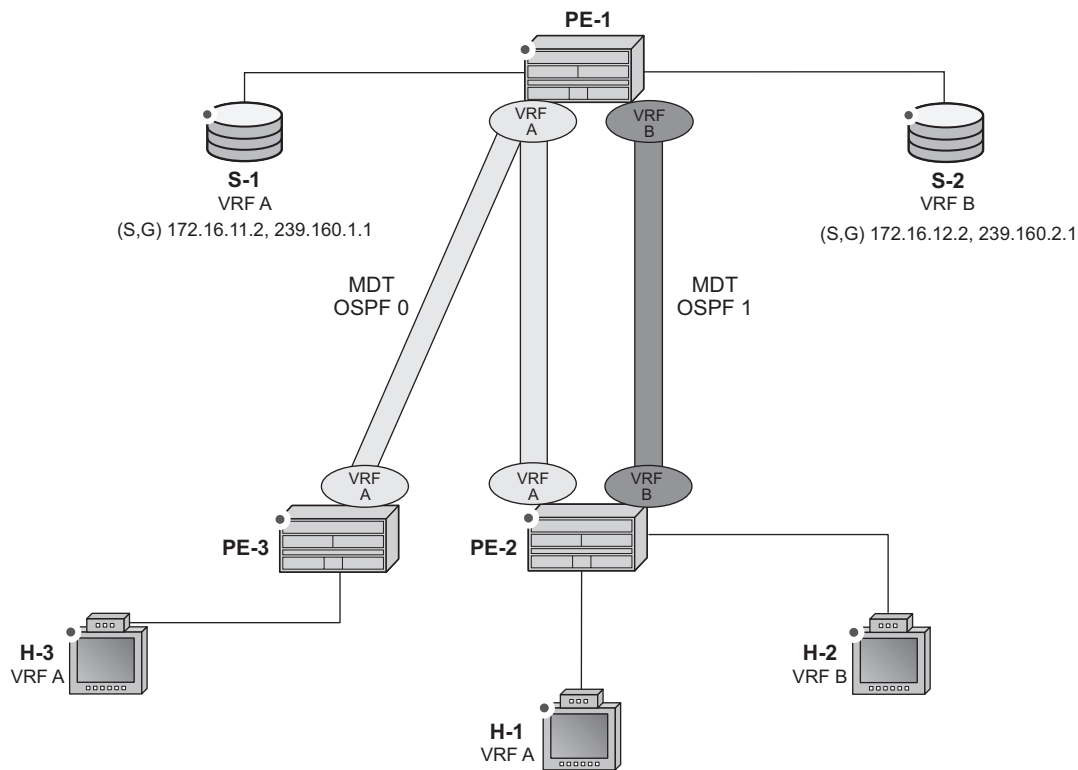
When established, the default MDT is used to transport c-multicast PIM signaling between PEs.

If a source S, of a c-multicast group G, is connected to a sender PE, the route to the source is advertised to remote PEs as a BGP-labeled VPN-IPv4 route.

Therefore, an (S,G) join toward this source at a remote PE will perform a reverse path forwarding (RPF) look-up of the unicast VRF table to find a suitable PIM-enabled interface. The next hop needs be resolved to the MDT source address of the sender PE. A PIM join must now be forwarded toward the sender PE that has a PIM neighbor that matches the next hop for this route, the system address of the sender PE. This is the default MDT.

The system address is a significant address in this process. Any other VPRN that uses the same set of PEs will also signal a set of default MDTs using a different group address, so they will follow the same path between PEs across the provider network.

Figure 212 shows an example of core diversity; multicast sources provided by two separate content providers are connected to a provider network. There is a requirement to provide topology diversity so that the default MDTs between the same PEs are routed across different paths within the core.



al_0794

Figure 212: Core Diversity Schematic

Content servers from two separate content providers are connected to PE-1 with directly connected multicast sources. For simplicity, this example uses only a single multicast group for provider S-1 and S-2.

Source S-1 is reachable via VRF A and source S-2 is reachable via VRF B.

Topology isolation for the Multicast Data Trees of each VPRN can be provided by using two separate IGP instances; in this case, OSPF instances. Multi-instance IS-IS could also be used.

Figure 213 shows a schematic of the full network, including the c-multicast groups.

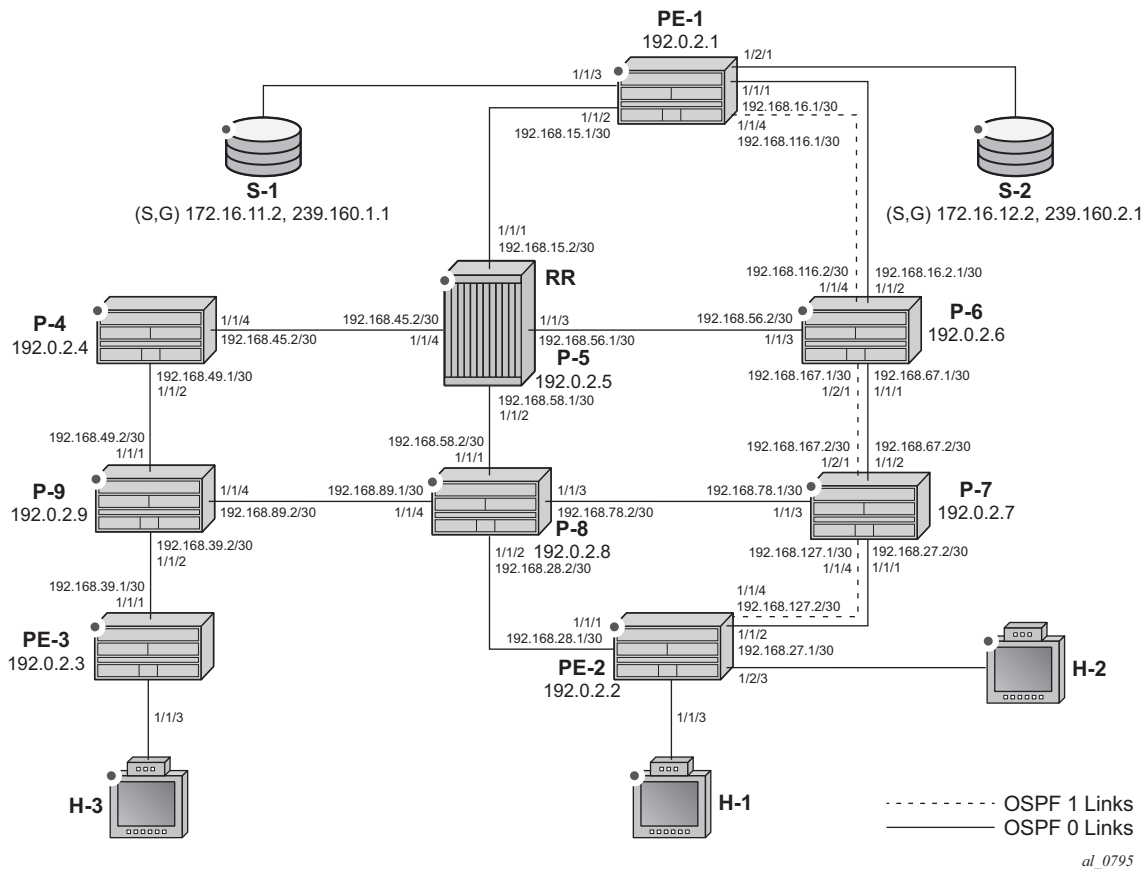


Figure 213: Core Diversity Network

All routers have interfaces in the OSPF base instance (instance 0) and routers interconnected by the dotted lines have interfaces in both the base instance and OSPF 1.

Figure 213 shows the extent of the OSPF base instance within the core network.

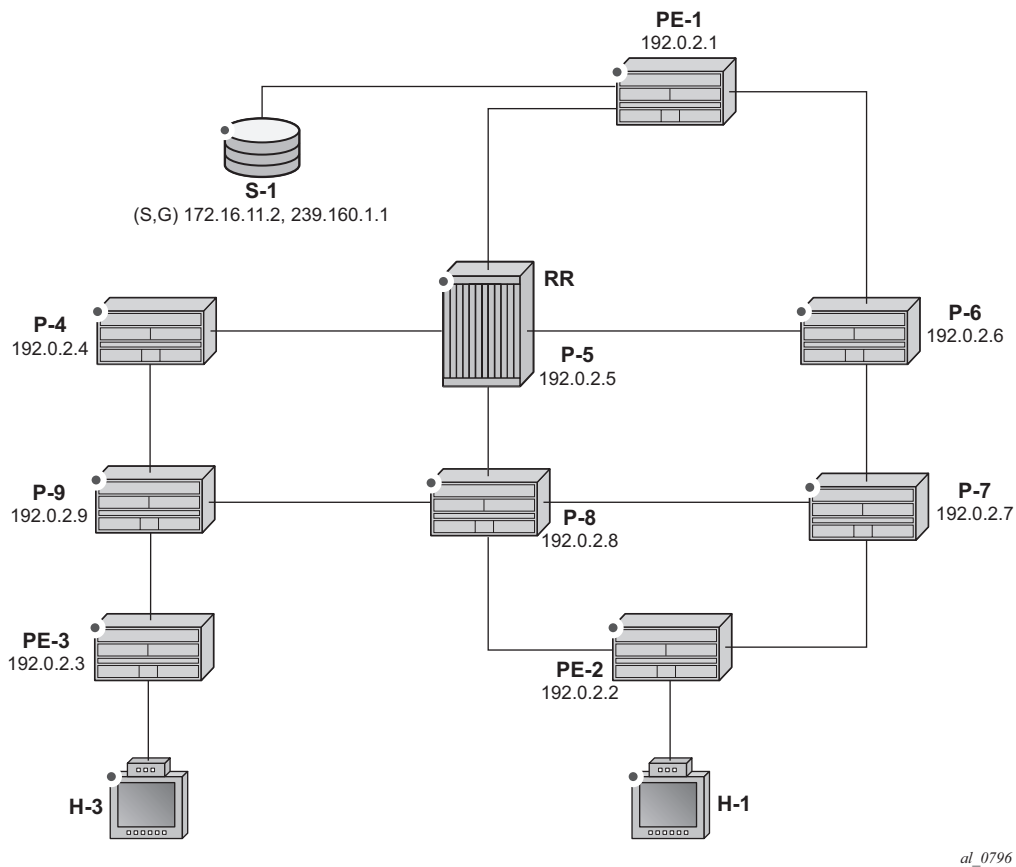


Figure 214: Core Diversity Network — Base OSPF

In this case, assume that the shortest path between PE-1 and PE-2 is the path PE-1 → P-5 → P-8 → PE-2.

Similarly, [Figure 215](#) shows the extent of OSPF instance 1.

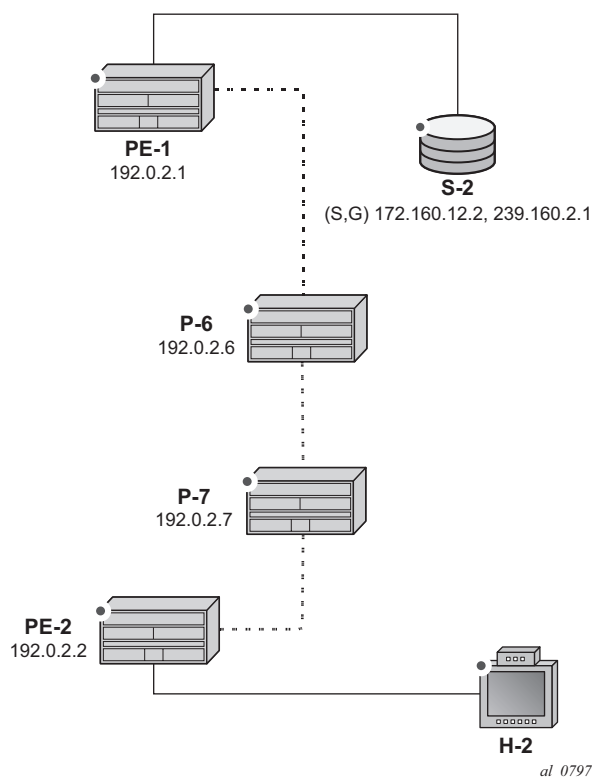


Figure 215: Core Diversity Network - OSPF Instance 1

The only path available between PE-1 and PE-2 is now completely diverse from the shortest path advertised between the same pair of PEs in the base OSPF instance.

Therefore, for any MDT to be signaled across the OSPF 1 topology, only addresses advertised within OSPF 1 must be used. As previously stated, the system address is used as the default MDT source address. This system address is not advertised within the OSPF 1 topology, so a replacement /32 loopback address is used as the default MDT source address within OSPF 1.

VPN-IPv4 routes that may represent a customer multicast source address should be reachable via the default MDT. In the non-default topology, the c-multicast signaling across the MVPN must resolve the c-multicast route via the MDT, which has its root at the non-default /32 loopback. Therefore, the VPN-IPv4 prefix representing the possible source routes needs to be advertised containing the non-default /32 loopback.

This can be achieved in one of two ways:

- a. Use a route policy at the advertising PE that changes the BGP next hop to match the MDT source address for non-default topology MDTs.
- b. Use the BGP connector attribute for all VPN-IPv4 route prefixes within a multicast VPRN that has auto-discovery set to MDT-SAFI. The connector attribute contains the MDT source address within the originator field.

This chapter describes the use of the connector attribute.

If the default IGP instance is used, the BGP next hop of the VPN-IPv4 route matches the source address of the default MDT.

Therefore, if a second /32 loopback is used that replaces the system address as MDT source address and also as the next hop for source address RPF look-up, the loopback could be advertised within the non-default IGP instance, and the paths between the PEs would follow this topology.

Core Diversity is achieved by configuring the following steps:

- a. Configuring multiple OSPF instances, as shown in [Figure 214](#) and [Figure 215](#), and including the appropriate interfaces. This includes a separate loopback address per instance.
- b. Configuring separate VPRNs with their own MDTs using BGP MDT-SAFI auto-discovery and PIM signaling across the appropriate PEs.
- c. Configuring the VPRN that uses the base OSPF instance to use the system address as the source addresses for the MDTs (this is default behavior).
- d. Configuring a separate loopback (/32) address that is advertised within OSPF instance 1 only.
- e. Configuring the VPRN that uses the OSPF instance 1 to use the separate loopback system address as the source addresses for the MDTs.
- f. Ensuring the unicast route that represents the c-source address is advertised as a VPN-IPv4 route and has a BGP connector attribute that contains an address that matches the MDT source address of the originating PE.

Configuration

The following configuration tasks must be completed as a prerequisite:

- Full mesh OSPF base instance between each of the nodes. Note that IS-IS could also be used for any or all of the IGP instances.
- Link-layer LDP between each P and PE router.
- PIM enabled on each router network interface.

Global BGP Configuration

The first step is to configure an iBGP session between each of the PEs and the route reflector (P-5) shown in [Figure 213](#). The address families negotiated between the iBGP peers are **vpn-ipv4**, for unicast routing, and **mdt-safi** for multicast routing.

The iBGP configuration for PE-1 is the following.

```
*A:PE-1# configure router
      bgp
        group "INTERNAL"
          family vpn-ipv4 mdt-safi
          type internal
          neighbor 192.0.2.5
          exit
        exit
      no shutdown
```

The configuration for the other PE nodes is the same.

P-5 is the route reflector for PE-1, PE-2, and PE-3.

```
*A:P-5# configure router
      bgp
        cluster 0.0.0.1
        group "RR_CLIENTS"
          family vpn-ipv4 mdt-safi
          type internal
          neighbor 192.0.2.1
          exit
          neighbor 192.0.2.2
          exit
          neighbor 192.0.2.3
          exit
        exit
      no shutdown
```

Global BGP Configuration

On PE-1, verify that the BGP session with the route reflector at P-5 is established with address families **mdt-safi** and **vpn-ipv4** capabilities negotiated:

```
*A:PE-1# show router bgp summary
```

```
=====
BGP Router ID:192.0.2.1          AS:64496          Local AS:64496
=====
BGP Admin State      : Up          BGP Oper State      : Up
Total Peer Groups    : 1           Total Peers          : 1
Total BGP Paths       : 15         Total Path Memory     : 2760
Total IPv4 Remote Rts : 0           Total IPv4 Rem. Active Rts : 0
Total McIPv4 Remote Rts : 0         Total McIPv4 Rem. Active Rts : 0
Total McIPv6 Remote Rts : 0         Total McIPv6 Rem. Active Rts : 0
Total IPv6 Remote Rts : 0           Total IPv6 Rem. Active Rts : 0
Total IPv4 Backup Rts : 0           Total IPv6 Backup Rts   : 0

Total Supressed Rts   : 0           Total Hist. Rts       : 0
Total Decay Rts       : 0

Total VPN Peer Groups : 0           Total VPN Peers        : 0
Total VPN Local Rts   : 0           Total VPN-IPv4 Rem. Act. Rts : 0
Total VPN-IPv4 Rem. Rts : 0         Total VPN-IPv6 Rem. Act. Rts : 0
Total VPN-IPv6 Rem. Rts : 0         Total VPN-IPv4 Bkup Rts   : 0
Total VPN-IPv4 Bkup Rts : 0         Total VPN-IPv6 Bkup Rts   : 0

Total VPN Supp. Rts   : 0           Total VPN Hist. Rts    : 0
Total VPN Decay Rts   : 0

Total L2-VPN Rem. Rts : 0           Total L2VPN Rem. Act. Rts : 0
Total MVPN-IPv4 Rem Rts : 0         Total MVPN-IPv4 Rem Act Rts : 0
Total MDT-SAFI Rem Rts : 0           Total MDT-SAFI Rem Act Rts : 0
Total MSPW Rem Rts     : 0           Total MSPW Rem Act Rts    : 0
Total RouteTgt Rem Rts : 0           Total RouteTgt Rem Act Rts : 0
Total McVpnIPv4 Rem Rts : 0         Total McVpnIPv4 Rem Act Rts : 0
Total MVPN-IPv6 Rem Rts : 0         Total MVPN-IPv6 Rem Act Rts : 0
Total EVPN Rem Rts     : 0           Total EVPN Rem Act Rts    : 0
Total FlowIpv4 Rem Rts : 0           Total FlowIpv4 Rem Act Rts : 0
Total FlowIpv6 Rem Rts : 0           Total FlowIpv6 Rem Act Rts : 0
=====
BGP Summary
=====
Neighbor
Description
AS PktRcvd InQ Up/Down State|Rcv/Act/Sent (Addr Family)
PktSent OutQ
-----
192.0.2.5
64496 5 0 00h01m00s 0/0/0 (VpnIPv4)
5 0 0/0/0 (MdtSafi)
-----
*A:PE-1#
```

Configuring VPRN on PEs

There are two VPRNs:

- VPRN 1 using the base instance OSPF topology. This is present on PE-1, PE-2, and PE-3.
- VPRN 2 using OSPF instance 1. This is present on PE-1 and PE-2.

The following output displays the configuration for VPRN 1 for the sender PE-1.

```
*A:PE-1# configure service
      vprn 1 customer 1 create
        route-distinguisher 64496:1
        auto-bind-tunnel
          resolution-filter
            ldp
          exit
        resolution filter
      exit
      vrf-target target:64496:1
      interface "int-PE-1-S-1" create
        address 172.16.11.1/24
        sap 1/1/3 create
      exit
    exit
    pim
      apply-to all
      no shutdown
    exit
  mvpn
    auto-discovery mdt-safi
    provider-tunnel
      inclusive
      pim ssm 239.160.1.1
      exit
    exit
  exit
  vrf-target unicast
  exit
exit
no shutdown
```

There is a single interface toward S-1, from which the multicast group is received.

PIM is enabled and applied to all interfaces.

The MVPN configuration enables BGP MDT-SAFI as the auto-discovery mechanism. The provider tunnels between the PEs within the MVPN are PIM SSM Multicast Data Trees with a group address of 239.160.1.1.

Configuring VPRN on PEs

The configuration for VPRN 1 for the receiver PE-2 is the following.

```
*A:PE-2# configure service
      vprn 1 customer 1 create
        route-distinguisher 64496:1
        auto-bind-tunnel
        resolution-filter
          ldp
        exit
        resolution filter
      exit
      vrf-target target:64496:1
      interface "int-PE-2-H-1" create
        address 172.16.21.1/24
        sap 1/1/3 create
      exit
    exit
    igmp
      interface "int-PE-2-H-1"
        no shutdown
      exit
      no shutdown
    exit
    pim
      apply-to all
      no shutdown
    exit
    mvpn
      auto-discovery mdt-safi
      provider-tunnel
        inclusive
        pim ssm 239.160.1.1
      exit
      exit
      vrf-target unicast
      exit
    exit
    no shutdown
```

The configuration for VPRN 1 for receiver PE-3 is as follows.

```
*A:PE-3# configure service
      vprn 1 customer 1 create
        route-distinguisher 64496:1
        auto-bind-tunnel
        resolution-filter
          ldp
        exit
        resolution filter
      exit
      vrf-target target:64496:1
      interface "int-PE-3-H-3" create
        address 172.16.33.1/24
        sap 1/1/3 create
      exit
    exit
```

```

igmp
    interface "int-PE-3-H-3"
        no shutdown
    exit
    no shutdown
exit
pim
    apply-to all
    no shutdown
exit
mvpn
    auto-discovery mdt-safi
    provider-tunnel
        inclusive
        pim ssm 239.160.1.1
    exit
    exit
    exit
    vrf-target unicast
    exit
exit
no shutdown

```

At PE-2, the MDT SAFI NLRI advertised by PE-1 is shown.

```

A:PE-2# show router bgp routes mdt-safi grp-address 239.160.1.1 source-ip 192.0.2.1 detail
=====
BGP Router ID:192.0.2.2      AS:64496      Local AS:64496
=====
Legend -
Status codes  : u - used, s - suppressed, h - history, d - decayed, * - valid
                l - leaked, x - stale, > - best, b - backup, p - purge
Origin codes  : i - IGP, e - EGP, ? - incomplete
=====
BGP MDT-SAFI Routes
=====
Original Attributes

Route Dist.    : 64496:1
Source Addr    : 192.0.2.1
Group Addr     : 239.160.1.1
Nexthop        : 192.0.2.1
From           : 192.0.2.5
Res. Nexthop   : 0.0.0.0
Local Pref.    : 100
Aggregator AS  : None
Atomic Aggr.   : Not Atomic
AIGP Metric    : None
Connector      : None
Community      : target:64496:1
Cluster        : 0.0.0.1
Originator Id  : 192.0.2.1
Flags          : Used Valid Best IGP
Route Source   : Internal
AS-Path        : No As-Path
Route Tag      : 0
Neighbor-AS    : N/A
Orig Validation: N/A

Interface Name : NotAvailable
Aggregator     : None
MED            : 0
Peer Router Id : 192.0.2.5

```

Configuring VPRN on PEs

```
Source Class   : 0                      Dest Class    : 0
Add Paths Send : Default
Last Modified  : 00h17m09s
```

```
---snip---
```

```
Routes : 1
```

```
*A:PE-2#
```

The source and group address is used by PE-2 (and PE-3) to join the MDT that has its root at PE-1. The source address used is the system address of PE-1.

Examining the MDTs for this VPRN at PE-1 shows the state.

```
*A:PE-1# show router pim group 239.160.1.1
```

```
Legend:  A = Active   S = Standby
```

```
PIM Groups ipv4
```

Group Address Source Address	Type RP	Spt Bit State	Inc Intf Inc Intf (S)	No.Oifs
239.160.1.1 192.0.2.1	(S,G)	spt	system	2
239.160.1.1 192.0.2.2	(S,G)	spt	int-PE-1-P-5	1
239.160.1.1 192.0.2.3	(S,G)	spt	int-PE-1-P-5	1

```
Groups : 3
```

```
*A:PE-1#
```

The MDT with the root of its tree at PE-1 is shown.

```
*A:PE-1# show router pim group 239.160.1.1 detail source 192.0.2.1
```

```
PIM Source Group ipv4
```

```
Group Address      : 239.160.1.1
Source Address     : 192.0.2.1
RP Address         : 0
Advt Router        : 192.0.2.1
Flags              : spt                      Type           : (S,G)
MRIB Next Hop      :
MRIB Src Flags     : self
Keepalive Timer Exp: 0d 00:03:05
Up Time            : 0d 00:10:28              Resolved By        : rtable-u

Up JP State        : Joined                    Up JP Expiry       : 0d 00:00:31
Up JP Rpt          : Not Joined StarG         Up JP Rpt Override : 0d 00:00:00
```



```

Register State      : No Info
Reg From Anycast RP: No

Rpf Neighbor       :
Incoming Intf      : system
Outgoing Intf List : system, int-PE-1-P-5

Curr Fwding Rate   : 6583.0 kbps
Forwarded Packets   : 1396587          Discarded Packets : 0
Forwarded Octets    : 64243802         RPF Mismatches     : 0
Spt threshold       : 0 kbps           ECMP opt threshold : 7
Admin bandwidth     : 1 kbps
-----
Groups : 1
=====
*A:PE-1#

```

Note that the source address of the tree is the system address of the router, which is determined from the MDT SAFI NLRI that is advertised to all other PEs via the route reflector. Also, the outgoing interface list contains an interface (int-PE-1-P-5) that is OSPF enabled, and advertised within the base OSPF instance.

From the MDT on PE-2, which has its root on PE-1, the incoming interface is an OSPF interface advertised in the base OSPF instance, as shown.

```

*A:PE-2# show router pim group 239.160.1.1 detail source 192.0.2.1
=====
PIM Source Group ipv4
=====
Group Address      : 239.160.1.1
Source Address     : 192.0.2.1
RP Address         : 0
Advt Router        : 192.0.2.1
Flags              : spt                      Type              : (S,G)
MRIB Next Hop      : 192.168.28.2
MRIB Src Flags     : remote
Keepalive Timer Exp: 0d 00:03:01
Up Time            : 0d 00:09:53          Resolved By           : rtable-u

Up JP State        : Joined                  Up JP Expiry          : 0d 00:00:06
Up JP Rpt          : Not Joined StarG       Up JP Rpt Override    : 0d 00:00:00

Register State     : No Info
Reg From Anycast RP: No

Rpf Neighbor       : 192.168.28.2
Incoming Intf      : int-PE-2-P-8
Outgoing Intf List : system

Curr Fwding Rate   : 5990.6 kbps
Forwarded Packets   : 1356531          Discarded Packets : 0
Forwarded Octets    : 94957338         RPF Mismatches     : 0
Spt threshold       : 0 kbps           ECMP opt threshold : 7
Admin bandwidth     : 1 kbps
-----
Groups : 1

```

Configuring VPRN on PEs

```
=====
*A:PE-2#
```

The incoming interface shown is “int-PE-2-P-8”. Similarly for PE-3, the incoming interface is “int-PE-3-P-9”.

```
*A:PE-3# show router pim group 239.160.1.1 detail source 192.0.2.1
```

```
=====
PIM Source Group ipv4
=====
```

```
Group Address      : 239.160.1.1
Source Address     : 192.0.2.1
RP Address         : 0
Advt Router       : 192.0.2.1
Flags              : spt                      Type              : (S,G)
MRIB Next Hop     : 192.168.39.2
MRIB Src Flags    : remote
Keepalive Timer Exp: 0d 00:03:27
Up Time           : 0d 00:09:56              Resolved By          : rtable-u

Up JP State        : Joined                  Up JP Expiry         : 0d 00:00:03
Up JP Rpt          : Not Joined StarG       Up JP Rpt Override  : 0d 00:00:00

Register State     : No Info
Reg From Anycast RP: No

Rpf Neighbor       : 192.168.39.2
Incoming Intf    : int-PE-3-P-9
Outgoing Intf List : system
```

```
Curr Fwding Rate   : 10084.8 kbps
Forwarded Packets  : 1330121                Discarded Packets   : 0
Forwarded Octets   : 93108646               RPF Mismatches      : 0
Spt threshold      : 0 kbps                  ECMP opt threshold : 7
Admin bandwidth    : 1 kbps
```

```
-----
Groups : 1
=====
```

```
*A:PE-3#
```

VPRN Using Non-Default IGP Instance

A VPRN instance is configured on each of PE-1 and PE-2 that uses an MDT topology governed by the non-default OSPF instance.

Additional interfaces need to be configured.

```
*A:PE-1# configure router
      interface "int-PE-1-P-6a"
          address 192.168.116.1/30
          port 1/1/4
      exit
      interface "loop-1"
          address 192.0.3.1/32
          loopback
      exit
```

There are parallel links between PE-1 and P-6. The interface name of the second link contains the suffix **a**.

In a Rosen Multicast VPN, each PE constructs a default MDT to all other PEs in the multicast VPN domain, as defined by the MDT SAFI BGP update received. The MDT update contains the source address of the MDT to which each PE should join.

When each of the other PEs receives the MDT SAFI Network Layer Reachability Information (NLRI), a PIM join is sent to the source address within the global PIM routing instance to create the MDT.

The MDT source address is usually the system address. Because the system address is advertised in the base instance of OSPF, another /32 address must be used as the source address for the default MDT. Therefore, a second loopback address is configured and used as the source address for the default MDT. For PE-1, this interface is called loop-1, and it is advertised in OSPF 1.

The interface loop-1 will be used as the source address for the MDTs and the next hop for the unicast route representing the source address of the c-multicast group.

The non-default OSPF instance for PE-1 is configured as follows, where 192.0.3.1 is the OSPF 1 router-ID. The router ID need not be equal to the IP address for loop-1, but in this case it is.

```
*A:PE-1# configure router
      ospf 1 192.0.3.1
          area 0.0.0.0
              interface "int-PE-1-P-6a"
                  interface-type point-to-point
              exit
          interface "loop-1"
              exit
      exit
```

LDP is also required for BGP next-hop resolution and is configured as follows for PE-1.

```
*A:PE-1# configure router
  ldp
    interface-parameters
      interface int-PE-1-P-6a
        ipv4
          transport-address interface
        exit
      exit
    exit
```

Note that the transport address is set to interface, rather than the default of system address; this is because the system address is not reachable within OSPF instance 1.

For completeness, the configuration of the additional interfaces, OSPF instance 1 and LDP of PE-2 is shown in the following three outputs.

```
*A:PE-2# configure router
  interface "int-PE-2-P-7a"
    address 192.168.127.1/30
    port 1/1/4
  exit
  interface "loop-1"
    address 192.0.3.2/32
    loopback
  exit
```

The following output displays the OSPF 1 instance configuration.

```
*A:PE-2# configure router
  ospf 1 192.0.3.2
    area 0.0.0.0
      interface "int-PE-2-P-7a"
        interface-type point-to-point
      exit
      interface "loop-1"
        exit
    exit
  no shutdown
exit
```

The following output displays the LDP configuration.

```
*A:PE-2# configure router
  ldp
    interface-parameters
      interface "int-PE-2-P-7a"
        ipv4
          transport-address interface
        exit
      exit
    exit
```

PIM needs to be enabled on all interfaces.

The MDT source address for VPRN 2 is the loop-1 address. Each PE within this VPRN has to join the MDT sourced at PE-1, so the MDT SAFI NLRI must advertise the source address of the MDT group as loop-1. This is achieved by specifying the MDT SAFI source address within the MVPN context. The following output displays the VPRN configuration for PE-1.

```
*A:PE-1# configure service
      vprn 2 customer 1 create
        route-distinguisher 64496:2
        auto-bind-tunnel
        resolution-filter
          ldp
        exit
        resolution filter
      exit
      vrf-target target:64496:2
      interface "int-PE-1-S-2" create
        address 172.16.12.1/24
        sap 1/2/1 create
      exit
    exit
    pim
      apply-to all
      no shutdown
    exit
    mvpn
      auto-discovery mdt-safi source-address 192.0.3.1
      provider-tunnel
        inclusive
        pim ssm 239.160.2.1
      exit
    exit
    vrf-target target:64496:2
  exit
exit
no shutdown
```

Note that the MDT SAFI source address modification is only required on PEs that use the non-default /32 addresses. The system address must not be explicitly configured as the MDT source address for MVPNs that use the default IGP instance. As previously stated, only three MVPNs can be used to create core diversity, one of which must be the default instance. Configuring the system address as a source address prevents the creation of a third MVPN because only two MVPNs are allowed to use explicitly configured MDT source addresses.

Verification of Core Diversity

The MDT SAFI NLRI advertised by the PE-1 sender router is shown.

```
*A:PE-1# show router bgp routes mdt-safi hunt rd 64496:2 | match "RIB Out" post-lines 20
pre-lines 1
```

```
-----
RIB Out Entries
-----
```

```
Route Dist.      : 64496:2
Source Addr      : 192.0.3.1
Group Addr       : 239.160.2.1
Nexthop          : 192.0.2.1
To               : 192.0.2.5
Res. Nexthop     : n/a
Local Pref.      : 100
Aggregator AS    : None
Atomic Aggr.     : Not Atomic
AIGP Metric      : None
Connector        : None
Community        : target:64496:2
Cluster          : No Cluster Members
Originator Id    : None
Origin           : IGP
AS-Path          : No As-Path
Route Tag        : 0
Neighbor-AS      : N/A
Orig Validation  : N/A
Interface Name   : NotAvailable
Aggregator       : None
MED              : 0
Peer Router Id   : 192.0.2.5
-----
```

Note that the source address is set to 192.0.3.1, which is the address of the loopback address used in the non-default OSPF instance 1 of PE-1.

The following output shows the MDT that has its root at PE-1, and that the source address is set to 192.0.3.1. The outgoing interface list includes the router interface contained within the OSPF 1 instance, proving that the non-default OSPF instance is used.

```
*A:PE-1# show router pim group 239.160.2.1 source 192.0.3.1 detail
```

```
=====
PIM Source Group ipv4
=====
```

```
Group Address      : 239.160.2.1
Source Address     : 192.0.3.1
RP Address         : 0
Advt Router        : 192.0.2.1
Flags              : spt
MRIB Next Hop      :
MRIB Src Flags     : self
Keepalive Timer Exp: 0d 00:03:06
Up Time            : 0d 00:06:27
Type               : (S,G)
Resolved By        : rtable-u

Up JP State        : Joined
Up JP Rpt          : Not Joined StarG
Up JP Expiry       : 0d 00:00:32
Up JP Rpt Override : 0d 00:00:00

Register State     : No Info
-----
```

Reg From Anycast RP: No

Rpf Neighbor :
Incoming Intf : loop-1
Outgoing Intf List : system, int-PE-1-P-6a

Curr Fwding Rate : 0.0 kbps
Forwarded Packets : 16 Discarded Packets : 0
Forwarded Octets : 1248 RPF Mismatches : 0
Spt threshold : 0 kbps ECMP opt threshold : 7
Admin bandwidth : 1 kbps

Groups : 1

=====

The PIM interfaces within VPRN 2 are now present on PE-1.

*A:PE-1# show router 2 pim interface

=====

PIM Interfaces ipv4

Interface	Adm	Opr	DR Prty	Hello Intvl	Mcast Send
DR					
int-PE-1-S-2	Up	Up	1	30	auto
172.16.12.1					
2-mt-239.160.2.1	Up	Up	1	N/A	auto
192.0.3.1					

Interfaces : 2 Tunnel-Interfaces : 0

=====

*A:PE-1#

Likewise, for PE-2, the PIM interfaces within VPRN 2 are displayed.

*A:PE-2# show router 2 pim interface

=====

PIM Interfaces ipv4

Interface	Adm	Opr	DR Prty	Hello Intvl	Mcast Send
DR					
int-PE-2-H-2	Up	Up	1	30	auto
172.16.22.1					
2-mt-239.160.2.1	Up	Up	1	N/A	auto
192.0.3.1					

Interfaces : 2 Tunnel-Interfaces : 0

=====

*A:PE-2#

Within the VPRN, there are PIM neighbors shown via the MDT. On PE-2, the PIM neighbor is 192.0.3.1.

```
*A:PE-2# show router 2 pim neighbor
=====
PIM Neighbor ipv4
=====
Interface          Nbr DR Prty    Up Time      Expiry Time   Hold Time
Nbr Address
-----
2-mt-239.160.2.1   1             0d 00:06:29   0d 00:01:36   105
192.0.3.1
-----
Neighbors : 1
=====
*A:PE-2#
```

The PIM interface on PE-2 designated as 2-mt-239.160.2.1 with a neighbor address of 192.0.3.1 is the MDT interface toward PE-1.

Note that the prefix that represents the source address on PE-1 is advertised as a VPN-IPv4 route, which contains a BGP connector attribute.

This can be shown when the VPN-IPv4 route is examined on PE-2.

```
*A:PE-2# show router bgp routes vpn-ipv4 172.16.12.0/24 hunt
=====
BGP Router ID:192.0.2.2      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked, x - stale, > - best, b - backup, p - purge
Origin codes : i - IGP, e - EGP, ? - incomplete
=====
BGP VPN-IPv4 Routes
=====
RIB In Entries
-----
Network       : 172.16.12.0/24
Nextthop      : 192.0.2.1
Route Dist.   : 64496:2      VPN Label     : 262141
Path Id       : None
From          : 192.0.2.5
Res. Nextthop : n/a
Local Pref.   : 100
Aggregator AS : None        Interface Name : int-PE-2-P-8
Atomic Aggr.  : Not Atomic  Aggregator    : None
AIGP Metric   : None       MED           : None
Connector     : RD 64496:2, Originator 192.0.3.1
Community     : target:64496:2
Cluster       : 0.0.0.1
Originator Id : 192.0.2.1     Peer Router Id : 192.0.2.5
Fwd Class     : None        Priority       : None
Flags         : Used Valid Best IGP
Route Source  : Internal
```



```

AS-Path      : No As-Path
Route Tag    : 0
Neighbor-AS  : N/A
Orig Validation: N/A
Source Class : 0                      Dest Class : 0
Add Paths Send : Default
Last Modified : 00h00m26s
VPRN Imported : 2

```

```

-----
RIB Out Entries
-----

```

```

Routes : 1

```

The originator value within the connector attribute is shown to be 192.0.3.1, which is the same as the MDT source address of PE-1. The BGP next hop is still set to the system address of PE-1, so the unicast route can still be resolved via an LDP tunnel.

PIM will now resolve the c-source address RPF using the originator value within the connector attribute.

Similarly, for VPRN 1, the route on PE-1 representing the source address is also advertised as a VPN-IPv4 address that contains a BGP connector attribute.

```

*A:PE-2# show router bgp routes vpn-ipv4 172.16.11.0/24 hunt
=====
BGP Router ID:192.0.2.2      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked, x - stale, > - best, b - backup, p - purge
Origin codes : i - IGP, e - EGP, ? - incomplete
=====
BGP VPN-IPv4 Routes
=====
RIB In Entries
-----
Network      : 172.16.11.0/24
Nexthop      : 192.0.2.1
Route Dist.  : 64496:1      VPN Label      : 262134
Path Id      : None
From         : 192.0.2.5
Res. Nexthop : n/a
Local Pref.  : 100
Aggregator AS : None      Interface Name : int-PE-2-P-8
Atomic Aggr. : Not Atomic Aggregator      : None
AIGP Metric  : None      MED              : None
Connector    : RD 64496:1, Originator 192.0.2.1
Community    : target:64496:1
Cluster      : 0.0.0.1
Originator Id : 192.0.2.1      Peer Router Id : 192.0.2.5
Fwd Class    : None      Priority        : None
Flags        : Used Valid Best IGP
Route Source  : Internal

```

Verification of Core Diversity

```
AS-Path      : No As-Path
Route Tag    : 0
Neighbor-AS  : N/A
Orig Validation: N/A
Source Class : 0
Add Paths Send : Default
Last Modified : 00h15m52s
VPRN Imported : 1
Dest Class   : 0
```

```
-----
RIB Out Entries
-----
-----
```

```
Routes : 1
```

```
=====
*A:PE-2#
```

Verification of Multicast Traffic

An IGMPv3 query is initiated from all 3 hosts: H-1, H-2, and H-3 in Figure 1, and the multicast streams from S-1 and S-2 into interfaces on the two VPRNs are enabled.

Consider VPRN 1, which uses the default topology. On PE-1, the group 239.160.1.123 can be shown. The outgoing and incoming interface lists are populated, with the outgoing interface being the MDT interface for the VPRN:

```
*A:PE-1# show router 1 pim group detail
=====
PIM Source Group ipv4
=====
Group Address       : 239.160.1.123
Source Address      : 172.16.11.2
RP Address          : 0
Advt Router         : 192.0.2.1
Flags               :                               Type           : (S,G)
MRIB Next Hop       : 172.16.11.2
MRIB Src Flags      : direct
Keepalive Timer     : Not Running
Up Time             : 0d 00:02:15      Resolved By           : rtable-u

Up JP State         : Joined           Up JP Expiry          : 0d 00:00:00
Up JP Rpt           : Not Joined StarG Up JP Rpt Override    : 0d 00:00:00

Register State      : No Info
Reg From Anycast RP: No

Rpf Neighbor        : 172.16.11.2
Incoming Intf      : int-PE-1-S-1
Outgoing Intf List : 1-mt-239.160.1.1

Curr Fwding Rate    : 7134.2 kbps
Forwarded Packets   : 16467196          Discarded Packets     : 0
Forwarded Octets    : 757491016         RPF Mismatches        : 0
Spt threshold       : 0 kbps             ECMP opt threshold    : 7
Admin bandwidth     : 1 kbps

-----
Groups : 1
=====
*A:PE-1#
```

The same groups can be shown within VPRN 1 on PE-2.

```
*A:PE-2# show router 1 pim group detail
=====
PIM Source Group ipv4
=====
Group Address       : 239.160.1.123
Source Address      : 172.16.11.2
RP Address          : 0
Advt Router         : 192.0.2.1
Flags               :                               Type           : (S,G)
```

Verification of Multicast Traffic

```
MRIB Next Hop      : 192.0.2.1
MRIB Src Flags     : remote
Keepalive Timer    : Not Running
Up Time            : 0d 00:14:49      Resolved By       : rtable-u

Up JP State        : Joined           Up JP Expiry       : 0d 00:00:58
Up JP Rpt          : Not Joined StarG Up JP Rpt Override : 0d 00:00:00

Register State     : No Info
Reg From Anycast RP: No

Rpf Neighbor       : 192.0.2.1
Incoming Intf    : 1-mt-239.160.1.1
Outgoing Intf List : int-PE-2-H-1

Curr Fwding Rate   : 5764.4 kbps
Forwarded Packets  : 14046700         Discarded Packets  : 0
Forwarded Octets   : 646148200        RPF Mismatches     : 0
Spt threshold      : 0 kbps           ECMP opt threshold : 7
Admin bandwidth    : 1 kbps

-----
Groups : 1
=====
*A:PE-2#
```

The MDT is now the incoming interface with an upstream RPF neighbor of 192.0.2.1, the system address of PE-1. Similarly for PE-3:

```
*A:PE-3# show router1 pim group detail
=====
PIM Source Group ipv4
=====
Group Address      : 239.160.1.123
Source Address     : 172.16.11.2
RP Address         : 0
Advt Router        : 192.0.2.1
Flags              :                               Type       : (S,G)
MRIB Next Hop      : 192.0.2.1
MRIB Src Flags     : remote
Keepalive Timer    : Not Running
Up Time            : 0d 00:14:18      Resolved By       : rtable-u

Up JP State        : Joined           Up JP Expiry       : 0d 00:00:36
Up JP Rpt          : Not Joined StarG Up JP Rpt Override : 0d 00:00:00

Register State     : No Info
Reg From Anycast RP: No

Rpf Neighbor       : 192.0.2.1
Incoming Intf    : 1-mt-239.160.1.1
Outgoing Intf List : int-PE-3-H-3

Curr Fwding Rate   : 7233.6 kbps
Forwarded Packets  : 4188552         Discarded Packets  : 0
Forwarded Octets   : 192673392        RPF Mismatches     : 0
Spt threshold      : 0 kbps           ECMP opt threshold : 7
Admin bandwidth    : 1 kbps
```

```
-----
Groups : 1
=====
```

```
*A:PE-3#
```

Consider VPRN 2, which uses the non-default topology. On PE-1, the group 239.160.2.123 can be shown. The outgoing and incoming interface lists are populated, with the outgoing interface being the MDT interface for the VPRN.

```
*A:PE-1# show router 2 pim group detail
```

```
=====
PIM Source Group ipv4
=====
```

```
Group Address      : 239.160.2.123
Source Address     : 172.16.12.2
RP Address         : 0
Advt Router        : 192.0.2.1
Flags              :
Type               : (S,G)
MRIB Next Hop      : 172.16.12.2
MRIB Src Flags     : direct
Keepalive Timer    : Not Running
Up Time            : 0d 00:00:39
Resolved By        : rtable-u

Up JP State        : Joined
Up JP Rpt          : Not Joined StarG
Up JP Expiry       : 0d 00:00:00
Up JP Rpt Override : 0d 00:00:00
```

```
Register State     : No Info
Reg From Anycast RP: No
```

```
Rpf Neighbor       : 172.16.12.2
Incoming Intf      : int-PE-1-S-2
Outgoing Intf List : 2-mt-239.160.2.1
```

```
Curr Fwding Rate   : 11945.5 kbps
Forwarded Packets   : 8110278
Forwarded Octets    : 10900213632
Spt threshold       : 0 kbps
Admin bandwidth     : 1 kbps
Discarded Packets   : 0
RPF Mismatches      : 0
ECMP opt threshold  : 7
```

```
-----
Groups : 1
=====
```

The outgoing interface list is again populated with the MDT being the interface. This MDT is encapsulated in the multicast tree shown in the global PIM context as multicast group 239.160.2.1 with source address 192.0.3.1. This can be shown to have an outgoing interface list containing the interface int-PE-1-P-6a, which is an OSPF 1 interface and was shown in a previous output.

```
*A:PE-1# show router pim group detail 239.160.2.1
```

```
---snip---
```

```
=====
PIM Source Group ipv4
=====
```

```
Group Address      : 239.160.2.1
Source Address     : 192.0.3.1
```

Verification of Multicast Traffic

```
RP Address      : 0
Advt Router     : 192.0.2.1
Flags           : spt                               Type           : (S,G)
MRIB Next Hop   :
MRIB Src Flags  : self
Keepalive Timer Exp: 0d 00:03:18
Up Time         : 0d 03:19:45                       Resolved By        : rtable-u

Up JP State     : Joined                           Up JP Expiry       : 0d 00:00:14
Up JP Rpt      : Not Joined StarG                   Up JP Rpt Override : 0d 00:00:00

Register State  : No Info
Reg From Anycast RP: No

Rpf Neighbor    :
Incoming Intf   : loop-1
Outgoing Intf List : system, int-PE-1-P-6a

Curr Fwding Rate : 0.0 kbps
Forwarded Packets : 403                               Discarded Packets : 0
Forwarded Octets  : 31434                             RPF Mismatches    : 0
Spt threshold     : 0 kbps                             ECMP opt threshold : 7
Admin bandwidth   : 1 kbps
-----
Groups : 2
=====
*A:PE-1#
```

Conclusion

MVPN Core Diversity allows service providers to provide separation in terms of topology between content providers that use a core network to provide transport between source and receivers in a multicast VPN. This chapter provides the configuration for multiple instances of OSPF which, together with the associated commands and outputs, can be used for verifying and troubleshooting.

Multicast VPN: Inter-AS Option B

In This Chapter

This section provides information about MVPN: Inter-AS Option B configurations.

Topics in this section include:

- [Applicability on page 1534](#)
- [Overview on page 1535](#)
- [Configuration on page 1543](#)
- [Conclusion on page 1558](#)

Applicability

This example is applicable to 7950 XRS, 7750 all variants, 7750 SR c4/12, 7450 mixed mode systems. Chassis mode C or D must be used. The configuration was tested on Release 11.0 R3.

Overview

This configuration note covers a basic technology overview, the network topology and configuration examples which are used for Multicast VPN Inter-AS option B.

Knowledge of the Alcatel-Lucent multicast and Layer 3 VPNs concepts are assumed throughout this document.

Overview

The Inter-AS MVPN feature allows the setup of Multicast Distribution Trees that span multiple Autonomous Systems.

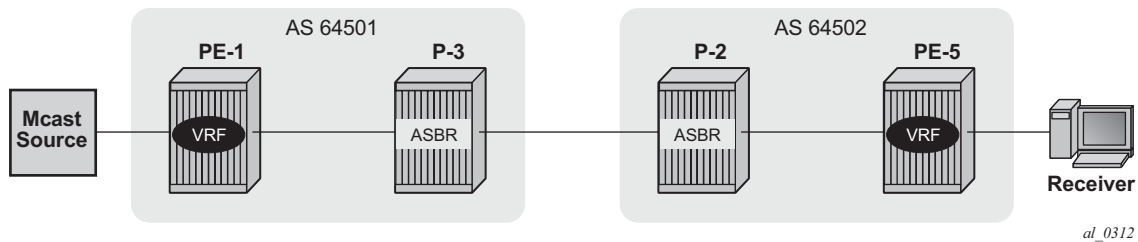


Figure 216: General Topology for Inter-AS MVPN

This example covers Draft-Rosen Inter-AS support (Option-B). Inter-AS Option B is supported for PIM SSM with Draft-Rosen MVPN using Multicast Distribution Tree (MDT) Subsequent Address Family (SAFI), using the BGP Connector attribute and PIM RPF vector.

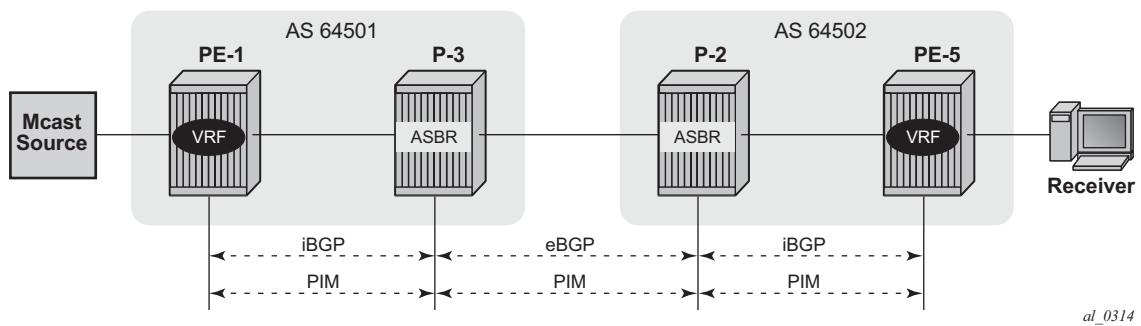


Figure 217: Protocols Used for Inter-AS MVPN

The following assumptions are made:

- PE-1 is named “sender PE” because the multicast source is directly connected to this router.
- PE-5 is named “receiver PE” because multicast receiver is directly connected to this router.

- P-2 and P-3 are named “ASBR” routers according to Inter-AS model.

The multicast receiver and source can be indirectly connected to PE routers via CE routers, but for the core multicast distribution these variations are conceptually the same. For simplicity, the PE and P router configurations will be provided.

There are several challenges which have to be solved in order to make complete inter-as solution operational:

Challenge 1:

In case of Inter-AS MVPN Option B, routing information towards the source PE is not available in a remote AS domain since IGP routes are not exchanged between ASs.

As a result a PIM-P Join would never be sent upstream (from the receiver PE to the sender PE in a different AS). However, the PIM-P join has to be propagated from PE-5 to PE-1. Therefore a solution is required to issue PIM-P Join and perform RPF.

Solution:

Use a PIM RPFV (Reverse Path Forwarding (RPF) vector) to segment the PIM-P propagation. In this example there are three segments:

- PE-5 -> ASBR P-2
- ASBR P-2 -> ASBR P-3
- ASBR P-3 -> PE-1

The RPF vector is added to a PIM join at the PE router when the following option is enabled:

```
*A:PE-5>config>router>pim# rpfv
- no rpfv [mvpn]
- rpfv mvpn
<mvpn> : Proxy RPF vector for inter-AS rosen mvpn
```

mvpn enables “mvpn RPF vector” processing for Inter-AS Option B MVPN based on RFC 5496 and RFC 6513. If a “core RPF” vector is received, it will be dropped before a message is processed.

All routers which are used for multicast traffic transportation must have this option enabled to allow RPF Vector processing. If the option is not enabled, the RPF Vector is dropped and the PIM Join is processed as if the PIM Vector is not present.

Details about RPF Vector can be found in the following RFCs: 5496, 5384, 6513.

Challenge 2:

With Inter-AS MVPN Option B, the BGP next-hop is modified by the local and remote ASBRs during re-advertisement of VPNv4 routes. When the BGP next-hop is changed, information regarding the originator of the prefix is lost when the advertisement reaches the receiver PE node. Therefore a solution is required to do a successful RPF check for the VPN source at receiver VPRN.

Note: This challenge does not apply to Model C since in Model C the BGP next-hop for VPN routes is not updated.

Solution:

A new transitive BGP attribute - Connector - is used to advertise an address of a sender PE node which is carried inside VPNv4 update. The BGP connector attribute allows the sender PE address information to be available to the receiver PE so that a receiver PE is able to associate VPNv4 advertisement to the corresponding source PE.

Inter-AS Option B will work when the following criteria are met:

- Draft-rosen MVPN is used with PIM SSM
- BGP MDT-SAFI address family is used
- PIM RPF Vector is configured
- BGP Connector attribute is used for vpn-ipv4 updates

SR OS inter-AS Option B is designed to be standard compliant based on the following RFCs:

- RFC 5384, *The Protocol Independent Multicast (PIM) Join Attribute Format*
- RFC 5496, *The Reverse Path Forwarding (RPF) Vector TLV*
- RFC 6513, *Multicast in MPLS/BGP IP VPNs*

Three global signaling stages can be identified when Inter-AS MVPN is configured:

Stage 1: BGP core signaling

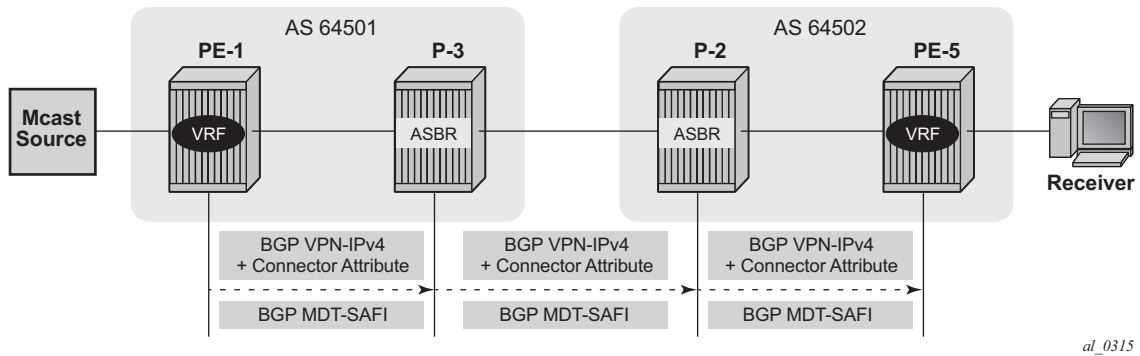


Figure 218: BGP Signaling Steps

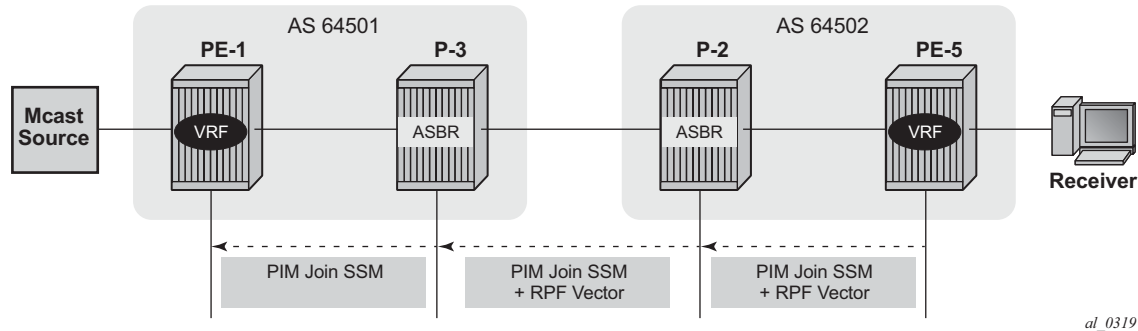
The sender PE sends VPN-IPv4 and MDT-SAFI BGP updates for this particular MVPN:

- Every ASBR propagates VPN-IPv4 and MDT-SAFI BGP updates:
 - Next-Hop (NH) attribute is modified every time
 - Connector attribute stays untouched

When this stage is completed, all routers have necessary information:

- to start PIM signaling in the core network (PIM-P) to prepare the Default MDT
- to start PIM signaling of customer's multicast streams (PIM-C) inside VPN

Stage 2: Core PIM signaling

**Figure 219: PIM-P Signaling Steps for Default MDT**

PE-5 determines the reverse Path to the source based on the RPF Vector (ASBR P-2 IP address) and not based on the IP address of the multicast source (PE-1) which is unknown to it.

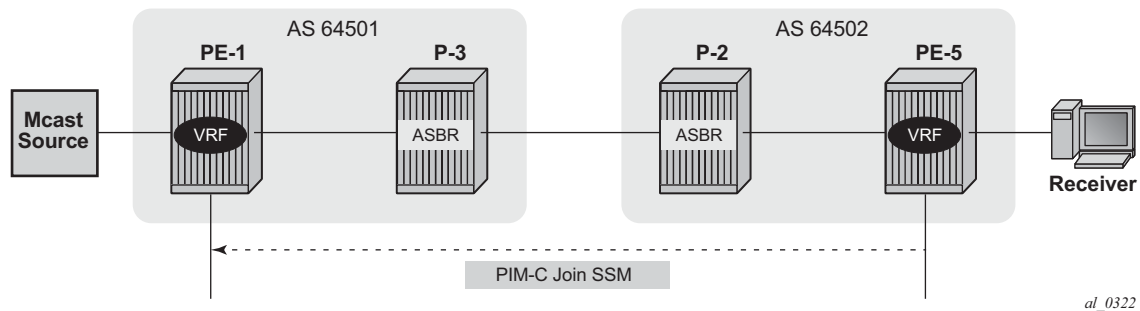
PE-5 inserts an RPF vector and sends a PIM-P Join to the immediate next-hop to reach ASBR P-2. Intermediate P-routers (if present) do not change the RPF vector.

P-2 finds itself in RPF Vector and has to make a decision based on MDT-SAFI BGP table:

- P-2 determines the reverse path to the multicast source based on the RPF Vector (ASBR P-3 IP address).
- If the multicast source and NH do not match, P-2 has to use RPFV.
- P-2 modifies the PIM-P Join received from PE-5 with ASBR P-3's IP address as the upstream (taken from Next-hop MDT-SAFI NLRI).
- P-3 can match the source IP with the NH in BGP MDT-SAFI. Therefore there is no need for RPF Vector to be used.
- P-3 removes the RPF vector and sends a normal PIM-P join towards PE-1.

When this stage is completed, the default MDT is established for this MVPN and PE routers have the necessary information to start PIM signaling inside VPRN (PIM-C).

Stage 3: Customer PIM signaling



al_0322

Figure 220: PIM-C Signaling

A PIM-C Join is sent to the source PE using the existing tunnel infrastructure to the RPF neighbor PE-1 provided by the BGP connector attribute of the vpn-ipv4 route of the multicast source.

When this stage is completed, the customer multicast flows throughout the network in a Default MDT.

Stage 4¹: The Multicast stream threshold is reached.

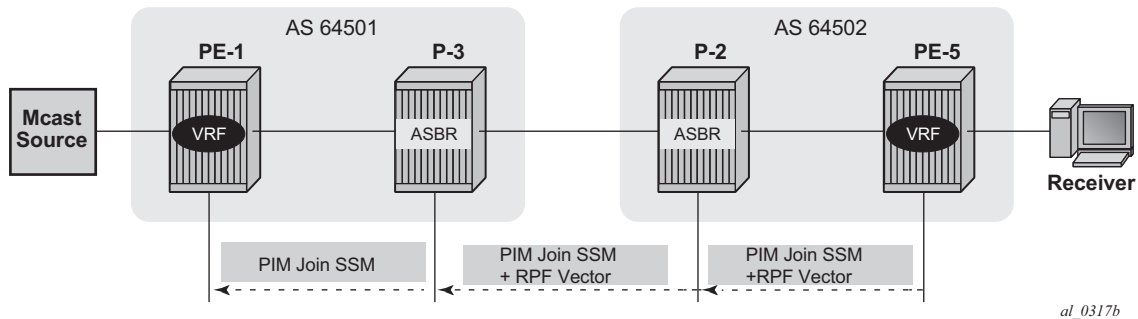


Figure 221: PIM-P Signaling Steps for Data MDT

The process is similar to the Default MDT setup:

- PE-5 determines reverse path to the source based on the RPF Vector (ASBR P-2's IP address) and not based on IP address of the multicast source (PE-1) which is unknown to it.
- PE-5 inserts an RPF vector and sends a PIM-P Join to the immediate next-hop to reach ASBR P-2.
- Intermediate P-routers (if present) do not change RPF vector.
- P-2 finds itself in the RPF Vector and has to make a decision based on the MDT-SAFI BGP table:
 - P-2 determines reverse path to the multicast source based on the RPF Vector (ASBR P-3's IP address).
 - If the multicast source and NH do not match, P-2 has to use the RPFV.
 - P-2 modifies the PIM-P Join received from PE-5 with ASBR P-3's IP address as upstream (taken from Next-hop MDT-SAFI NLRI).
- P-3 can match the source IP with the NH in the BGP MDT-SAFI. Therefore there is no need for RPF Vector to be used.
- P-3 removes the RPF vector and sends a normal PIM-P join towards PE-1.

When this optional stage is completed, the customer multicast flows in a dedicated Data MDT.

1. This stage is optional and applicable when S-PMSI instance and S-PMSI threshold are configured.

Known interoperability issues:

The SR OS implementation was also designed to interoperate with Cisco routers' Inter-AS implementations that do not fully comply with the RFC 5384 and RFC5496.

When the following option is enabled:

```
configure router pim rpfv mvpn
```

Cisco routers need to be configured to include **RD** in an RPF vector using the following command for interoperability:

```
ip multicast vrf <name> rpf proxy rd vector
```

Configuration

The test topology is shown in [Figure 222](#).

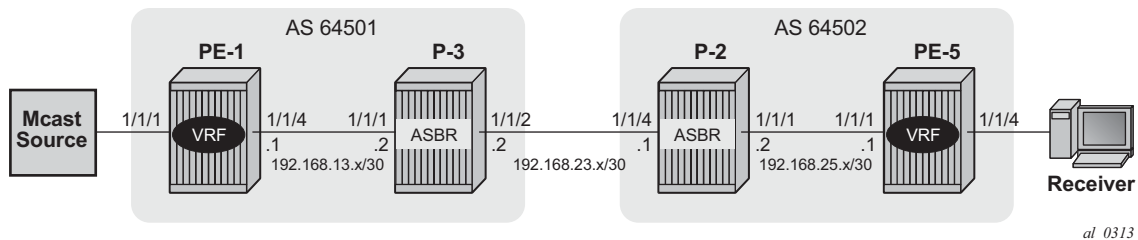


Figure 222: Test Topology Details

The following parameters are used in the test scenario:

- VPRN 1 is used
- Customer multicast group is 232.0.0.0/8
- Default MDT multicast group is 239.255.0.1
- Data MDT multicast group is 239.255.1.0/24
- Multicast source is 172.16.1.1
- PE-x routers have system IP addresses 192.0.2.x
- P-x routers have system IP addresses 192.0.2.x
- Interface between Router A and B has IP address 192.168.AB.x

Global BGP configuration for PE-1 router using the family mdt-safi with an iBGP neighbor to P-3. System interface IP address is used for iBGP session.

```
configure
router
  bgp
    group "iBGP"
    family vpn-ipv4 mdt-safi
    type internal
    neighbor 192.0.2.3
    next-hop-self
  exit
exit
```

Global BGP configuration for P-3 router using the family mdt-safi with an iBGP neighbor to PE-1 and an eBGP neighbor to P-2. System interface IP address is used for iBGP session and network interface IP address is used for eBGP session.

Configuration

```
configure router bgp
  enable-inter-as-vpn
  group "eBGP"
    family vpn-ipv4 mdt-safi
    neighbor 192.168.23.1
      type external
      peer-as 64502
    exit
  exit
  group "iBGP"
    family vpn-ipv4 mdt-safi
    neighbor 192.0.2.1
      next-hop-self
      type internal
    exit
  exit
```

Global BGP configuration for P-2 router using the family mdt-safi with an iBGP neighbor to PE-5 and an eBGP neighbor to P-3. System interface IP address is used for iBGP session and network interface IP address is used for eBGP session.

```
configure router bgp
  enable-inter-as-vpn
  group "eBGP"
    family vpn-ipv4 mdt-safi
    neighbor 192.168.23.2
      type external
      peer-as 64501
    exit
  exit
  group "iBGP"
    family vpn-ipv4 mdt-safi
    neighbor 192.0.2.5
      next-hop-self
      type internal
    exit
  exit
```

Global BGP configuration for PE-5 router using the family mdt-safi with an iBGP neighbor to P-2. System interface IP address is used for iBGP session.

```
configure
  router
    bgp
      group "iBGP"
        family vpn-ipv4 mdt-safi
        type internal
        neighbor 192.0.2.2
          next-hop-self
        exit
      exit
```

Global PIM configuration for ALL routers.

```

configure router pim
  rpf-table both
  apply-to non-ies
  rp
    static
    exit
    bsr-candidate
      shutdown
    exit
    rp-candidate
      shutdown
    exit
  exit
  no shutdown
  rpfv mvpn

```

VPRN configuration for PE routers.

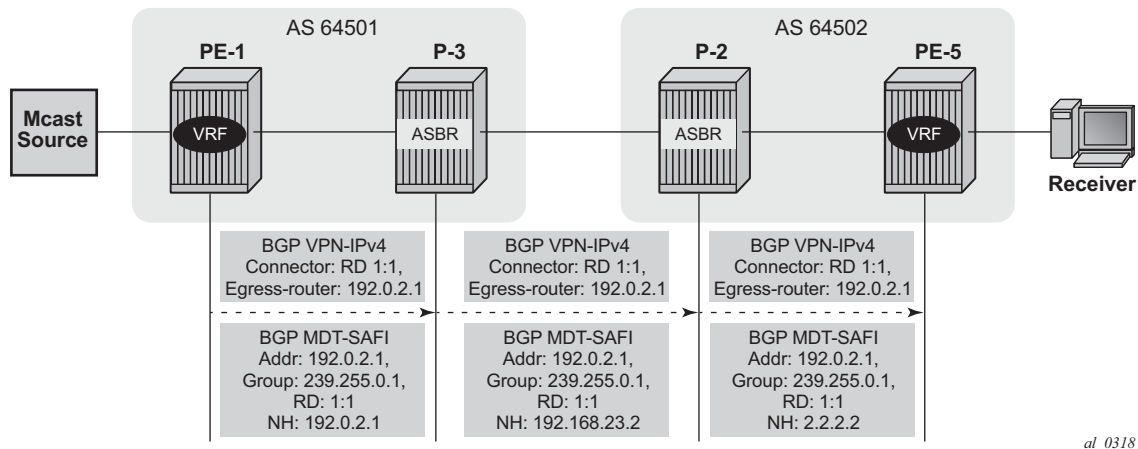
```

PE-x>config>service>vprn# info
-----
<snip>
  mvpn
    auto-discovery mdt-safi
    c-mcast-signaling pim
      inclusive
        pim ssm 239.255.0.1
      exit
    exit
    selective
      data-threshold 232.0.0.0/8 1
      pim-ssm 239.255.1.0/24
    exit
  exit
  vrf-target unicast
  exit
exit

```

MVPN Verification and Debugging

BGP Core Signaling



al_0318

Figure 223: BGP Signaling Steps

On PE-1, the **debug router bgp update** output shows the BGP update messages which are sent to P-3. The VPN-IPv4 update contains a connector attribute and the MDT-SAFI update is used for signaling multicast group 239.255.0.1.

```
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 95
  Flag: 0x90 Type: 14 Len: 49 Multiprotocol Reachable NLRI:
    Address Family VPN_IPV4
    NextHop len 12 NextHop 192.0.2.1
    172.16.1.0/30 RD 1:1 Label 262142
    192.0.2.1/32 RD 1:1 Label 262142
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:1:1
  Flag: 0xc0 Type: 20 Len: 14 Connector:
    RD 1:1, Egress-router 192.0.2.1

"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 62
  Flag: 0x90 Type: 14 Len: 26 Multiprotocol Reachable NLRI:
    Address Family MDT-SAFI
    NextHop len 4 NextHop 192.0.2.1
```

```

[MDT-SAFI] Addr 192.0.2.1, Group 239.255.0.1, RD 1:1
Flag: 0x40 Type: 1 Len: 1 Origin: 0
Flag: 0x40 Type: 2 Len: 0 AS Path:
Flag: 0x80 Type: 4 Len: 4 MED: 0
Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
Flag: 0xc0 Type: 16 Len: 8 Extended Community:
target:1:1

```

On P-3, the **debug router bgp update** output shows the BGP update messages which are sent to P-2. The VPN-IPv4 update contains an unmodified connector attribute and the MDT-SAFI update is used for signaling multicast group 239.255.0.1.

```

"Peer 1: 192.168.23.1: UPDATE
Peer 1: 192.168.23.1 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 126
  Flag: 0x90 Type: 14 Len: 81 Multiprotocol Reachable NLRI:
    Address Family VPN_IPV4
    NextHop len 12 NextHop 192.168.23.2
    192.0.2.4/32 RD 1:1 Label 262142
    192.0.2.1/32 RD 1:1 Label 262142
    172.16.1.0/30 RD 1:1 Label 262142
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 6 AS Path:
    Type: 2 Len: 1 < 64501 >
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:1:1
  Flag: 0xc0 Type: 20 Len: 14 Connector:
    RD 1:1, Egress-router 192.0.2.1

```

```

"Peer 1: 192.168.23.1: UPDATE
Peer 1: 192.168.23.1 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 54
  Flag: 0x90 Type: 14 Len: 26 Multiprotocol Reachable NLRI:
    Address Family MDT-SAFI
    NextHop len 4 NextHop 192.168.23.2
    [MDT-SAFI] Addr 192.0.2.1, Group 239.255.0.1, RD 1:1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 6 AS Path:
    Type: 2 Len: 1 < 64501 >
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:1:1

```

On P-2, the **debug router bgp update** output shows the BGP update messages which are sent to PE-5. The VPN-IPv4 update contains an unmodified connector attribute and the MDT-SAFI update is used for signaling multicast group 239.255.0.1.

```

"Peer 1: 192.0.2.5: UPDATE
Peer 1: 192.0.2.5 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 133
  Flag: 0x90 Type: 14 Len: 81 Multiprotocol Reachable NLRI:
    Address Family VPN_IPV4
    NextHop len 12 NextHop 192.0.2.2

```

Configuration

```
192.0.2.4/32 RD 1:1 Label 262142
172.16.1.0/30 RD 1:1 Label 262142
192.0.2.1/32 RD 1:1 Label 262142
Flag: 0x40 Type: 1 Len: 1 Origin: 0
Flag: 0x40 Type: 2 Len: 6 AS Path:
    Type: 2 Len: 1 < 64501 >
Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:1:1
Flag: 0xc0 Type: 20 Len: 14 Connector:
    RD 1:1, Egress-router 192.0.2.1

"Peer 1: 192.0.2.5: UPDATE
Peer 1: 192.0.2.5 - Send BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 61
    Flag: 0x90 Type: 14 Len: 26 Multiprotocol Reachable NLRI:
        Address Family MDT-SAFI
        NextHop len 4 NextHop 192.0.2.2
        [MDT-SAFI] Addr 192.0.2.1, Group 239.255.0.1, RD 1:1
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 6 AS Path:
        Type: 2 Len: 1 < 64501 >
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:1:1
```

The BGP tables on PE1 and PE-5 are updated accordingly. The most interesting aspect here is MDT-SAFI routes received.

PE-5 has one MDT-SAFI update received from PE-1. The next-hop was modified accordingly to Option-B model.

```
*A:PE-5# show router bgp neighbor 192.0.2.2 received-routes mdt-safi
=====
BGP Router ID:192.0.2.5          AS:64502          Local AS:64502
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP MDT-SAFI Routes
=====
Flag  Network                               LocalPref  MED
     Nexthop                               Group-Addr  Label
     As-Path
-----
u*>i  1:1:192.0.2.1                           100        None
      192.0.2.2                             239.255.0.1
      64501
```

PE-1 has one MDT-SAFI update received from PE-5. The next-hop was modified accordingly to Option-B model.


```

*A:PE-1# show router bgp neighbor 192.0.2.4 received-routes mdt-safi
=====
BGP Router ID:192.0.2.1          AS:64501          Local AS:64501
=====
Legend -
Status codes  : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP MDT-SAFI Routes
=====
Flag  Network                               LocalPref  MED
      Nexthop                               Group-Addr  Label
      As-Path
-----
u*>i  1:5:192.0.2.5                           100         None
      192.0.2.4                               239.255.0.1  -
      64502

```

Core PIM Signaling

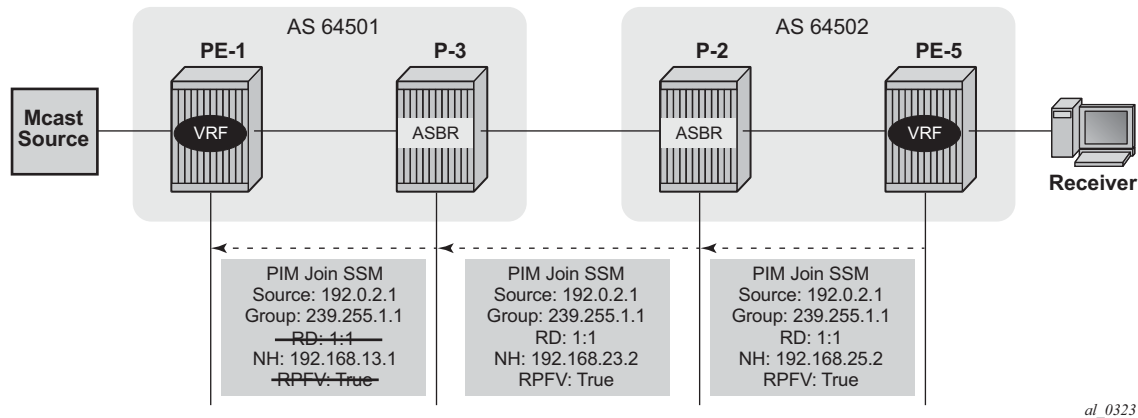


Figure 224: PIM-P Signaling Steps for Default MDT

On PE-5, the **debug router pim packet jp** output shows the PIM join/prune message which is sent to P-2. This message contains the original source of the multicast traffic (PE-1: 192.0.2.1) and the RPF Vector (P-2: 192.0.2.2).

```
"PIM[Instance 1 Base]: Join/Prune
[007 14:55:54.020] PIM-TX ifId 3 ifName int-PE5-P2 -> 224.0.0.13 Length: 48
PIM Version: 2 Msg Type: Join/Prune Checksum: 0x8b5e
Upstream Nbr IP : 192.168.25.2 Resvd: 0x0, Num Groups 1, HoldTime 210
  Group: 239.255.0.1/32 Num Joined Srcs: 1, Num Pruned Srcs: 0
  Join Srcs:
    192.0.2.1/32 Flag S <S,G> JA={rpfvMvnpn 192.0.2.2 1:1}
```

On P-2, the **debug router pim packet jp** output shows the PIM join/prune message which is propagated to P-3. The source of multicast traffic is untouched while the RPF Vector is modified for Inter-AS propagation.

```
"PIM[Instance 1 Base]: Join/Prune
[001 12:25:19.590] PIM-TX ifId 4 ifName int-P2-P3 -> 224.0.0.13 Length: 48
PIM Version: 2 Msg Type: Join/Prune Checksum: 0x835e
Upstream Nbr IP : 192.168.23.2 Resvd: 0x0, Num Groups 1, HoldTime 210
  Group: 239.255.0.1/32 Num Joined Srcs: 1, Num Pruned Srcs: 0
  Join Srcs:
    192.0.2.1/32 Flag S <S,G> JA={rpfvMvnpn 192.168.23.2 1:1}
```

On P-3, the **debug router pim packet jp** output shows the PIM join/prune message which is propagated to P-3. The source of multicast traffic is untouched while the RPF Vector is not present anymore.

```
"PIM[Instance 1 Base]: Join/Prune
```

```
[001 12:25:16.000] PIM-TX ifId 2 ifName int-P3-PE1 -> 224.0.0.13 Length: 34
PIM Version: 2 Msg Type: Join/Prune Checksum: 0xd694
Upstream Nbr IP : 192.168.13.1 Resvd: 0x0, Num Groups 1, HoldTime 210
  Group: 239.255.0.1/32 Num Joined Srcs: 1, Num Pruned Srcs: 0
  Join Srcs:
    192.0.2.1/32 Flag S <S,G>
```

As a result of this signaling, Default MDT is established between the two ASs. This can be checked with **show router pim group** command.

The PE-1 output shows the active multicast groups which are used as Default MDT.

```
*A:PE-1#show router pim group
=====
PIM Groups ipv4
=====
Group Address          Type          Spt Bit Inc Intf          No.Oifs
  Source Address        RP              Inc Intf(S)
-----
239.255.0.1            (S,G)          spt   system                2
  192.0.2.1
239.255.0.1            (S,G)          spt   int-PE1-P3             1
  192.0.2.5
```

The PE-5 output shows active multicast groups which are used as Default MDT:

```
*A:PE-5# show router pim group
=====
PIM Groups ipv4
=====
Group Address          Type          Spt Bit Inc Intf          No.Oifs
  Source Address        RP              Inc Intf(S)
-----
239.255.0.1            (S,G)          spt   int-PE5-P2             1
  192.0.2.1
239.255.0.1            (S,G)          spt   system                  2
  192.0.2.5
```

The detailed information about the PIM-P group shows that the Default MDT is used to deliver traffic. Key parameters such as correct the incoming/outgoing interfaces and non-zero flow rate allow this conclusion to be made.

PE-5 has the incoming interface “int-PE5-P2”, outgoing interface “system” and flow rate of 5.4 kbps.

```
*A:PE-5# show router pim group detail
Group Address      : 239.255.0.1
Source Address     : 192.0.2.1
RP Address         : 0
Advt Router        : 192.0.2.2
Upstream RPFV Nbr  : 192.168.25.2
RPFV Type          : Mvpn 1:1          RPFV Proxy      : 192.0.2.2
Flags              : spt                Type            : (S,G)
```

Configuration

```
MRIB Next Hop      : 192.168.25.2
MRIB Src Flags     : remote
Keepalive Timer Exp: 0d 00:03:00
Up Time           : 0d 04:57:13      Resolved By       : rtable-u

Up JP State        : Joined           Up JP Expiry       : 0d 00:00:47
Up JP Rpt          : Not Joined StarG Up JP Rpt Override : 0d 00:00:00

Register State     : No Info
Reg From Anycast RP: No

Rpf Neighbor       : 192.168.25.2
Incoming Intf      : int-PE5-P2
Outgoing Intf List : system

Curr Fwding Rate   : 5.4 kbps
Forwarded Packets  : 178895           Discarded Packets  : 0
Forwarded Octets   : 11814210        RPF Mismatches     : 0
Spt threshold      : 0 kbps           ECMP opt threshold : 7
Admin bandwidth    : 1 kbps
```

PE-1 has incoming the interface “system”, outgoing interfaces “system, int-PE1-P” and flow rate of 3.4 kbps.

```
A:PE-1# show router pim group detail
Group Address      : 239.255.0.1
Source Address     : 192.0.2.1
RP Address         : 0
Advt Router        : 192.0.2.1
Flags              : spt              Type              : (S,G)
MRIB Next Hop      :
MRIB Src Flags     : self
Keepalive Timer Exp: 0d 00:03:30
Up Time           : 0d 23:02:04      Resolved By       : rtable-m

Up JP State        : Joined           Up JP Expiry       : 0d 00:00:56
Up JP Rpt          : Not Joined StarG Up JP Rpt Override : 0d 00:00:00

Register State     : No Info
Reg From Anycast RP: No

Rpf Neighbor       :
Incoming Intf      : system
Outgoing Intf List : system, int-PE1-P3

Curr Fwding Rate   : 3.4 kbps
Forwarded Packets  : 826316           Discarded Packets  : 0
Forwarded Octets   : 34805244        RPF Mismatches     : 0
Spt threshold      : 0 kbps           ECMP opt threshold : 7
Admin bandwidth    : 1 kbps
```

Customer PIM Signaling

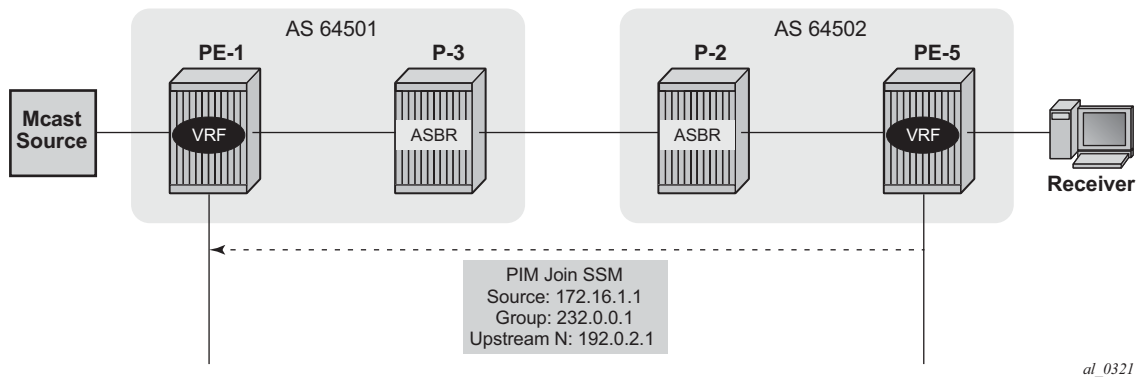


Figure 225: PIM-C Signaling

The PIM-C Join is sent to the sender PE using the existing tunnel infrastructure.

On PE-5, the **debug router 1 pim packet jp** output shows the PIM join/prune message which is sent to PE-1 using PMSI interface “1-mt-239.255.0.1” inside VPRN 1. All of this information and more can be found in the output of the **debug** command.

```
49 2013/05/02 11:23:32.11 UTC MINOR: DEBUG #2001 vprn1 PIM[Instance 9 vprn1]
"PIM[Instance 9 vprn1]: Join/Prune
[006 04:23:58.880] PIM-TX ifId 16390 ifName 1-mt-239.255.0.1 -> 224.0.0.13 Len
gth: 34
PIM Version: 2 Msg Type: Join/Prune Checksum: 0xdbed
Upstream Nbr IP : 192.0.2.1 Resvd: 0x0, Num Groups 1, HoldTime 210
Group: 232.0.0.1/32 Num Joined Srcs: 1, Num Pruned Srcs: 0
Join Srcs:
    172.16.1.1/32 Flag S <S,G>
```

The detailed information about the PIM-C group for a particular VPRN shows that default MDT is used to deliver traffic. For this purpose the **show router 1 pim group detail** command is used. Key parameters such as the correct multicast group, correct incoming/outgoing interfaces and non-zero flow rate allow this conclusion to be made.

PE-1 has the incoming interface “int-source”, outgoing interface “1-mt-239.255.0.1” and flow rate of 3.5 kbps.

```
*A:PE-1#show router 1 pim group detail
Group Address      : 232.0.0.1
Source Address     : 172.16.1.1
RP Address         : 192.0.2.4
Advt Router        : 192.0.2.4
Flags              : spt                               Type      : (S,G)
MRIB Next Hop     : 172.16.1.1
```

Configuration

```
MRIB Src Flags      : remote
Keepalive Timer Exp: 0d 00:03:22
Up Time             : 0d 06:39:09      Resolved By          : rtable-u

Up JP State         : Joined            Up JP Expiry          : 0d 00:00:50
Up JP Rpt           : Not Pruned        Up JP Rpt Override    : 0d 00:00:00

Register State      : No Info
Reg From Anycast RP: No

Rpf Neighbor        : 172.16.1.1
Incoming Intf       : int-source
Outgoing Intf List  : 1-mt-239.255.0.1

Curr Fwding Rate    : 3.5 kbps
Forwarded Packets   : 239467            Discarded Packets     : 0
Forwarded Octets    : 10057614          RPF Mismatches        : 0
Spt threshold       : 0 kbps            ECMP opt threshold    : 7
Admin bandwidth     : 1 kbps
```

PE-5 has the incoming interface “1-mt-239.255.0.1”, outgoing interface “int-receiver” and flow rate of 3.5 kbps.

```
*A:PE-5 show router 1 pim group detail
Group Address       : 232.0.0.1
Source Address      : 172.16.1.1
RP Address          : 192.0.2.4
Advt Router         : 192.0.2.2
Flags               : spt                Type                : (S,G)
MRIB Next Hop       : 192.0.2.1
MRIB Src Flags      : remote
Keepalive Timer Exp: 0d 00:02:24
Up Time             : 0d 00:01:10      Resolved By          : rtable-u

Up JP State         : Joined            Up JP Expiry          : 0d 00:00:58
Up JP Rpt           : Not Joined StarG  Up JP Rpt Override    : 0d 00:00:00

Register State      : No Info
Reg From Anycast RP: No

Rpf Neighbor        : 192.0.2.1
Incoming Intf       : 1-mt-239.255.0.1
Outgoing Intf List  : int-receiver

Curr Fwding Rate    : 3.4 kbps
Forwarded Packets   : 649              Discarded Packets     : 0
Forwarded Octets    : 27258            RPF Mismatches        : 0
Spt threshold       : 0 kbps            ECMP opt threshold    : 7
Admin bandwidth     : 1 kbps
```

When Multicast Stream Threshold is Reached

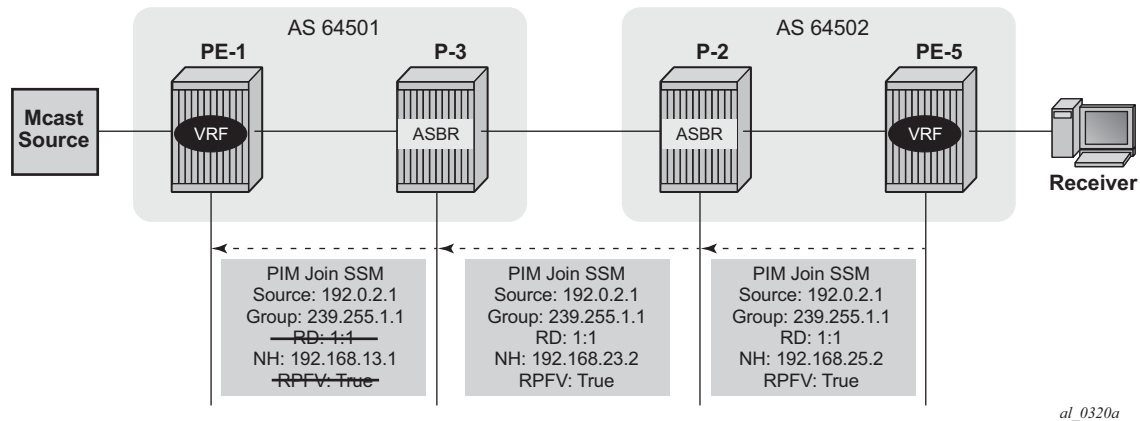


Figure 226: PIM-P Signaling Steps for Data MDT

On PE-5, the **debug router pim packet jp** output shows the PIM join/prune message which is sent to P-2. This message contains the original source of multicast traffic (PE-1: 192.0.2.1) and the RPF Vector (P-2: 192.0.2.2). Note a new multicast group (239.255.1.1) which is signalled for purposes of the Data MDT.

```
"PIM[Instance 1 Base]: Join/Prune
[000 09:48:16.140] PIM-TX ifId 3 ifName int-PE5-P2 -> 224.0.0.13 Length: 48
PIM Version: 2 Msg Type: Join/Prune Checksum: 0x3aae
Upstream Nbr IP : 192.168.25.2 Resvd: 0x0, Num Groups 1, HoldTime 210
  Group: 239.255.1.1/32 Num Joined Srcs: 1, Num Pruned Srcs: 0
  Join Srcs:
    192.0.2.1/32 Flag S <S,G> JA={rpfvMvpn 192.0.2.2 1:1}
```

On P-2, the **debug router pim packet jp** output shows the PIM join/prune message which is propagated to P-3. The source of multicast traffic is untouched while the RPF Vector is modified for Inter-AS propagation.

```
"PIM[Instance 1 Base]: Join/Prune
[001 22:30:36.630] PIM-TX ifId 4 ifName int-P2-P3 -> 224.0.0.13 Length: 48
PIM Version: 2 Msg Type: Join/Prune Checksum: 0x32ae
Upstream Nbr IP : 192.168.23.2 Resvd: 0x0, Num Groups 1, HoldTime 210
  Group: 239.255.1.1/32 Num Joined Srcs: 1, Num Pruned Srcs: 0
  Join Srcs:
    192.0.2.1/32 Flag S <S,G> JA={rpfvMvpn 192.168.23.2 1:1}
```

On P-3, the **debug router pim packet jp** output shows the PIM join/prune message which is propagated to P-3. The source of multicast traffic is untouched while the RPF Vector is not present anymore.

Configuration

```
"PIM[Instance 1 Base]: Join/Prune
[001 22:30:32.770] PIM-TX ifId 2 ifName int-P3-PE1 -> 224.0.0.13 Length: 34
PIM Version: 2 Msg Type: Join/Prune Checksum: 0x85e4
Upstream Nbr IP : 192.168.13.1 Resvd: 0x0, Num Groups 1, HoldTime 210
      Group: 239.255.1.1/32 Num Joined Srcs: 1, Num Pruned Srcs: 0
      Join Srcs:
            192.0.2.1/32 Flag S <S,G>
```

As a result of this signaling, the Data MDT is established between the two ASs. This can be checked with **show router pim group** command.

The PE-1 output shows an additional multicast group (239.255.1.3), which was created in the global routing table (GRT).

```
*A:PE-1# show router pim group
=====
PIM Groups ipv4
=====
Group Address          Type          Spt Bit Inc Intf          No.Oifs
Source Address         RP
-----
239.255.0.1            (S,G)         spt    system          2
192.0.2.1
239.255.0.1            (S,G)         spt    int-PE1-P3      1
192.0.2.5
239.255.1.3            (S,G)         spt    system          1
192.0.2.1
```

The PE-5 output shows an additional multicast group (239.255.1.3), which was created in the GRT.

```
*A:PE-5# show router pim group
=====
PIM Groups ipv4
=====
Group Address          Type          Spt Bit Inc Intf          No.Oifs
Source Address         RP
-----
239.255.0.1            (S,G)         spt    int-PE5-P2      1
192.0.2.1
239.255.0.1            (S,G)         spt    system          2
192.0.2.5
239.255.1.3            (S,G)         spt    int-PE5-P2      1
192.0.2.1
```

The detailed information about the PIM group in a VPRN shows that the Data MDT is used to receive traffic instead of Default MDT.

The PE-5 output for multicast groups in a VPRN 1 has slightly changed: new line “Incoming SPMSI Intf” was added. This indicates that the S-PMSI instance and dedicated Data MDT are used for this particular multicast group. The non-zero rate for the multicast flow is also an indication that multicast traffic is forwarded.


```

*A:PE-5#show router 1 pim group detail
=====
PIM Source Group ipv4
=====
Group Address       : 232.0.0.1
Source Address      : 172.16.1.1
RP Address          : 192.0.2.4
Advt Router         : 192.0.2.2
Flags               : spt                               Type           : (S,G)
MRIB Next Hop       : 192.0.2.1
MRIB Src Flags      : remote
Keepalive Timer Exp: 0d 00:01:10
Up Time             : 0d 00:30:21                       Resolved By        : rtable-u

Up JP State         : Joined                             Up JP Expiry        : 0d 00:00:40
Up JP Rpt           : Not Joined StarG                   Up JP Rpt Override  : 0d 00:00:00

Register State      : No Info
Reg From Anycast RP: No

Rpf Neighbor        : 192.0.2.1
Incoming Intf       : 1-mt-239.255.0.1
Incoming SPMSI Intf: 1-mt-239.255.0.1*
Outgoing Intf List  : int-receiver

Curr Fwding Rate    : 3.4 kbps
Forwarded Packets   : 18187                             Discarded Packets   : 0
Forwarded Octets    : 763854                             RPF Mismatches      : 0
Spt threshold       : 0 kbps                             ECMP opt threshold  : 7
Admin bandwidth     : 1 kbps

```

The **show router 1 pim s-pmsi detail** command can also be used to verify existence of the S-PMSI instance for the VPRN 1. The output is short, but very informative: the multicast group inside VPRN, multicast source IP, multicast group which is used for S-PMSI tunneling and current flow rate can be found.

```

*A:PE-5#show router 1 pim s-pmsi detail
=====
PIM Selective provider tunnels
=====
Md Source Address   : 192.0.2.1                         Md Group Address    : 239.255.1.1
Number of VPN SGs   : 1                                 Uptime              : 0d 00:29:57
MT IfIndex          : 24580                             Egress Fwding Rate  : 3.4 kbps
VPN Group Address    : 232.0.0.1                         VPN Source Address   : 172.16.1.1
State                : RX Joined
Expiry Timer         : 0d 00:02:23

```

Conclusion

Inter-AS MVPN offers flexibility for the operators who can use it to provide additional value added services to their customers. Before implementing this feature in the network the following are required:

- The RPF vector must be enabled on every router for inter-AS MVPN.
- Can be used only with Draft-Rosen mVPN with PIM SSM and MDT SAFI.

Multicast VPN: Sender-Only, Receiver-Only

In This Chapter

This section provides information about multicast VPN sender-only and receiver-only configurations.

Topics in this section include:

- [Applicability on page 1560](#)
- [Summary on page 1561](#)
- [Overview on page 1562](#)
- [Overview on page 1562](#)
- [Conclusion on page 1610](#)

Applicability

This example is applicable to 7950 XRS, 7750 all variants, 7750 SR c4/12 and 7450 mixed mode systems. Chassis mode C or higher must be used. The configuration was tested on release 11.0R3.

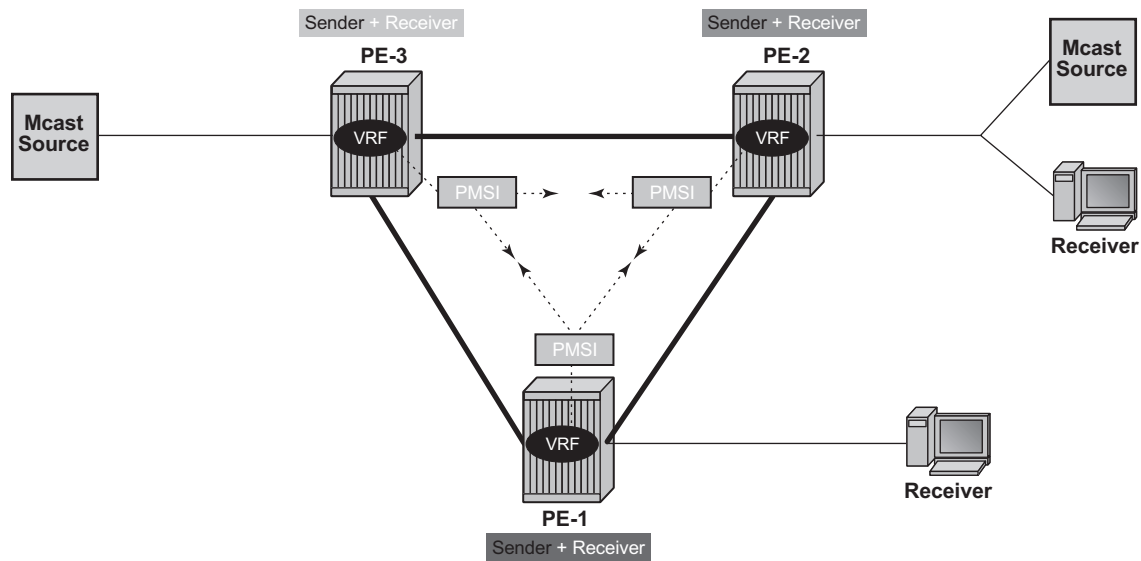
Summary

This example covers a basic technology overview, the network topology and configuration examples which are used for the Multicast VPN (MVPN) sender-only, receiver-only feature.

Knowledge of the Alcatel-Lucent multicast and Layer 3 VPNs concepts are assumed throughout this document.

Overview

By default, if multiple PE nodes form a peering relationship with a common MVPN instance then each PE node originates a multicast tree locally towards the remaining PE nodes that are members of this MVPN instance. This behavior creates a full mesh of Inclusive-Provider Multicast Service Interfaces (I-PMSIs) across all PE nodes in the MVPN.



al_0327

Figure 227: Default PMSI Hierarchy

It is often a case that a given MVPN has many sites with multicast receivers, but only a few sites that host either both receivers and sources, or sources only.

The MVPN sender-only/receiver-only feature optimizes control and data plane resources by preventing unnecessary I-PMSI meshing when a given PE hosts multicast sources only, or multicast receivers only, for a given MVPN. An example of such an optimization is presented in [Figure 228](#).

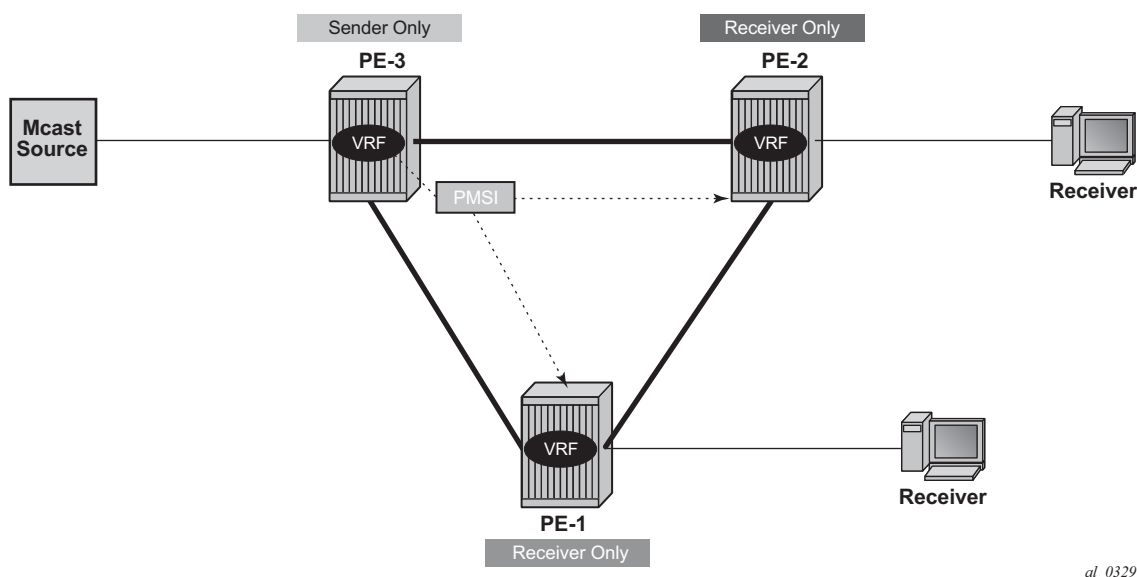


Figure 228: Optimized PMSI Structure

The general rules to follow are:

- For PE nodes that host only multicast sources for a given MVPN, operators can now block PEs, through configuration, from joining I-PMSIs from other PEs in this MVPN.
- For PE nodes that host only multicast receivers for a given MVPN, operators can now block PEs, through configuration, to set-up a local I-PMSI to other PEs in this MVPN.

MVPN sender-only/receiver-only is supported with next generation-MVPN for both IPv4 and IPv6 customer multicast using:

- IPv4 RSVP-TE provider tunnels
- IPv4 LDP provider tunnels

Note: Extra attention should be given to the Bootstrap Router/Rendezvous Point (BSR/RP) placement when sender-only/receiver-only is enabled:

- The RP should be at sender-receiver or sender-only site so that $(*,G)^1$ traffic can be sent over the tunnel
- The BSR should be deployed at the sender-receiver site.
- The BSR can be at a sender-only site if the RPs are at the same site.

1. $*,G$ refers to an individual multicast stream indicating any source (*) and the multicast group (G) used by the stream.

Configuration

The test topology is shown in [Figure 229](#).

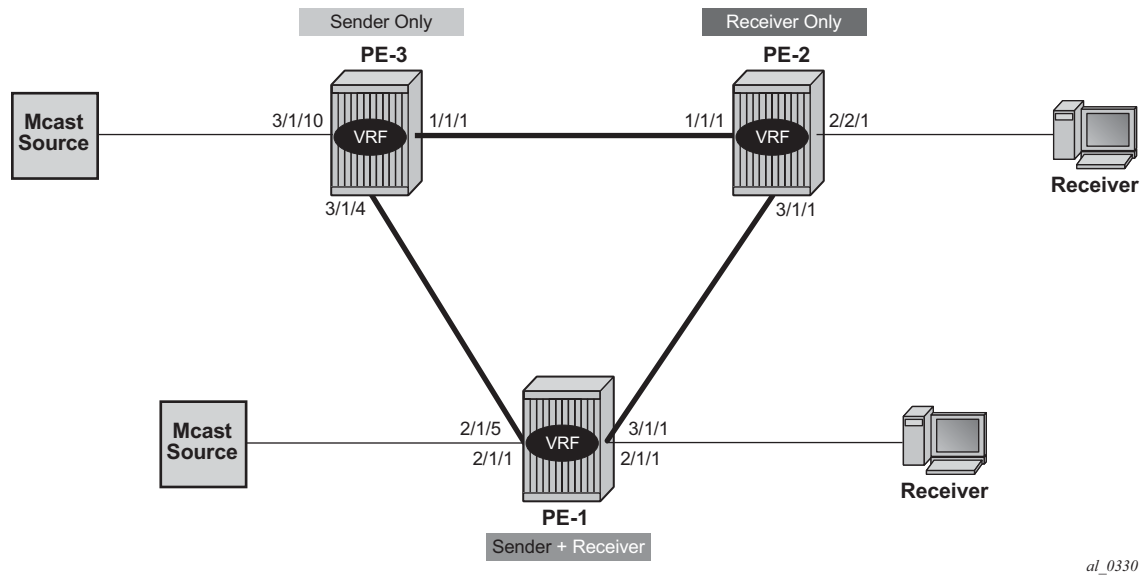


Figure 229: Test Topology

To configure the sender-only/receiver-only feature the following configuration command is used:

```
*A:PE>config>service>vprn>mvpn# mdt-type
- mdt-type {sender-only|receiver-only|sender-receiver}
- no mdt-type
```

sender-receiver is the default option and is visible using the **info detail** command.

This command restricts the MVPN instance per PE node to a specific role and provides an option to configure either a sender-only or receiver-only mode per PE node per service.

Parameters:

sender-only — MVPN has only senders connected to PE node.

receiver-only — MVPN has only receivers connected to PE node.

sender-receiver — MVPN has both sender and receivers connected to PE node.

Considerations:

- Two general approaches for building MVPNs will be covered in detail in this example:
 - Point-to-multipoint (P2MP) RSVP MVPNs
 - Multicast LDP (mLDP) MVPNs
- IPv4 and IPv6 multicast streaming are used for every MVPN at the same time.
- Basic principles of an MVPN including I-PMSI, S-PMSI, mLDP and P2MP RSVP are covered in the [Multicast in a VPN I on page 1389](#) and chapters of this guide.

PIM SSM is used for IPv4/IPv6 Customer (C)-multicast groups.

RSVP-Based MVPN Configuration

Step 0. Configure a basic MVPN using P2MP RSVP as a transport protocol for C-multicast groups. For this setup PE-1 and PE-2 are configured to receive the following multicast groups:

- IPv4 group 232.0.0.1 source 172.16.3.1
- IPv6 group FF3E::8000:1 source 2001:DB8:3::1

Step 1. Configure the MDT type for the MVPN.

Based on the test topology, PE-3 is configured as **sender-only** for the MVPN.

```
*A:PE-3>config>service>vprn# info
-----
description "RSVP based MVPN"
<snip>
interface "int-mcast-source" create
  description "10G STC port 12/2"
  address 172.16.3.2/30
  ipv6
    address 2001:DB8:3::2/127
  exit
  sap 3/1/10:3.1001 create
  exit
exit
pim
  no ipv6-multicast-disable
  apply-to all
  rp
    static
    exit
    bsr-candidate
    shutdown
    exit
    rp-candidate
    shutdown
    exit
  exit
  no shutdown
exit
mvpn
  auto-discovery default
  c-mcast-signaling bgp
  mdt-type sender-only
  provider-tunnel
    inclusive
    rsvp
      lsp-template "mvpn-p2mp-lsp"
      no shutdown
    exit
  exit
exit
vrf-target unicast
exit
```

```

exit
service-name "RSVP based MVPN"
<snip>

```

Based on the test topology PE-2 is configured as **receiver-only** for the MVPN. PE-2 has also static joins for the IPv4 and IPv6 multicast groups:

- group 232.0.0.1, source 172.16.3.1
- group FF3E::8000:1, source 2001:DB8:3::1

```

*A:PE-2>config>service>vprn# info
-----
description "RSVP based MVPN"
<snip>
interface "int-mcast-receiver" create
description "10G STC port 10/2"
address 172.16.2.2/30
ipv6
address 2001:DB8:2::2/127
exit
sap 2/2/1:3.1001 create
exit
exit
igmp
interface "int-mcast-receiver"
static
group 232.0.0.1
source 172.16.3.1
exit
exit
no shutdown
exit
no shutdown
exit
mld
interface "int-mcast-receiver"
static
group FF3E::8000:1
source 2001:DB8:3::1
exit
exit
no shutdown
exit
no shutdown
exit
pim
no ipv6-multicast-disable
rp
static
exit
bsr-candidate
shutdown
exit
rp-candidate
shutdown
exit

```

RSVP-Based MVPN Configuration

```
        exit
        no shutdown
    exit
    mvpn
        auto-discovery default
        c-mcast-signaling bgp
        mdt-type receiver-only
        provider-tunnel
            inclusive
            rsvp
                lsp-template "mvpn-p2mp-lsp"
                no shutdown
            exit
        exit
    exit
    vrf-target unicast
    exit
exit
service-name "RSVP based MVPN"
<snip>
```

Based on the test topology, PE-1 is configured as **sender-receiver** (default) for the MVPN. PE-1 has also static joins for the IPv4 and IPv6 multicast groups:

- group 232.0.0.1, source 172.16.3.1
- group FF3E::8000:1, source 2001:DB8:3::1

```
*A:PE-1>config>service>vprn# info
-----
description "RSVP based MVPN"
<snip>
interface "int-mcast-receiver" create
    description "10G STC port 10/1"
    address 172.16.1.2/30
    ipv6
        address 2001:DB8:1::2/127
    exit
    sap 2/1/1:3.1001 create
    exit
exit
igmp
    interface "int-mcast-receiver"
        static
            group 232.0.0.1
            source 172.16.3.1
        exit
    exit
    no shutdown
exit
no shutdown
exit
mld
    interface "int-mcast-receiver"
        static
            group FF3E::8000:1
            source 2001:DB8:3::1
```

```

        exit
    exit
    no shutdown
exit
no shutdown
exit
pim
    no ipv6-multicast-disable
    apply-to all
    rp
        static
        exit
        bsr-candidate
        shutdown
        exit
        rp-candidate
        shutdown
        exit
    exit
    no shutdown
exit
mvpn
    auto-discovery default
    c-mcast-signaling bgp
    provider-tunnel
        inclusive
        rsvp
            lsp-template "mvpn-p2mp-lsp"
            no shutdown
        exit
    exit
    exit
    vrf-target unicast
    exit
exit
service-name "RSVP based MVPN"
no shutdown

```

Note: The PIM instance must be **shutdown** before the mdt-type is modified; this leads to a multicast service disruption. Trying to change the mdt-type with PIM instance active will result in the message below being displayed.

```

*A:PE-1>config>service>vprn>mvpn# mdt-type sender-only
MINOR: PIM #1100 PIM instance must be shutdown before changing this configuration

```

RSVP-Based MVPN Verification and Debugging

MDT-Type Verification

The status of the MVPN can be checked using the **show>router <service-number> mvpn** command:

PE-1 output:

```
A:PE-1# show router 1 mvpn
=====
MVPN 1 configuration data
=====
signaling          : Bgp                auto-discovery    : Default
UMH Selection      : Highest-Ip          intersite-shared   : Enabled
vrf-import         : N/A
vrf-export         : N/A
vrf-target         : unicast
C-Mcast Import RT  : target:192.0.2.1:3

ipmsi              : rsvp mvpn-p2mp-lsp
i-pmsi P2MP AdmSt  : Up
i-pmsi Tunnel Name : mvpn-p2mp-lsp-1-73741
enable-bfd-root    : false              enable-bfd-leaf    : false
Mdt-type           : sender-receiver

s-pmsi             : none
data-delay-interval: 3 seconds
enable-asm-mdt     : N/A

=====
```

PE-2 output:

```
A:PE-2# show router 1 mvpn
=====
MVPN 1 configuration data
=====
signaling          : Bgp                auto-discovery    : Default
UMH Selection      : Highest-Ip          intersite-shared   : Enabled
vrf-import         : N/A
vrf-export         : N/A
vrf-target         : unicast
C-Mcast Import RT  : target:192.0.2.2:1905

ipmsi              : rsvp mvpn-p2mp-lsp
i-pmsi P2MP AdmSt  : Up
i-pmsi Tunnel Name : mpls-virt-if-640323
enable-bfd-root    : false              enable-bfd-leaf    : false
Mdt-type           : receiver-only

s-pmsi             : none
data-delay-interval: 3 seconds
```

```
enable-asm-mdt      : N/A
```

```
=====
```

PE-3 output:

```
*A:PE-3# show router 1 mvpn
```

```
=====
```

```
MVPN 1 configuration data
```

```
=====
```

```
signaling          : Bgp                auto-discovery      : Default
UMH Selection       : Highest-Ip         intersite-shared     : Enabled
vrf-import          : N/A
vrf-export          : N/A
vrf-target          : unicast
C-Mcast Import RT   : target:192.0.2.3:2086
```

```
ipmsi              : rsvp mvpn-p2mp-lsp
i-pmsi P2MP AdmSt   : Up
i-pmsi Tunnel Name  : mvpn-p2mp-lsp-1-73741
enable-bfd-root     : false              enable-bfd-leaf      : false
Mdt-type            : sender-only
```

```
s-pmsi             : none
data-delay-interval: 3 seconds
enable-asm-mdt      : N/A
```

```
=====
```

BGP Verification and Debugging

When the MDT type is changed, the BGP signaling is slightly modified in order to achieve the signaling optimization.

The PE router does not include the PMSI part in Intra-AD BGP messages when the MVPN is configured with mdt-type as **receiver-only**. The message flow is presented in [Figure 230](#).

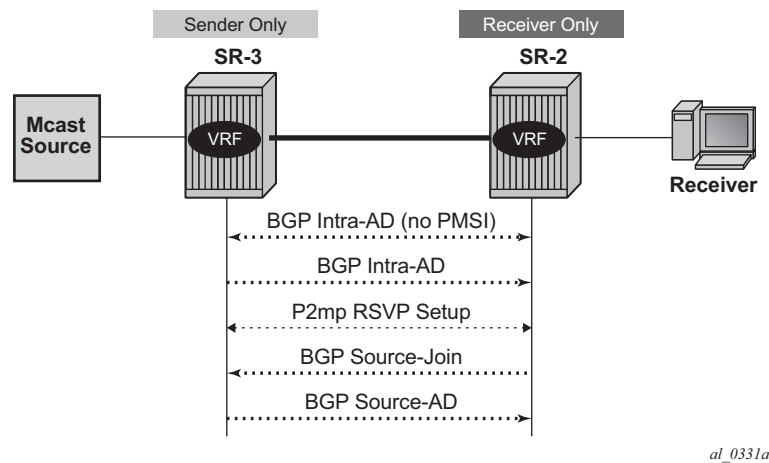


Figure 230: RSVP-Based BGP Message Flow Between PE-2 and PE-3

The BGP debug output below is taken from PE-2 and demonstrates the message flow between PE-2 and PE-3 for MVPN-IPv4 address family.

Note that there is no PMSI part in debug message 62, but the PMSI part is present in message 57, which is sent by PE-3 (**sender-only**).

```
57 2013/10/21 16:58:09.43 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 86
  Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.3
    Type: Intra-AD Len: 12 RD: 64500:103 Orig: 192.0.2.3
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 8 Len: 4 Community:
    no-export
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:64500:1
```



```

Flag: 0xc0 Type: 22 Len: 17 PMSI:
    Tunnel-type RSVP-TE P2MP LSP (1)
    Flags [Leaf not required]
    MPLS Label 0
    P2MP-ID 0x7919, Tunnel-ID: 62688, Extended-Tunnel-ID 192.0.2.3
"

62 2013/10/21 16:58:10.35 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 66
    Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
        Address Family MVPN_IPV4
        NextHop len 4 NextHop 192.0.2.2
        Type: Intra-AD Len: 12 RD: 64500:102 Orig: 192.0.2.2
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 8 Len: 4 Community:
        no-export
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:64500:1
"

67 2013/10/21 16:58:10.65 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 69
    Flag: 0x90 Type: 14 Len: 33 Multiprotocol Reachable NLRI:
        Address Family MVPN_IPV4
        NextHop len 4 NextHop 192.0.2.2
        Type: Source-Join Len:22 RD: 64500:103 SrcAS: 64500 Src: 172.16.3.1
Grp: 232.0.0.1
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:192.0.2.3:2086
"

72 2013/10/21 16:58:11.31 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 65
    Flag: 0x90 Type: 14 Len: 29 Multiprotocol Reachable NLRI:
        Address Family MVPN_IPV4
        NextHop len 4 NextHop 192.0.2.3
        Type: Source-AD Len: 18 RD: 64500:103 Src: 172.16.3.1 Grp: 232.0.0.1
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:64500:1

```

RSVP-Based MVPN Configuration

"

Similar behavior is observed for IPv6 multicast. The BGP debug output below is taken from PE-2 and demonstrates the message flow between PE-2 and PE-3 for the MVPN-IPv6 address family.

Note that there is no PMSI part in debug message 63.

```
58 2013/10/21 16:58:09.42 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 86
  Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV6
    NextHop len 4 NextHop 192.0.2.3
    Type: Intra-AD Len: 12 RD: 64500:103 Orig: 192.0.2.3
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 8 Len: 4 Community:
    no-export
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:64500:1
  Flag: 0xc0 Type: 22 Len: 17 PMSI:
    Tunnel-type RSVP-TE P2MP LSP (1)
    Flags [Leaf not required]
    MPLS Label 0
    P2MP-ID 0x7919, Tunnel-ID: 62688, Extended-Tunnel-ID 192.0.2.3
"

63 2013/10/21 16:58:10.34 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 66
  Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV6
    NextHop len 4 NextHop 192.0.2.2
    Type: Intra-AD Len: 12 RD: 64500:102 Orig: 32.1.13.184
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 8 Len: 4 Community:
    no-export
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:64500:1
"

66 2013/10/21 16:58:10.65 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 93
  Flag: 0x90 Type: 14 Len: 57 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV6
    NextHop len 4 NextHop 192.0.2.2
    Type: Source-Join Len: 46 RD: 64500:103 SrcAS: 64500 Src: 2001:DB8:3
::1 Grp: FF3E::8000:1
```

```

Flag: 0x40 Type: 1 Len: 1 Origin: 0
Flag: 0x40 Type: 2 Len: 0 AS Path:
Flag: 0x80 Type: 4 Len: 4 MED: 0
Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:192.0.2.3:2086
"

71 2013/10/21 16:58:11.30 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 89
    Flag: 0x90 Type: 14 Len: 53 Multiprotocol Reachable NLRI:
        Address Family MVPN_IPV6
        NextHop len 4 NextHop 192.0.2.3
        Type: Source-AD Len: 42 RD: 64500:103 Src: 2001:DB8:3::1 Grp: FF3E
::8000:1
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:64500:1
"

```

The PE router does not change its BGP behavior when the MVPN is configured with mdt-type as **sender-only**. The message flow is presented in [Figure 231](#).

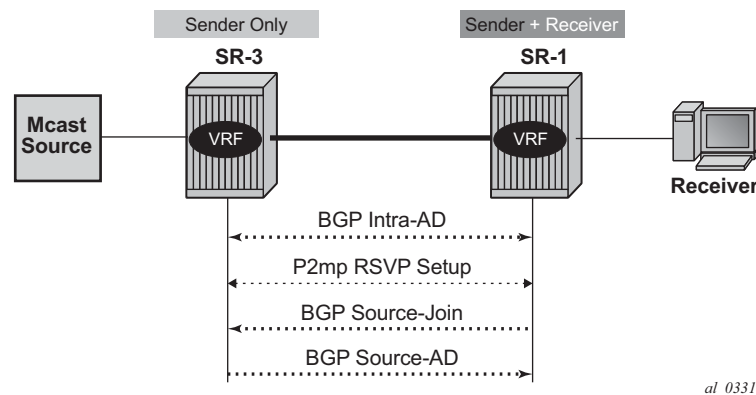


Figure 231: RSVP-Based BGP Message Flow Between PE-1 and PE-3

The BGP debug output below is taken from PE-1 and demonstrates the message flow between PE-1 and PE-3 for the MVPN-IPv4 address family.

Note that the PMSI part is present in debug message 107, which is sent by PE-3 (**sender-only**).

RSVP-Based MVPN Configuration

```
107 2013/10/21 16:58:09.43 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 86
  Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.3
    Type: Intra-AD Len: 12 RD: 64500:103 Orig: 192.0.2.3
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 8 Len: 4 Community:
    no-export
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:64500:1
  Flag: 0xc0 Type: 22 Len: 17 PMSI:
    Tunnel-type RSVP-TE P2MP LSP (1)
    Flags [Leaf not required]
    MPLS Label 0
    P2MP-ID 0x7919, Tunnel-ID: 62688, Extended-Tunnel-ID 192.0.2.3
"

109 2013/10/21 16:58:10.35 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 86
  Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.1
    Type: Intra-AD Len: 12 RD: 64500:101 Orig: 192.0.2.1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 8 Len: 4 Community:
    no-export
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:64500:1
  Flag: 0xc0 Type: 22 Len: 17 PMSI:
    Tunnel-type RSVP-TE P2MP LSP (1)
    Flags [Leaf not required]
    MPLS Label 0
    P2MP-ID 0x7919, Tunnel-ID: 62342, Extended-Tunnel-ID 192.0.2.1
"

116 2013/10/21 16:58:11.30 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 69
  Flag: 0x90 Type: 14 Len: 33 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.1
    Type: Source-Join Len:22 RD: 64500:103 SrcAS: 64500 Src: 172.16.3.1
Grp: 232.0.0.1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
```

```

Flag: 0x40 Type: 2 Len: 0 AS Path:
Flag: 0x80 Type: 4 Len: 4 MED: 0
Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:192.0.2.3:2086
"

120 2013/10/21 16:58:11.31 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 65
    Flag: 0x90 Type: 14 Len: 29 Multiprotocol Reachable NLRI:
        Address Family MVPN_IPV4
        NextHop len 4 NextHop 192.0.2.3
        Type: Source-AD Len: 18 RD: 64500:103 Src: 172.16.3.1 Grp: 232.0.0.1
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:64500:1
"

```

Similar behavior is observed for IPv6 multicast.

The BGP debug output below is taken from PE-1 and demonstrates the message flow between PE-1 and PE-3 for the MVPN-IPv6 address family.

Note: The PMSI part is present in debug message 108, which is sent by PE-3 (**sender-only**).

```

108 2013/10/21 16:58:09.43 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 86
    Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
        Address Family MVPN_IPV6
        NextHop len 4 NextHop 192.0.2.3
        Type: Intra-AD Len: 12 RD: 64500:103 Orig: 192.0.2.3
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 8 Len: 4 Community:
        no-export
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:64500:1
    Flag: 0xc0 Type: 22 Len: 17 PMSI:
        Tunnel-type RSVP-TE P2MP LSP (1)
        Flags [Leaf not required]
        MPLS Label 0
        P2MP-ID 0x7919, Tunnel-ID: 62688, Extended-Tunnel-ID 192.0.2.3
"

110 2013/10/21 16:58:10.34 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE

```

RSVP-Based MVPN Configuration

```
Peer 1: 192.0.2.3 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 86
  Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV6
    NextHop len 4 NextHop 192.0.2.1
    Type: Intra-AD Len: 12 RD: 64500:101 Orig: 192.0.2.1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 8 Len: 4 Community:
    no-export
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:64500:1
  Flag: 0xc0 Type: 22 Len: 17 PMSI:
    Tunnel-type RSVP-TE P2MP LSP (1)
    Flags [Leaf not required]
    MPLS Label 0
    P2MP-ID 0x7919, Tunnel-ID: 62342, Extended-Tunnel-ID 192.0.2.1
"

118 2013/10/21 16:58:11.31 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 93
  Flag: 0x90 Type: 14 Len: 57 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV6
    NextHop len 4 NextHop 192.0.2.1
    Type: Source-Join Len: 46 RD: 64500:103 SrcAS: 64500 Src: 2001:DB8:3
::1 Grp: FF3E::8000:1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:192.0.2.3:2086
"

121 2013/10/21 16:58:11.31 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 89
  Flag: 0x90 Type: 14 Len: 53 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV6
    NextHop len 4 NextHop 192.0.2.3
    Type: Source-AD Len: 42 RD: 64500:103 Src: 2001:DB8:3::1 Grp: FF3E
::8000:1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:64500:1
"
```

The BGP routing table of each router is populated accordingly.

PE-1 (**sender-receiver**) has two Intra-Ad messages from PE-2 and PE-3 and one Source-Ad from PE-3.

```
*A:PE-1# show router bgp routes mvpn-ipv4
=====
BGP Router ID:192.0.2.1      AS:64500      Local AS:64500
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP MVPN-IPv4 Routes
=====
Flag  RouteType      OriginatorIP      LocalPref  MED
      RD          SourceAS          Label
      Nexthop      SourceIP
      As-Path      GroupIP
-----
u*>i  Intra-Ad        192.0.2.2         100        0
      64500:102      -                 -
      192.0.2.2      -
      No As-Path      -
u*>i  Intra-Ad        192.0.2.3         100        0
      64500:103      -                 -
      192.0.2.3      -
      No As-Path      -
u*>i  Source-Ad       -                 100        0
      64500:103      -                 -
      192.0.2.3      172.16.3.1
      No As-Path      232.0.0.1
-----
Routes : 3
=====
```

PE-2 (receiver-only) has two Intra-Ad messages from PE-1 and PE-3 and one Source-Ad from PE-3.

```
*A:PE-2# show router bgp routes mvpn-ipv4
=====
BGP Router ID:192.0.2.2      AS:64500      Local AS:64500
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup

=====
BGP MVPN-IPv4 Routes
=====
Flag  RouteType      OriginatorIP      LocalPref  MED
      RD          SourceAS          Label
      Nexthop      SourceIP
      As-Path      GroupIP
-----
```

RSVP-Based MVPN Configuration

```

u*>i  Intra-Ad          192.0.2.1          100          0
      64500:101         -                  -
      192.0.2.1         -
      No As-Path        -
u*>i  Intra-Ad          192.0.2.3          100          0
      64500:103         -                  -
      192.0.2.3         -
      No As-Path        -
u*>i  Source-Ad         -                  100          0
      64500:103         -                  -
      192.0.2.3         172.16.3.1
      No As-Path        232.0.0.1
-----
Routes : 3
=====

```

PE-3 (**sender-only**) has two Intra-Ad and two Source-Join messages from PE-1 and PE-2.

```

A:PE-3#      show router bgp routes mvpn-ipv4
=====
BGP Router ID:192.0.2.3      AS:64500      Local AS:64500
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP MVPN-IPv4 Routes
=====
Flag  RouteType      OriginatorIP      LocalPref  MED
      RD            SourceAS          Label
      Nexthop       SourceIP
      As-Path       GroupIP
-----
u*>i  Intra-Ad          192.0.2.1          100          0
      64500:101         -                  -
      192.0.2.1         -
      No As-Path        -
u*>i  Intra-Ad          192.0.2.2          100          0
      64500:102         -                  -
      192.0.2.2         -
      No As-Path        -
u*>i  Source-Join      -                  100          0
      64500:103         64500             -
      192.0.2.1         172.16.3.1
      No As-Path        232.0.0.1
*>i  Source-Join      -                  100          0
      64500:103         64500             -
      192.0.2.2         172.16.3.1
      No As-Path        232.0.0.1
-----
Routes : 4
=====

```


RSVP Verification and Debugging

When BGP intra-AD messages are exchanged every PE starts to build multicast tunnels based on the following criteria:

- PE nodes which are configured as **sender-only** for a given MVPN do not join P2MP LSPs from other PEs in this MVPN.
- PE nodes which are configured as receiver-only for a given MVPN do not originate P2MP LSPs to other PEs in this MVPN.

The RSVP session can be checked with the **show>router>rsvp>session** command:

PE-1 (192.0.2.1) has two originating LSPs towards PE-2 (192.0.2.2) and PE-3 (192.0.2.3) and one incoming LSP from PE-3 (**mdt-type sender-only**).

```
*A:PE-1#show router rsvp session
=====
RSVP Sessions
=====
```

From	To	Tunnel ID	LSP ID	Name	State
192.0.2.1	192.0.2.2	62342	52224	mvpn-p2mp-lsp-1-73741::*	Up
192.0.2.1	192.0.2.3	62342	52224	mvpn-p2mp-lsp-1-73741::*	Up
192.0.2.3	192.0.2.1	62688	39424	mvpn-p2mp-lsp-1-73741::*	Up

PE-2 (192.0.2.2) has two incoming LSPs from PE-1 (192.0.2.1) and PE-3 (192.0.2.3) and no originating LSPs due to the fact that PE-2 has **mdt-type receiver-only**.

```
*A:PE-2#show router rsvp session
=====
RSVP Sessions
=====
```

From	To	Tunnel ID	LSP ID	Name	State
192.0.2.1	192.0.2.2	62342	52224	mvpn-p2mp-lsp-1-73741::*	Up
192.0.2.3	192.0.2.2	62688	39424	mvpn-p2mp-lsp-1-73741::*	Up

PE-3 (192.0.2.3) has two originating LSPs towards PE-2 (192.0.2.2) and PE-1 (192.0.2.1) and one incoming LSP from PE-1 (**mdt-type sender-receiver**).

Theoretically there is no need for the LSP from PE-1 towards PE-3 as PE-3 is a sender-only; this minor limitation should be taken into account during planning phase.

```
*A:PE-3#show router rsvp session
=====
RSVP Sessions
=====
```

From	To	Tunnel ID	LSP ID	Name	State
------	----	-----------	--------	------	-------

RSVP-Based MVPN Configuration

```

                                     ID      ID
-----
192.0.2.1      192.0.2.3      62342  52224  mvpn-p2mp-lsp-1-73741::* Up
192.0.2.3      192.0.2.1      62688  39424  mvpn-p2mp-lsp-1-73741::* Up
192.0.2.3      192.0.2.2      62688  39424  mvpn-p2mp-lsp-1-73741::* Up

```

Additional details about originating P2MP paths can be found using the following command:

show>router>mpls p2mp-lsp <lsp name> p2mp-instance <service number> s2l

PE-1 output:

```

*A:PE-1# show router mpls p2mp-lsp "mvpn-p2mp-lsp-1-73741" p2mp-instance "1" s2l
=====
MPLS LSP mvpn-p2mp-lsp-1-73741 S2L
=====
-----
LSP Name      : mvpn-p2mp-lsp-1-73741      P2MP ID      : 1
Adm State     : Up                        Oper State    : Up
P2MPInstance: 1                        Inst-type     : Primary
Adm State     : Up                        Oper State    : PartialInService
-----
S2l Name      To      Next Hop      Adm  Opr
-----
mvpn-p2mp-path 192.0.2.2  192.168.12.1  Up   Up
mvpn-p2mp-path 192.0.2.3  192.168.13.1  Up   Up

```

PE-2 output:

```

*A:PE-2# show router mpls p2mp-lsp
=====
MPLS P2MP LSPs (Originating)
=====
-----
LSP Name      Tun      Fastfail  Adm  Opr
               Id      Config
-----
No Matching Entries Found
=====

```

PE-3 output:

```

A:PE-# show router mpls p2mp-lsp "mvpn-p2mp-lsp-1-73741" p2mp-instance "1" s2l
=====
MPLS LSP mvpn-p2mp-lsp-1-73741 S2L
=====
-----
LSP Name      : mvpn-p2mp-lsp-1-73741      P2MP ID      : 1
Adm State     : Up                        Oper State    : Up
P2MPInstance: 1                        Inst-type     : Primary
Adm State     : Up                        Oper State    : PartialInService
-----
S2l Name      To      Next Hop      Adm  Opr
-----
mvpn-p2mp-path 192.0.2.1  192.168.13.0  Up   Up
mvpn-p2mp-path 192.0.2.2  192.168.23.0  Up   Up

```

Multicast Stream Verification

The status of the multicast groups/streams can be verified using **show>router <sid>>pim group detail ipv6** command:

There is an IPv4 receiver connected to PE-1. An I-PMSI is used as the incoming interface and the physical interface where the receiver is connected is used as the outgoing interface.

```
A:PE-1#show router 1 pim group detail
=====
PIM Source Group ipv4
=====
Group Address      : 232.0.0.1
Source Address     : 172.16.3.1
RP Address         : 0
Advt Router        : 192.0.2.3
Flags              :                               Type           : (S,G)
MRIB Next Hop      : 192.0.2.3
<snip>
Rpf Neighbor       : 192.0.2.3
Incoming Intf      : mpls-if-73743
Outgoing Intf List : int-mcast-receiver

Curr Fwding Rate   : 85636.0 kbps
<snip>
```

There is an IPv4 receiver connected to PE-2. An I-PMSI is used as the incoming interface and the physical interface where the receiver is connected is used as the outgoing interface.

A:PE-2# show router 1 pim group detail

```
=====
PIM Source Group ipv4
=====
Group Address      : 232.0.0.1
Source Address     : 172.16.3.1
RP Address         : 0
Advt Router        : 192.0.2.3
Flags              :                               Type           : (S,G)
MRIB Next Hop      : 192.0.2.3
<snip>
Rpf Neighbor       : 192.0.2.3
Incoming Intf      : mpls-if-73753
Outgoing Intf List : int-mcast-receiver

Curr Fwding Rate   : 85624.6 kbps
<snip>
```

There is an IPv4 sender connected to PE-3. An I-PMSI is used as the outgoing interface and the physical interface where sender is connected is used as the incoming interface.

A:PE-3#show router 1 pim group detail

RSVP-Based MVPN Configuration

```
=====
PIM Source Group ipv4
=====
Group Address      : 232.0.0.1
Source Address     : 172.16.3.1
RP Address         : 0
Advt Router        : 192.0.2.3
Flags              :                               Type          : (S,G)
MRIB Next Hop      : 172.16.3.1
<snip>
Rpf Neighbor       : 172.16.3.1
Incoming Intf      : int-mcast-source
Outgoing Intf List : mpls-if-73741

Curr Fwding Rate   : 85625.0 kbps
<snip>
```

Similar behavior is observed for IPv6 multicast.

There is an IPv6 receiver connected to PE-1. An I-PMSI is used as the incoming interface and the physical interface where the receiver is connected is used as the outgoing interface.

```
A:PE-1#show router 1 pim group detail ipv6
=====
PIM Source Group ipv6
=====
Group Address      : FF3E::8000:1
Source Address     : 2001:DB8:3::1
RP Address         : 0
Advt Router        : 192.0.2.3
Flags              :                               Type          : (S,G)
MRIB Next Hop      : 192.0.2.3
<snip>
Rpf Neighbor       : 192.0.2.3
Incoming Intf      : mpls-if-73743
Outgoing Intf List : int-mcast-receiver

Curr Fwding Rate   : 2140.5 kbps
<snip>
```

There is an IPv6 receiver connected to PE-2. An I-PMSI is used as the incoming interface and the physical interface where the receiver is connected is used as the outgoing interface.

```
A:PE-2# show router 1 pim group detail ipv6
=====
PIM Source Group ipv6
=====
Group Address      : FF3E::8000:1
Source Address     : 2001:DB8:3::1
RP Address         : 0
Advt Router        : 192.0.2.3
Flags              :                               Type          : (S,G)
MRIB Next Hop      : 192.0.2.3
<snip>
Rpf Neighbor       : 192.0.2.3
```

```
Incoming Intf      : mpls-if-73753
Outgoing Intf List : int-mcast-receiver

Curr Fwding Rate   : 2139.4 kbps
<snip>
```

There is an IPv6 sender connected to PE-3. An I-PMSI is used as the outgoing interface and the physical interface where the sender is connected is used as the incoming interface.

```
A:PE-3# show router 1 pim group detail ipv6
=====
PIM Source Group ipv6
=====
Group Address      : FF3E::8000:1
Source Address     : 2001:DB8:3::1
RP Address         : 0
Advt Router        : 192.0.2.3
Flags              :
MRIB Next Hop      : 2001:DB8:3::1
<snip>
Rpf Neighbor       : 2001:DB8:3::1
Incoming Intf      : int-mcast-source
Outgoing Intf List : mpls-if-73741

Curr Fwding Rate   : 2140.5 kbps
<snip>
```

mLDP-Based MVPN Configuration

Step 0: Configure a basic MVPN using mLDP as a transport protocol for C-multicast groups. PE-1 and PE-2 have static joins for the IPv4/IPv6 multicast groups:

- group 232.0.0.1, source 172.16.3.1
- group FF3E::8000:1, source 2001:DB8:3::1

Step 1. Configure the MDT type for the MVPN.

Based on the test topology PE-3 is configured as sender-only for MVPN:

```
*A:PE-3>config>service>vprn# info
-----
description "mLDP based MVPN"
<snip>
interface "int-mcast-source" create
description "10G STC port 12/2"
address 172.16.3.2/30
ipv6
address 2001:DB8:3::2/126
exit
sap 3/1/10:3.1002 create
exit
exit
pim
no ipv6-multicast-disable
apply-to all
rp
static
exit
bsr-candidate
shutdown
exit
rp-candidate
shutdown
exit
exit
no shutdown
exit
mvpn
auto-discovery default
c-mcast-signaling bgp
mdt-type sender-only
provider-tunnel
inclusive
mldp
no shutdown
exit
exit
vrf-target unicast
exit
```

```

exit
service-name "mLDP based MVPN"
spoke-sdp 10132 create
exit
no shutdown
<snip>

```

Based on the test topology, PE-2 is configured as receiver-only for the MVPN. PE-2 has also static joins for the IPv4 and IPv6 multicast groups:

- group 232.0.0.1, source 172.16.3.1
- group FF3E::8000:1, source 2001:DB8:3::1

```

*A:PE-2>config>service>vprn# info
-----
description "mLDP based MVPN"
<snip>
interface "int-mcast-receiver" create
description "10G STC port 10/2"
address 172.16.2.2/30
ipv6
address 2001:DB8:2::2/126
exit
sap 2/2/1:3.1002 create
exit
exit
igmp
interface "int-mcast-receiver"
static
group 232.0.0.1
source 172.16.3.1
exit
exit
no shutdown
exit
no shutdown
exit
mld
interface "int-mcast-receiver"
static
group FF3E::8000:1
source 2001:DB8:3::1
exit
exit
no shutdown
exit
no shutdown
exit
pim
no ipv6-multicast-disable
apply-to all
rp
static
exit
bsr-candidate
shutdown

```

mLDP-Based MVPN Configuration

```
        exit
        rp-candidate
        shutdown
    exit
    exit
    no shutdown
exit
mvpn
    auto-discovery default
    c-mcast-signaling bgp
    mdt-type receiver-only
    provider-tunnel
        inclusive
        mldp
        no shutdown
    exit
    exit
    exit
    vrf-target unicast
    exit
exit
service-name "mLDP based MVPN"
<snip>
```

Based on the test topology, PE-1 is configured as **sender-receiver** (default) for the MVPN. PE-1 has also static joins for the IPv4 and IPv6 multicast groups:

- group 232.0.0.1, source 172.16.3.1
- group FF3E::8000:1, source 2001:DB8:3::1

```
*A:PE-1>config>service>vprn# info
-----
description "mLDP based MVPN"
<snip>
interface "int-mcast-receiver" create
    description "10G STC port 10/1"
    address 172.16.1.2/30
    ipv6
        address 2001:DB8:1::2/126
    exit
    sap 2/1/1:3.1002 create
    exit
exit
igmp
    interface "int-mcast-receiver"
        static
            group 232.0.0.1
            source 172.16.3.1
        exit
    exit
    no shutdown
exit
no shutdown
exit
mld
    interface "int-mcast-receiver"
```



```

        static
        group FF3E::8000:1
        source 2001:DB8:3::1
        exit
    exit
    no shutdown
exit
no shutdown
exit
pim
no ipv6-multicast-disable
apply-to all
rp
    static
    exit
    bsr-candidate
    shutdown
    exit
    rp-candidate
    shutdown
    exit
    exit
    no shutdown
exit
mvpn
    auto-discovery default
    c-mcast-signaling bgp
    provider-tunnel
    inclusive
    mldp
    no shutdown
    exit
    exit
    vrf-target unicast
    exit
exit
service-name "mLDP based MVPN"

```

Note: The PIM instance must be **shutdown** before the mdt-type is modified; this leads to multicast service disruption. Trying to change mdt-type with the PIM instance active will result in the message below being displayed.

```

*A:PE-1>config>service>vprn>mvpn# mdt-type sender-only
MINOR: PIM #1100 PIM instance must be shutdown before changing this configuration

```

mLDP-Based MVPN Verification and Debugging

MDT-Type Verification

The status of the MVPN can be checked using the following command:

```
show router <service-number> mvpn
```

PE-1 output:

```
*A:PE-1# show router 2 mvpn
=====
MVPN 2 configuration data
=====
signaling          : Bgp          auto-discovery      : Default
UMH Selection      : Highest-Ip   intersite-shared    : Enabled
vrf-import         : N/A
vrf-export         : N/A
vrf-target         : unicast
C-Mcast Import RT  : target:192.0.2.1:2

ipmsi              : ldp
i-pmsi P2MP AdmSt  : Up
i-pmsi Tunnel Name : mpls-if-73734
Mdt-type           : sender-receiver

s-pmsi             : none
data-delay-interval: 3 seconds
enable-asm-mdt     : N/A
=====
```

PE-2 output:

```
A:PE-2# show router 2 mvpn
=====
MVPN 2 configuration data
=====
signaling          : Bgp          auto-discovery      : Default
UMH Selection      : Highest-Ip   intersite-shared    : Enabled
vrf-import         : N/A
vrf-export         : N/A
vrf-target         : unicast
C-Mcast Import RT  : target:192.0.2.2:1906

ipmsi              : ldp
i-pmsi P2MP AdmSt  : Up
i-pmsi Tunnel Name : mpls-virt-if-640321
Mdt-type           : receiver-only

s-pmsi             : none
data-delay-interval: 3 seconds
enable-asm-mdt     : N/A
=====
```

PE-3 output:

```
*A:PE-3# show router 2 mvpn
=====
MVPN 2 configuration data
=====
signaling          : Bgp                auto-discovery    : Default
UMH Selection      : Highest-IP         intersite-shared   : Enabled
vrf-import         : N/A
vrf-export         : N/A
vrf-target         : unicast
C-Mcast Import RT  : target:192.0.2.3:2087

ipmsi              : ldp
i-pmsi P2MP AdmSt  : Up
i-pmsi Tunnel Name : mpls-if-73752
Mdt-type           : sender-only

s-pmsi             : none
data-delay-interval: 3 seconds
enable-asm-mdt     : N/A
=====
```

BGP Verification and Debugging

When the MDT type is changed the BGP signaling is slightly modified in order to achieve the signaling optimization. The PE router does not include the PMSI part in Intra-AD BGP messages when the MVPN is configured with mdt-type as **receiver-only**.

The message flow is presented in [Figure 232](#).

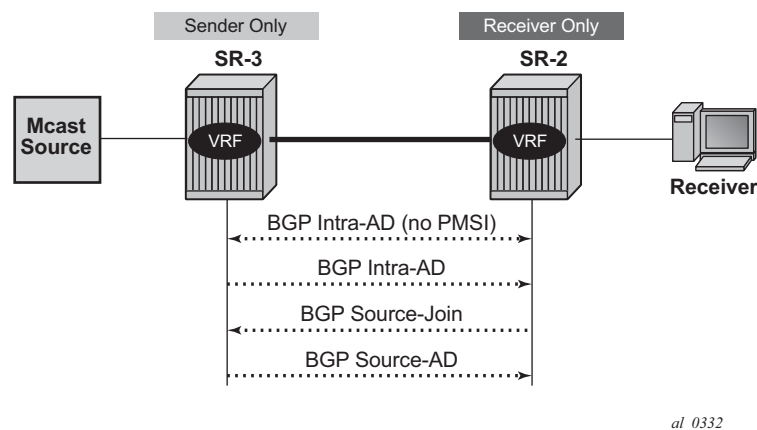


Figure 232: mLDP-Based BGP Message Flow Between PE-2 and PE-3

In order to demonstrate the BGP message flow sequence the following initialization steps are taken:

0. Bring down the VPRN service, PIM protocol in a VPRN and IGMP/MLD protocol. As a result the state of all signaling protocols is cleared.
1. Bring up the VPRN service. BGP exchanges unicast routing information.
2. Bring up the IPv4 PIM protocol. BGP exchanges IPv4 multicast routing information in order to build the PMSI infrastructure.
3. Bring up IGMP and add a static IGMP join where it is applicable. BGP exchanges IPv4 multicast routing information in order to propagate the multicast traffic to the receiver.
4. Bring up the IPv6 PIM protocol. BGP exchanges IPv6 multicast routing information in order to build the PMSI infrastructure.
5. Bring up MLD and add a static MLD join where it is applicable. BGP exchanges IPv6 multicast routing information in order to propagate the multicast traffic to the receiver.

The BGP debug below is taken from PE-2 and demonstrates the message flow between PE-2 and PE-3. VPN-IPv4 and VPN-IPv6 updates are not present in this output.

Step 0: Bring down service and protocols to clear the state of all signalling protocols.

```
*A:PE-2>config>service>vprn# shutdown
*A:PE-2>config>service>vprn# pim shutdown
*A:PE-2>config>service>vprn# igmp shutdown
*A:PE-2>config>service>vprn# mld shutdown
*A:PE-2>config>service>vprn# pim ipv6-multicast-disable
```

Step 1. Enable the VPRN service on PE-2. PE-2 immediately receives Intra-AD messages from PE-3 because the remote VPRN service is already enabled for IPv4 and IPv6 multicast propagation.

```
*A:PE-2>config>service>vprn# no shutdown
*A:PE-2>config>service>vprn#

4099 2013/10/25 13:43:04.45 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 91
  Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.3
    Type: Intra-AD Len: 12 RD: 64500:203 Orig: 192.0.2.3
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 8 Len: 4 Community:
    no-export
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
```

```

        target:64500:2
    Flag: 0xc0 Type: 22 Len: 22 PMSI:
        Tunnel-type LDP P2MP LSP (2)
        Flags [Leaf not required]
        MPLS Label 0
        Root-Node 192.0.2.3, LSP-ID 0x2001
"

4100 2013/10/25 13:43:04.46 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 91
    Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
        Address Family MVPN_IPV6
        NextHop len 4 NextHop 192.0.2.3
        Type: Intra-AD Len: 12 RD: 64500:203 Orig: 192.0.2.3
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 8 Len: 4 Community:
        no-export
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:64500:2
    Flag: 0xc0 Type: 22 Len: 22 PMSI:
        Tunnel-type LDP P2MP LSP (2)
        Flags [Leaf not required]
        MPLS Label 0
        Root-Node 192.0.2.3, LSP-ID 0x2001
"

```

Step 2. Enable only PIM IPv4 for the service on PE-2. PE-2 immediately sends Intra-AD messages to PE-3. Note: there is no PMSI part present in debug message 4101.

```

*A:PE-2>config>service>vprn# pim no shutdown
*A:PE-2>config>service>vprn#
4101 2013/10/25 13:43:16.34 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 66
    Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
        Address Family MVPN_IPV4
        NextHop len 4 NextHop 192.0.2.2
        Type: Intra-AD Len: 12 RD: 64500:202 Orig: 192.0.2.2
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 8 Len: 4 Community:
        no-export
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:64500:2
"

```

Step 3. Bring up IGMP and add a static IGMP join for the service on a PE-2. PE-2 immediately sends a source-join message to PE-3 and receives a source-AD message from PE-3.

```
*A:PE-2>config>service>vprn# igmp shutdown
*A:PE-2>config>service>vprn# igmp interface "int-mcast-receiver" static group 232.0.0.1
source 172.16.3.1
```

```
*A:PE-2>config>service>vprn#
4102 2013/10/25 13:43:25.36 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 69
  Flag: 0x90 Type: 14 Len: 33 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.2
    Type: Source-Join Len:22 RD: 64500:203 SrcAS: 64500 Src: 172.16.3.1
Grp: 232.0.0.1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:192.0.2.3:2087
"
```

```
4103 2013/10/25 13:43:25.36 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 65
  Flag: 0x90 Type: 14 Len: 29 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.3
    Type: Source-AD Len: 18 RD: 64500:203 Src: 172.16.3.1 Grp: 232.0.0.1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:64500:2
"
```

Step 4. Enable PIM IPv6 for the service on PE-2. PE-2 immediately sends Intra-AD messages to PE-3.

```
*A:PE-2>config>service>vprn# pim no ipv6-multicast-disable
*A:PE-2>config>service>vprn#
4104 2013/10/25 13:43:47.36 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 66
  Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV6
    NextHop len 4 NextHop 192.0.2.2
    Type: Intra-AD Len: 12 RD: 64500:202 Orig: 32.1.13.184
```

```

Flag: 0x40 Type: 1 Len: 1 Origin: 0
Flag: 0x40 Type: 2 Len: 0 AS Path:
Flag: 0x80 Type: 4 Len: 4 MED: 0
Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
Flag: 0xc0 Type: 8 Len: 4 Community:
    no-export
Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:64500:2
"

```

Step 5. Bring up MLD and add a static MLD join for the service on a PE-2. PE-2 immediately sends a source-join message to PE-3 and receives a source-AD message from PE-3.

```

*A:PE-2>config>service>vprn# mld shutdown
*A:PE-2>config>service>vprn# mld interface "int-mcast-receiver" static group FF3E::8000:1
source 2001:DB8:3::1

*A:PE-2>config>service>vprn#
4105 2013/10/25 13:43:54.36 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 93
    Flag: 0x90 Type: 14 Len: 57 Multiprotocol Reachable NLRI:
        Address Family MVPN_IPV6
        NextHop len 4 NextHop 192.0.2.2
        Type: Source-Join Len: 46 RD: 64500:203 SrcAS: 64500 Src: 2001:DB8:3
:::1 Grp: FF3E::8000:1
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:192.0.2.3:2087
"

4106 2013/10/25 13:43:54.36 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 89
    Flag: 0x90 Type: 14 Len: 53 Multiprotocol Reachable NLRI:
        Address Family MVPN_IPV6
        NextHop len 4 NextHop 192.0.2.3
        Type: Source-AD Len: 42 RD: 64500:203 Src: 2001:DB8:3::1 Grp: FF3E
:::8000:1
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:64500:2
"

```

The same information can be gathered using the following show commands.

show>router>bgp>neighbor <peer> advertised-routes [mvpn-ipv4|mvpn-ipv6]

show>router>bgp>neighbor <peer> received-routes [mvpn-ipv4|mvpn-ipv6]

PE-2 output for the advertised routes for the mvpn-ipv4 address family.

```
*A:PE-2# show router bgp neighbor 192.0.2.3 advertised-routes mvpn-ipv4
=====
BGP MVPN-IPv4 Routes
=====
```

Flag	RouteType	OriginatorIP	LocalPref	MED
	RD	SourceAS		Label
	Nexthop	SourceIP		
	As-Path	GroupIP		
i	Intra-Ad	192.0.2.2	100	0
	64500:202	-	-	
	192.0.2.2	-		
	No As-Path	-		
i	Source-Join	-	100	0
	64500:203	64500	-	
	192.0.2.2	172.16.3.1		
	No As-Path	232.0.0.1		

```
-----
Routes : 2
=====
```

PE-2 output for the advertised routers for the mvpn-ipv6 address family.

```
*A:PE-2# show router bgp neighbor 192.0.2.3 advertised-routes mvpn-ipv6
=====
BGP MVPN-IPv6 Routes
=====
```

Flag	RouteType	OriginatorIP	LocalPref	MED
	RD	SourceAS		Label
	Nexthop	SourceIP		
	As-Path	GroupIP		
i	Intra-Ad	32.1.13.184	100	0
	64500:202	-	-	
	192.0.2.2	-		
	No As-Path	-		
i	Source-Join	-	100	0
	64500:203	64500	-	
	192.0.2.2	2001:DB8:3::1		
	No As-Path	FF3E::8000:1		

```
-----
Routes : 2
=====
```


PE-2 output for the received routes for the mvpn-ipv4 address family.

```
*A:PE-2# show router bgp neighbor 192.0.2.3 received-routes mvpn-ipv4
=====
BGP MVPN-IPv4 Routes
=====
```

Flag	RouteType	OriginatorIP	LocalPref	MED
	RD	SourceAS		Label
	Nexthop	SourceIP		
	As-Path	GroupIP		
u*>i	Intra-Ad	192.0.2.3	100	0
	64500:203	-	-	-
	192.0.2.3	-		
	No As-Path	-		
u*>i	Source-Ad	-	100	0
	64500:203	-	-	-
	192.0.2.3	172.16.3.1		
	No As-Path	232.0.0.1		

```
-----
Routes : 2
=====
```

PE-2 output for the received routes for the mvpn-ipv6 address family.

```
*A:PE-2# show router bgp neighbor 192.0.2.3 received-routes mvpn-ipv6
=====
BGP MVPN-IPv6 Routes
=====
```

Flag	RouteType	OriginatorIP	LocalPref	MED
	RD	SourceAS		Label
	Nexthop	SourceIP		
	As-Path	GroupIP		
u*>i	Intra-Ad	192.0.2.3	100	0
	64500:203	-	-	-
	192.0.2.3	-		
	No As-Path	-		
u*>i	Source-Ad	-	100	0
	64500:203	-	-	-
	192.0.2.3	2001:DB8:3::1		
	No As-Path	FF3E::8000:1		

```
-----
Routes : 2
=====
```

The PE router does not change the BGP behavior when the MVPN is configured with mdt-type as **sender-only**. A schematic of the message flow is presented in [Figure 233](#).

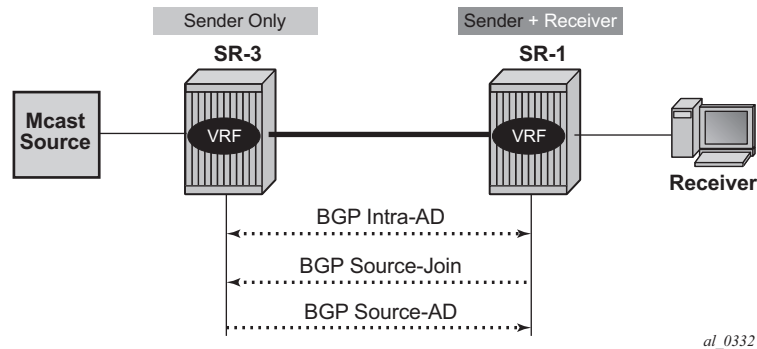


Figure 233: mLDP-Based BGP Message Flow Between PE-1 and PE-3

In order to demonstrate the BGP message flow sequence the following initialization steps are taken:

0. Bring down VPRN service, PIM protocol in a VPRN and IGMP/MLD protocol. As a result the state of all signaling protocols is cleared.
1. Bring up the VPRN service. BGP exchanges unicast routing information.
2. Bring up the IPv4 PIM protocol. BGP exchanges IPv4 multicast routing information in order to build PMSI infrastructure.
3. Bring up IGMP and add a static IGMP join where it is applicable. BGP exchanges IPv4 multicast routing information in order to propagate the multicast traffic to the receiver.
4. Bring up the IPv6 PIM protocol. BGP exchanges IPv6 multicast routing information in order to build the PMSI infrastructure.
5. Bring up MLD and add a static MLD join where it is applicable. BGP exchanges IPv6 multicast routing information in order to propagate the multicast traffic to the receiver.

The BGP debug output below is taken from PE-1 and demonstrates the message flow between PE-1 and PE-3.

Note: The PMSI part is present in debug messages 7584 and 7585, which are sent by PE-3 (**sender-only**).

Step 0: Bring down service and protocols to clear the state of all signaling protocols.

```
*A:PE-1>config>service>vprn# shutdown
*A:PE-1>config>service>vprn# pim shutdown
*A:PE-1>config>service>vprn# igmp shutdown
*A:PE-1>config>service>vprn# mld shutdown
*A:PE-1>config>service>vprn# pim ipv6-multicast-disable
```

Step 1. Enable the VPRN service on PE-1. PE-1 immediately receives Intra-AD messages from PE-3 because the remote VPRN service is already enabled for IPv4 and IPv6 multicast propagation. The PMSI attribute is present in both messages.

```
*A:PE-1>config>service>vprn# no shutdown
7584 2013/10/25 13:15:30.73 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 91
  Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.3
    Type: Intra-AD Len: 12 RD: 64500:203 Orig: 192.0.2.3
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 8 Len: 4 Community:
    no-export
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:64500:2
  Flag: 0xc0 Type: 22 Len: 22 PMSI:
    Tunnel-type LDP P2MP LSP (2)
    Flags [Leaf not required]
    MPLS Label 0
    Root-Node 192.0.2.3, LSP-ID 0x2001
"

7585 2013/10/25 13:15:30.73 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 91
  Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV6
    NextHop len 4 NextHop 192.0.2.3
    Type: Intra-AD Len: 12 RD: 64500:203 Orig: 192.0.2.3
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 8 Len: 4 Community:
    no-export
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:64500:2
  Flag: 0xc0 Type: 22 Len: 22 PMSI:
    Tunnel-type LDP P2MP LSP (2)
    Flags [Leaf not required]
    MPLS Label 0
    Root-Node 192.0.2.3, LSP-ID 0x2001
"
```

Step 2. Enable PIM IPv4 only for the service on PE-1. PE-1 immediately sends Intra-AD messages to PE-3.

```
*A:PE-1>config>service>vprn# pim no shutdown
*A:PE-1>config>service>vprn#

7586 2013/10/25 13:16:43.72 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 91
  Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.1
    Type: Intra-AD Len: 12 RD: 64500:201 Orig: 192.0.2.1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 8 Len: 4 Community:
    no-export
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:64500:2
  Flag: 0xc0 Type: 22 Len: 22 PMSI:
    Tunnel-type LDP P2MP LSP (2)
    Flags [Leaf not required]
    MPLS Label 0
    Root-Node 192.0.2.1, LSP-ID 0x2001
"
```

Step 3. Bring up IGMP and add a static IGMP join for the service on a PE-1. PE-1 immediately sends a source-join message to PE-3 and receives a source-AD message from PE-3.

```
*A:PE-1>config>service>vprn# igmp no shutdown
*A:PE-1>config>service>vprn# igmp interface "int-mcast-receiver" static group 232.0.0.1
source 172.16.3.1
```

```
7587 2013/10/25 13:17:19.68 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 69
  Flag: 0x90 Type: 14 Len: 33 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.1
    Type: Source-Join Len:22 RD: 64500:203 SrcAS: 64500 Src: 172.16.3.1
Grp: 232.0.0.1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:192.0.2.3:2087
"
```

```
7588 2013/10/25 13:17:20.43 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 65
  Flag: 0x90 Type: 14 Len: 29 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.3
    Type: Source-AD Len: 18 RD: 64500:203 Src: 172.16.3.1 Grp: 232.0.0.1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:64500:2
"
```

Step 4. Enable PIM IPv6 for the service on PE-1. PE-1 immediately sends Intra-AD messages to PE-3.

```
*A:PE-1>config>service>vprn# pim no ipv6-multicast-disable
*A:PE-1>config>service>vprn#
```

```
7589 2013/10/25 13:18:42.72 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 91
  Flag: 0x90 Type: 14 Len: 23 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV6
```

```

        NextHop len 4 NextHop 192.0.2.1
        Type: Intra-AD Len: 12 RD: 64500:201 Orig: 192.0.2.1
Flag: 0x40 Type: 1 Len: 1 Origin: 0
Flag: 0x40 Type: 2 Len: 0 AS Path:
Flag: 0x80 Type: 4 Len: 4 MED: 0
Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
Flag: 0xc0 Type: 8 Len: 4 Community:
        no-export
Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:64500:2
Flag: 0xc0 Type: 22 Len: 22 PMSI:
        Tunnel-type LDP P2MP LSP (2)
        Flags [Leaf not required]
        MPLS Label 0
        Root-Node 192.0.2.1, LSP-ID 0x2001
"

```

Step 5. Bring up MLD and add a static MLD join for the service on a PE-1. PE-1 immediately sends a source-join message to PE-3 and receives a source-AD message from PE-3.

```

*A:PE-1>config>service>vprn# mld no shutdown
*A:PE-1>config>service>vprn# mld interface "int-mcast-receiver" static group FF3E::8000:1
source 2001:DB8:3::1

```

```

7590 2013/10/25 13:18:57.68 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Send BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 93
    Flag: 0x90 Type: 14 Len: 57 Multiprotocol Reachable NLRI:
        Address Family MVPN_IPV6
        NextHop len 4 NextHop 192.0.2.1
        Type: Source-Join Len: 46 RD: 64500:203 SrcAS: 64500 Src: 2001:DB8:3
::1 Grp: FF3E::8000:1
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:192.0.2.3:2087
"

```

```

7591 2013/10/25 13:18:58.43 CEST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.3
"Peer 1: 192.0.2.3: UPDATE
Peer 1: 192.0.2.3 - Received BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 89
    Flag: 0x90 Type: 14 Len: 53 Multiprotocol Reachable NLRI:
        Address Family MVPN_IPV6
        NextHop len 4 NextHop 192.0.2.3
        Type: Source-AD Len: 42 RD: 64500:203 Src: 2001:DB8:3::1 Grp: FF3E
::8000:1
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:

```

```
target:64500:2
"
```

The same information can be gathered using the following show commands.

show router bgp neighbor <peer> advertised-routes [mvpn-ipv4|mvpn-ipv6]

show router bgp neighbor <peer> received-routes [mvpn-ipv4|mvpn-ipv6]

PE-1 output for the advertised routes for the mvpn-ipv4 address family.

```
*A:PE-1# show router bgp neighbor 192.0.2.3 advertised-routes mvpn-ipv4
=====
BGP MVPN-IPv4 Routes
=====
Flag  RouteType      OriginatorIP      LocalPref  MED
      RD          SourceAS          Label
      Nexthop      SourceIP
      As-Path      GroupIP
-----
i     Intra-Ad      192.0.2.1         100        0
      64500:201    -                 -
      192.0.2.1    -
      No As-Path   -
i     Source-Join   -                 100        0
      64500:203    64500             -
      192.0.2.1    172.16.3.1
      No As-Path   232.0.0.1
-----
Routes : 2
=====
```

PE-1 output for the advertised routes for the mvpn-ipv6 address family.

```
*A:PE-1# show router bgp neighbor 192.0.2.3 advertised-routes mvpn-ipv6
=====
BGP MVPN-IPv6 Routes
=====
Flag  RouteType      OriginatorIP      LocalPref  MED
      RD          SourceAS          Label
      Nexthop      SourceIP
      As-Path      GroupIP
-----
i     Intra-Ad      192.0.2.1         100        0
      64500:201    -                 -
      192.0.2.1    -
      No As-Path   -
i     Source-Join   -                 100        0
      64500:203    64500             -
      192.0.2.1    2001:DB8:3::1
      No As-Path   FF3E::8000:1
-----
Routes : 2
=====
```

PE-1 output for the received routes for the mvpn-ipv4 address family.

```
A:PE-1# show router bgp neighbor 192.0.2.3 received-routes mvpn-ipv4
=====
BGP MVPN-IPv4 Routes
=====
```

Flag	RouteType	OriginatorIP	LocalPref	MED
	RD	SourceAS		Label
	Nexthop	SourceIP		
	As-Path	GroupIP		
u*>i	Intra-Ad	192.0.2.3	100	0
	64500:203	-		-
	192.0.2.3	-		
	No As-Path	-		
u*>i	Source-Ad	-	100	0
	64500:203	-		-
	192.0.2.3	172.16.3.1		
	No As-Path	232.0.0.1		

```
-----
Routes : 2
=====
```

PE-1 output for the received routes for the mvpn-ipv6 address family.

```
A:PE-1# show router bgp neighbor 192.0.2.3 received-routes mvpn-ipv6
=====
BGP MVPN-IPv6 Routes
=====
```

Flag	RouteType	OriginatorIP	LocalPref	MED
	RD	SourceAS		Label
	Nexthop	SourceIP		
	As-Path	GroupIP		
u*>i	Intra-Ad	192.0.2.3	100	0
	64500:203	-		-
	192.0.2.3	-		
	No As-Path	-		
u*>i	Source-Ad	-	100	0
	64500:203	-		-
	192.0.2.3	2001:DB8:3::1		
	No As-Path	FF3E::8000:1		

```
-----
Routes : 2
=====
```


LDP Verification and Debugging

When BGP intra-AD messages are exchanged every PE starts to build a multicast tunnel based on the following criteria:

PE nodes which are configured as **sender-only** do not distribute mLDP Forward Equivalence Classes (FECs) to remote PEs for this MVPN.

PE nodes which configured as receiver-only do not include the PMSI part for intra-AD messages and remote PEs do not send mLDP FECs for this MVPN.

LDP bindings can be verified using the following command:

show router ldp bindings fec-type p2mp

PE-1 (192.0.2.1) has one ingress FEC and one egress FEC due to the fact that PE-1 has the default **mdt-type sender-receiver**.

```
*A:PE-1# show router ldp bindings fec-type p2mp
=====
LDP LSR ID: 192.0.2.1
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
       WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
=====
LDP Generic P2MP Bindings
=====
P2MP-Id      RootAddr
Interface    Peer          IngLbl    EgrLbl  EgrIntf/  EgrNextHop
                               LspId
-----
8193         192.0.2.1
73734        192.0.2.2      --      256033  3/1/1     192.168.12.1

8193         192.0.2.3
73735        192.0.2.3     261935U   --      --        --

-----
No. of Generic P2MP Bindings: 2
```

PE-2 (192.0.2.2) has two ingress FECs due to the fact that PE-2 has **mdt-type receiver-only**.

```
A:PE-2# show router ldp bindings fec-type p2mp
=====
LDP LSR ID: 192.0.2.2
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
       WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
=====
LDP Generic P2MP Bindings
=====
P2MP-Id      RootAddr
Interface    Peer          IngLbl    EgrLbl  EgrIntf/  EgrNextHop
-----
```

mLDP-Based MVPN Configuration

```

-----
                                     LspId
-----
8193          192.0.2.1
73733        192.0.2.1      256033U    --    --    --

8193          192.0.2.3
73732        192.0.2.3      256034U    --    --    --

-----
No. of Generic P2MP Bindings: 2

```

PE-3 (192.0.2.3) has two egress FECs due to the fact that PE-3 has **mdt-type sender-only**.

```

*A:PE-3# show router ldp bindings fec-type p2mp
=====
LDP LSR ID: 192.0.2.3
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
=====
LDP Generic P2MP Bindings
=====
P2MP-Id      RootAddr
Interface    Peer          IngLbl    EgrLbl  EgrIntf/  EgrNextHop
                                     LspId
-----
8193          192.0.2.3
73752        192.0.2.1      --        261935  3/1/4     192.168.13.0

8193          192.0.2.3
73752        192.0.2.2      --        256034  1/1/1     192.168.23.0

-----
No. of Generic P2MP Bindings: 2

```

Multicast Stream Verification

Status of multicast group/stream can be verified using the following command

show router <sid> pim group detail [ipv6]

There is IPv4 receiver connected to PE-1. I-PMSI is used as incoming interface and physical interface where receiver is connected is used as outgoing.

```
A:PE-1# show router 2 pim group detail
=====
PIM Source Group ipv4
=====
Group Address      : 232.0.0.1
Source Address     : 172.16.3.1
RP Address         : 0
Advt Router        : 192.0.2.3
Flags              :                               Type           : (S,G)
MRIB Next Hop      : 192.0.2.3
<snip>
Rpf Neighbor       : 192.0.2.3
Incoming Intf      : mpls-if-73735
Outgoing Intf List : int-mcast-receiver

Curr Fwding Rate   : 85614.0 kbps
<snip>
```

There is IPv4 receiver connected to PE-2. I-PMSI is used as incoming interface and physical interface where receiver is connected is used as outgoing.

```
A:PE-2# show router 2 pim group detail
=====
PIM Source Group ipv4
=====
Group Address      : 232.0.0.1
Source Address     : 172.16.3.1
RP Address         : 0
Advt Router        : 192.0.2.3
Flags              :                               Type           : (S,G)
MRIB Next Hop      : 192.0.2.3
<snip>
Rpf Neighbor       : 192.0.2.3
Incoming Intf      : mpls-if-73732
Outgoing Intf List : int-mcast-receiver

Curr Fwding Rate   : 85615.1 kbps
<snip>
```

There is IPv4 sender connected to PE-3. I-PMSI is used as outgoing interface and physical interface where sender is connected is used as incoming.

```
*A:PE-3# show router 2 pim group detail
=====
PIM Source Group ipv4
=====
Group Address      : 232.0.0.1
Source Address     : 172.16.3.1
RP Address         : 0
Advt Router        : 192.0.2.3
Flags              :                               Type           : (S,G)
MRIB Next Hop      : 172.16.3.1
<snip>
Rpf Neighbor       : 172.16.3.1
Incoming Intf      : int-mcast-source
Outgoing Intf List : mpls-if-73752

Curr Fwding Rate   : 85638.1 kbps
<snip>
```

Similar behavior is observed for IPv6 multicast.

There is an IPv6 receiver connected to PE-1. An I-PMSI is used as the incoming interface and the physical interface where the receiver is connected is used as the outgoing interface.

```
A:PE-1# show router 2 pim group detail ipv6
=====
PIM Source Group ipv6
=====
Group Address      : FF3E::8000:1
Source Address     : 2001:DB8:3::1
RP Address         : 0
Advt Router        : 192.0.2.3
Flags              :                               Type           : (S,G)
MRIB Next Hop      : 192.0.2.3
<snip>
Rpf Neighbor       : 192.0.2.3
Incoming Intf      : mpls-if-73735
Outgoing Intf List : int-mcast-receiver

Curr Fwding Rate   : 2140.5 kbps
<snip>
```

There is an IPv6 receiver connected to PE-2. An I-PMSI is used as the incoming interface and the physical interface where the receiver is connected is used as the outgoing interface.

```
A:PE-2# show router 2 pim group detail ipv6
=====
PIM Source Group ipv6
=====
Group Address      : FF3E::8000:1
Source Address     : 2001:DB8:3::1
```

Multicast VPN: Sender-Only, Receiver-Only

```
RP Address      : 0
Advt Router     : 192.0.2.3
Flags           :                                     Type      : (S,G)
MRIB Next Hop   : 192.0.2.3
<snip>
Rpf Neighbor    : 192.0.2.3
Incoming Intf   : mpls-if-73732
Outgoing Intf List : int-mcast-receiver

Curr Fwding Rate : 2140.5 kbps
<snip>
```

There is an IPv6 sender connected to PE-3. An I-PMSI is used as the outgoing interface and the physical interface where the sender is connected is used as the incoming interface.

```
*A:PE-3# show router 2 pim group detail ipv6
=====
PIM Source Group ipv6
=====
Group Address   : FF3E::8000:1
Source Address  : 2001:DB8:3::1
RP Address      : 0
Advt Router     : 192.0.2.3
Flags           :                                     Type      : (S,G)
MRIB Next Hop   : 2001:DB8:3::1
<snip>
Rpf Neighbor    : 2001:DB8:3::1
Incoming Intf   : int-mcast-source
Outgoing Intf List : mpls-if-73752

Curr Fwding Rate : 2140.5 kbps
<snip>
```

Conclusion

The sender-only/receiver-only feature provides significant signaling optimization in the core network for RSVP and LDP protocols and is recommended to be used when such functionality is required. The following are required before implementing this feature in the network:

- MDT-types **sender-only**, **receiver-only** and **sender-receiver** are enabled per MVPN.
- The default mdt-type is **sender-receiver** mode for backward compatibility.
- This is purely a control plane feature and there are no hardware dependencies, except for requiring chassis mode C or later.
- Draft Rosen or MDT-SAFI based MVPNs are not supported.
- IPv4 and IPv6 C-signaling are supported.
- mLDP and RSVP demonstrate slightly different behavior due to the nature of each protocol.
- mLDP provides a better optimization than RSVP in all cases, as mLDP does not initiate LSPs to sender-only routers.

Multicast VPN: Use of Wildcard Selective PMSI

In This Chapter

This section provides information about Multicast VPNs: Use of Wildcard Selective PMSI.

Topics in this section include:

- [Applicability on page 1612](#)
- [Overview on page 1613](#)
- [Configuration on page 1617](#)
- [Configuration on page 1617](#)

Applicability

This chapter is applicable to 7750 SR-7/12, 7750 SR-c4/12, and 7950 XRS. The functionality can also be configured on any supported IOM (IOM2, IOM3-XP, or IOM4) or IMM card. The example is also applicable to the 7450 ESS-7/12 with IOM3-XP or IMM. Chassis mode C or D is required for MPLS provider tunnels using multicast LDP (mLDP) or point-to-multipoint (P2MP) RSVP-TE label switched paths (LSPs). Release 12.0.R4 or later is required for route reflectors (RRs) peering with Multicast Virtual Private Network (MVPN) PEs.

The configuration was tested on release 13.0.R4, using Next Generation MVPN techniques. Provider Multicast Service Interfaces (PMSIs) are signaled using mLDP. PE MVPN auto-discovery uses BGP MVPN IPv4 network layer routing.

Knowledge of Multi-Protocol BGP (MP-BGP), RFC 4364, *BGP/MPLS IP Virtual Private Networks (VPNs)*, RFC 6513, *Multicast in MPLS/BGP IP VPNs*/RFC 6514, *BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPNs*, and RFC 6625, *Wildcards in Multicast VPN Auto-Discovery Routes*, is assumed throughout this section.

Overview

Consider a service provider core network used to deliver multicast services to a number of receiver PEs using Next Generation MVPN techniques, as defined in RFC6513/6514, where multicast traffic is forwarded between PEs across a mesh of provider tunnels.

The provider tunnel used is the default Inclusive PMSI (I-PMSI) that is instantiated between all source and receiver PEs. This results in a full mesh of provider tunnels between all PEs in the MVPN. In a large network, this can result in an inefficient use of bandwidth because multicast traffic is forwarded to all PEs regardless of whether the PE has an interested receiver. Some of the mesh scaling issues can be mitigated by using source-only/destination-only configuration on PEs. However, this technique requires additional configuration and is not fully optimal when mLDP is used in the core.

To address the above limitation, wildcard Selective PMSI (S-PMSI) has been developed, as per RFC6625. In the standard customer signaling notation of (C-S,C-G), this becomes (C-*,C-*). Using methods defined in RFC6625, it is possible to use a (C-*,C-*) S-PMSI as the default tunnel, where the receiver PE can join the S-PMSI by mapping the channel join to a wildcard channel group. Multiple channels can be transported by the wildcard (C-*,C-*) S-PMSI, where an S-PMSI auto-discovery route is advertised with an empty channel group and source address:

1. Bandwidth savings can be achieved by the delivery of multicast channels on S-PMSIs, because traffic is not forwarded to PEs that have no interested receivers.
2. Control plane savings can be achieved by eliminating the need for the full tunnel mesh between all PES. The wildcard S-PMSI is only signaled on PEs containing attached upstream multicast sources, for which the PE is resolved as an upstream multicast hop (UMH) within the MVPN.

[Figure 234](#) shows the concept of an MVPN.

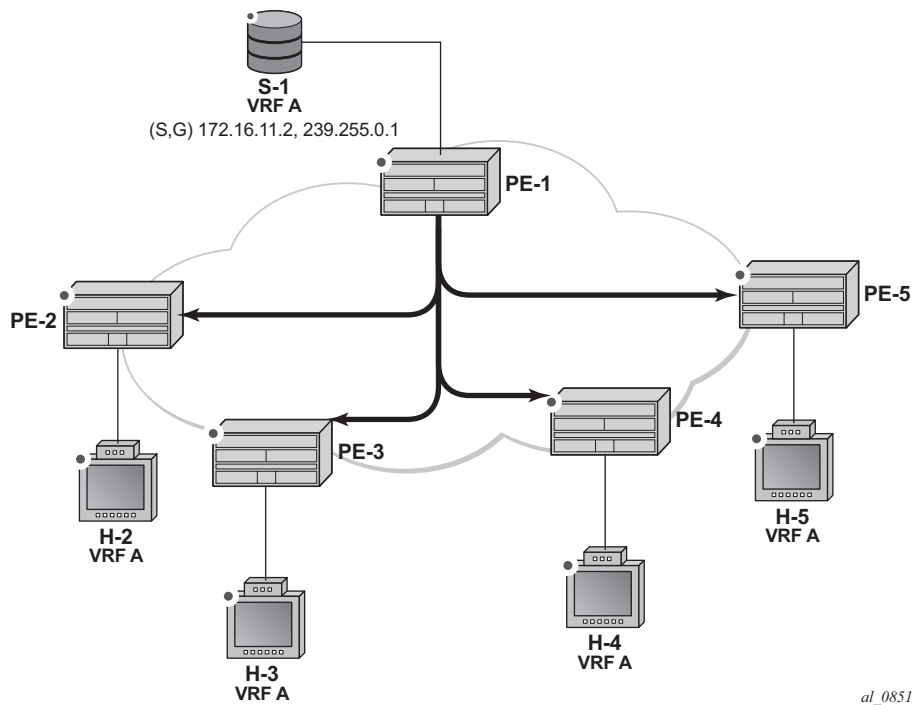


Figure 234: Multicast VPN

In [Figure 234](#), PE-1 has a directly connected multicast source (S-1). For clarity, consider this MVPN as a single multicast group. PE-1 is configured as a sender PE because it is the PE closest to the source. PE-2, PE-3, PE-4, and PE-5 are configured as receiver-only PEs because they have connected receiver hosts H-2, H-3, H-4, and H-5, respectively. Hosts H-2 to H-5 connected to receiver PEs can receive multicast channels from the source, S-1, connected to the source PE, PE-1, within the same Virtual Private Routed Network (VPRN).

Within the provider network, multicast traffic is delivered from the source PE to the receiver PE across a PMSI tunnel. This tunnel is, in this case, a P2MP LSP, with its root at PE-1 and with a leaf at each of the receiver PEs. Any traffic that is forwarded into the tunnel interface is replicated so that a single copy of a multicast stream is received at all PEs.

The PMSI tunnel is created after each PE has declared themselves as a member of the MVPN using BGP MVPN auto-discovery techniques.

There are two choices of PMSI:

- An I-PMSI, which is created on each PE within the MVPN, with a root at each PE and a leaf at all other PEs that are members of the MVPN. The I-PMSI is the default tunnel for all multicast traffic carried between sender PE and receiver PEs. When at least one

receiver PE has a host interested in becoming a member of a multicast group, traffic for that group is delivered to all PEs via the I-PMSI, regardless of whether they have an interested host. Receiver PEs with no such interested host then drop the traffic.

- An S-PMSI, which is created to carry multicast traffic to the subset of receiver PEs that have connected hosts interested in receiving multicast traffic. This can be for a specific group, so that one S-PMSI carries traffic for one multicast group, denoted as (C-S,C-G) or (C-*,C-G). The S-PMSI can also be signaled to carry traffic for multiple multicast groups, denoted using a wildcard: (C-*,C-*) or (C-S,C-*). The wildcard S-PMSI can be signaled in place of the I-PMSI, so that all traffic can be carried on the S-PMSI by default. In this case, no I-PMSI is signaled.

In the case of an I-PMSI, the tunnel type is included in the BGP auto-discovery intra-AD route originated and advertised to all other PEs within the VPRN.

If a wildcard S-PMSI is to be used and no I-PMSI tunnel is to be signaled, then an intra-AD route update for I-PMSI is advertised with no tunnel type attribute included. In addition, the source PE will originate an additional S-PMSI auto-discovery route containing no source-encoding wildcard information.

[Table 11](#) shows the S-PMSI MVPN BGP Network Layer Reachability Information (NLRI) advertisement.

Table 11: S-PMSI Auto-Discovery BGP NLRI

Route Distinguisher (8 octets)
Multicast Source Length (1 octet)
Multicast Source (variable)
Multicast Group Length (1 octet)
Multicast Group (variable)
Originating Router IP Address

To signal the S-PMSI as wildcard (C-*,C-*) S-PMSI, the multicast source length and multicast group length fields are encoded with the value of zero (0), representing (C-*,C-*) wildcard.

The objectives are to:

- Configure multicast in a VPRN on PE-1 to PE-5 using mLDP as the tunnel signaling method.
- Connect multicast sources to the sender PE-1.

- Create receiver hosts that can receive multicast traffic from the source, and to observe the effect on the provider network.

The following configuration tasks should be completed as a prerequisite:

- Full mesh IS-IS or OSPF between each of the PE routers and the RR.
- Link-layer LDP between all PEs.
- mLDP used as the provider tunnel signaling protocol. This is enabled by default when link-layer LDP is enabled.

Note: RSVP-TE is also supported as a provider tunnel signaling mechanism and could be used.

Configuration

The test topology is shown in [Figure 235](#), containing five PE routers. P-6 acts as an RR.

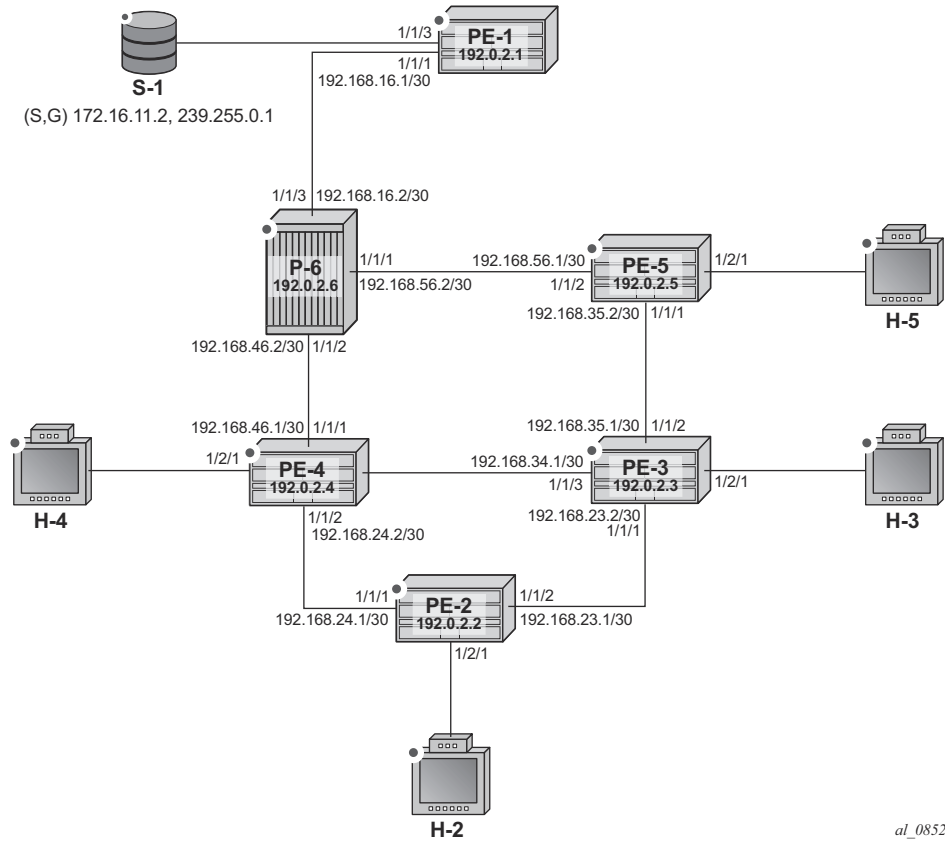


Figure 235: Schematic Topology

Global BGP Configuration

The first step is to configure an IBGP session between each of the PEs and the RR (PE-6) shown in [Figure 235](#). The address families negotiated between the IBGP peers are vpn-ipv4 (unicast routing) and mvpn-ipv4 (multicast routing).

The configuration for PE1 is:

```
configure router
  bgp
    group INTERNAL
      family vpn-ipv4 mvpn-ipv4
      type internal
      neighbor 192.0.2.6
    exit
  exit
```

The configuration for the other PE nodes is exactly the same.

The configuration for the RR at P-6 is:

```
configure router
  bgp
    cluster 0.0.0.1
    group "RR_CLIENTS"
      family vpn-ipv4 mvpn-ipv4
      type internal
      neighbor 192.0.2.1
    exit
    neighbor 192.0.2.2
    exit
    neighbor 192.0.2.3
    exit
    neighbor 192.0.2.4
    exit
    neighbor 192.0.2.5
    exit
  exit
```

On PE-1, verify that the BGP session with RR at P-6 is established with address families vpn-ipv4 and mvpn-ipv4 capabilities negotiated:

```
*A:PE-1# show router bgp summary
=====
BGP Router ID:192.0.2.1      AS:65545      Local AS:65545
=====
BGP Admin State      : Up      BGP Oper State      : Up
Total Peer Groups    : 1        Total Peers          : 1
Total BGP Paths       : 29      Total Path Memory    : 5724
Total IPv4 Remote Rts : 0        Total IPv4 Rem. Active Rts : 0
Total McIPv4 Remote Rts : 0      Total McIPv4 Rem. Active Rts: 0
Total McIPv6 Remote Rts : 0      Total McIPv6 Rem. Active Rts: 0
```

Multicast VPN: Use of Wildcard Selective PMSI

```

Total IPv6 Remote Rts      : 0      Total IPv6 Rem. Active Rts : 0
Total IPv4 Backup Rts      : 0      Total IPv6 Backup Rts      : 0

Total Supressed Rts       : 0      Total Hist. Rts           : 0
Total Decay Rts           : 0

Total VPN Peer Groups     : 0      Total VPN Peers           : 0
Total VPN Local Rts       : 6      Total VPN-IPv4 Rem. Act. Rts: 4
Total VPN-IPv4 Rem. Rts   : 6      Total VPN-IPv6 Rem. Act. Rts: 0
Total VPN-IPv6 Rem. Rts   : 0      Total VPN-IPv4 Bkup Rts   : 0
Total VPN-IPv4 Bkup Rts   : 0      Total VPN-IPv6 Bkup Rts   : 0

Total VPN Supp. Rts       : 0      Total VPN Hist. Rts       : 0
Total VPN Decay Rts       : 0

Total L2-VPN Rem. Rts     : 0      Total L2VPN Rem. Act. Rts : 0
Total MVPN-IPv4 Rem Rts   : 6      Total MVPN-IPv4 Rem Act Rts: 4
Total MDT-SAFI Rem Rts    : 0      Total MDT-SAFI Rem Act Rts : 0
Total MSPW Rem Rts        : 0      Total MSPW Rem Act Rts     : 0
Total RouteTgt Rem Rts    : 0      Total RouteTgt Rem Act Rts : 0
Total McVpnIPv4 Rem Rts   : 0      Total McVpnIPv4 Rem Act Rts: 0
Total MVPN-IPv6 Rem Rts   : 0      Total MVPN-IPv6 Rem Act Rts: 0
Total EVPN Rem Rts        : 0      Total EVPN Rem Act Rts     : 0
Total FlowIpv4 Rem Rts    : 0      Total FlowIpv4 Rem Act Rts : 0
Total FlowIpv6 Rem Rts    : 0      Total FlowIpv6 Rem Act Rts : 0

```

```

=====
BGP Summary
=====
Neighbor
Description
          AS PktRcvd InQ  Up/Down  State|Rcv/Act/Sent (Addr Family)
          PktSent OutQ
-----
192.0.2.6
          65545      47    0 00h14m20s 6/4/2 (VpnIPv4)
                   37    0          6/4/2 (MvpnIPv4)
-----
*A:PE-1#

```

The same command can be used on the other PEs to verify their BGP sessions to the RR.

Configuring VPRN on PEs

The following outputs show the VPRN configurations for each PE. The specific MVPN configuration is shown later.

PE-1

The VPRN configuration for PE-1 is:

```
configure service
  vprn 1 customer 1 create
    route-distinguisher 65545:1
    auto-bind-tunnel
    resolution-filter
    ldp
    exit
    resolution filter
  exit
  vrf-target target:65545:1
  interface "int-PE-1-S-1" create
    address 172.16.11.1/24
    sap 1/1/3 create
    exit
  exit
  interface "rp" create
    address 10.0.0.1/32
    loopback
  exit
  pim
    apply-to all
    rp
      static
        address 10.0.0.1
        group-prefix 239.255.0.0/16
      exit
    exit
  exit
  no shutdown
exit
no shutdown
```

There is a single interface toward S-1 from which the multicast group is transmitted.

If the customer signaling uses PIM ASM, a customer Rendezvous Point (RP) is required.

A loopback interface called “rp” acts as the RP for all group prefixes in the 239.255.0.0/16 range. This will be the RP for all groups.

MVPN configuration enables BGP as both the auto-discovery mechanism and the customer multicast signaling protocol across the VPRN. The provider tunnel between PEs within the MVPN is signaled using mLDP.

PE-2

PE-2 contains an attached receiver, so a single interface is configured to accommodate this. The RP is configured as a static RP:

```
configure service
  vprn 1 customer 1 create
    route-distinguisher 65545:1
    auto-bind-tunnel
    resolution-filter
      ldp
    exit
    resolution filter
  exit
  vrf-target target:65545:1
  interface "int-PE-2-H-2" create
    address 172.16.22.1/24
    sap 1/2/1 create
  exit
  exit
  igmp
    interface "int-PE-2-H-2"
      no shutdown
    exit
    no shutdown
  exit
  pim
    apply-to all
    rp
      static
        address 10.0.0.1
        group-prefix 239.255.0.0/16
      exit
    exit
    no shutdown
  exit
  no shutdown
```

PE-3

PE-3 also contains an attached receiver:

```
configure service
  vprn 1 customer 1 create
    route-distinguisher 65545:1
    auto-bind-tunnel
    resolution-filter
      ldp
    exit
    resolution filter
  exit
  vrf-target target:65545:1
  interface "int-PE-3-H-3" create
    address 172.16.33.1/24
    sap 1/2/1 create
```

MVPN Configuration for PEs

```
        exit
    exit
    igmp
        interface "int-PE-3-H-3"
            no shutdown
        exit
        no shutdown
    exit
    pim
        apply-to all
        rp
            static
                address 10.0.0.1
                group-prefix 239.255.0.0/16
            exit
        exit
    exit
    no shutdown
exit
no shutdown
```

The configuration for PE-4 and PE-5 is similar.

MVPN Configuration for PEs

The provider-tunnel inclusive configuration specifies that a wildcard S-PMSI will be used instead of an I-PMSI as the default PMSI. The MVPN configuration for all PEs is:

```
configure service
    vprn 1
        mvpn
            auto-discovery default
            c-mcast-signaling bgp
            provider-tunnel
                inclusive
                mldp
                    no shutdown
                exit
                wildcard-spmsi
            exit
        exit
        vrf-target unicast
    exit
exit
```

The keyword **wildcard-spmsi** reduces the number of PMSIs signaled. If there are no group sources on the receiver PEs, there will be no S-PMSI signaled. This has an effect similar to configuring the receiver PEs as MDT-type receiver-only.

Provider Tunnel Signaling

Each PE originates BGP MVPN intra-AD routes to determine membership of an MVPN.

The provider tunnels constructed between the PEs within the VPRN are signaled on receipt of an intra-AD route update from other PEs. The intra-AD update message contains details of the originator, along with the VRF route-target extended community. If an I-PMSI is to be signaled, a PMSI tunnel attribute is included that determines the tunnel type, such as LDP P2MP LSP. PEs that receive this intra-AD update will import the route into the MVPN, then signal a P2MP LDP label map toward the originator, which is the root of the LDP P2MP LSP.

However, if a wildcard S-PMSI is to be used as the default PMSI, no PMSI tunnel attribute is included within the intra-AD update.

The following output shows a BGP update originated by PE-1, and received by PE-2:

```
*A:PE-2# show router bgp routes mvpn-ipv4 type intra-ad rd 65545:1 detail originator-ip
192.0.2.1
=====
BGP Router ID:192.0.2.2          AS:65545          Local AS:65545
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked, x - stale, > - best, b - backup, p - purge
Origin codes  : i - IGP, e - EGP, ? - incomplete
=====
BGP MVPN-IPv4 Routes
=====
Original Attributes

Route Type      : Intra-Ad
Route Dist.     : 65545:1
Originator IP   : 192.0.2.1
Nexthop         : 192.0.2.1
From            : 192.0.2.6
Res. Nexthop    : 0.0.0.0
Local Pref.     : 100
Aggregator AS   : None
Atomic Aggr.    : Not Atomic
AIGP Metric     : None
Connector       : None
Community       : no-export target:65545:1
Cluster         : 0.0.0.1
Originator Id   : 192.0.2.1          Peer Router Id : 192.0.2.6
Flags           : Used Valid Best IGP
Route Source    : Internal
AS-Path         : No As-Path

Interface Name  : NotAvailable
Aggregator      : None
MED             : 0
```

Provider Tunnel Signaling

```
Route Tag      : 0
Neighbor-AS    : N/A
Orig Validation: N/A
Source Class   : 0                      Dest Class    : 0
Add Paths Send : Default
Last Modified  : 00h22m08s
VPRN Imported  : 1
---snip---
```

Note: There is no PMSI tunnel attribute included, and the route is imported into the correct VPRN (VPRN 1).

The intra-AD originated by PE-2 is:

```
*A:PE-1# show router bgp routes mvpn-ipv4 type intra-ad rd 65545:1 originator-ip 192.0.2.2
hunt
```

```
=====
BGP Router ID:192.0.2.1      AS:65545      Local AS:65545
=====
```

```
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked, x - stale, > - best, b - backup, p - purge
Origin codes  : i - IGP, e - EGP, ? - incomplete
=====
```

```
BGP MVPN-IPv4 Routes
=====
```

```
-----
RIB In Entries
-----
```

```
Route Type      : Intra-Ad
Route Dist.     : 65545:1
Originator IP   : 192.0.2.2
Nexthop         : 192.0.2.2
From            : 192.0.2.6
Res. Nexthop    : 0.0.0.0
Local Pref.     : 100                      Interface Name : NotAvailable
Aggregator AS   : None                     Aggregator      : None
Atomic Aggr.    : Not Atomic               MED             : 0
AIGP Metric     : None
Connector       : None
Community       : no-export target:65545:1
Cluster         : 0.0.0.1
Originator Id   : 192.0.2.2                Peer Router Id  : 192.0.2.6
Flags           : Used Valid Best IGP
Route Source    : Internal
AS-Path         : No As-Path
Route Tag       : 0
Neighbor-AS     : N/A
Orig Validation: N/A
Source Class    : 0                      Dest Class      : 0
Add Paths Send  : Default
Last Modified   : 00h23m49s
VPRN Imported   : 1
```

```
-----
RIB Out Entries
-----
```

```
-----
Routes : 1
=====
```

```
*A:PE-1#
```

Note: This output also contains no PMSI tunnel attribute: PE-2 has no group source and there is no S-PMSI signaled. All other receiver PEs will originate a similar intra-AD route.

The following output shows all intra-AD routes originated by the PEs within the VPRN, as received by PE-1:

```
*A:PE-1# show router bgp routes mvpn-ipv4 type intra-ad rd 65545:1
=====
BGP Router ID:192.0.2.1      AS:65545      Local AS:65545
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked, x - stale, > - best, b - backup, p - purge
Origin codes : i - IGP, e - EGP, ? - incomplete
=====
BGP MVPN-IPv4 Routes
=====
```

Flag	RouteType RD Nexthop As-Path	OriginatorIP SourceAS SourceIP GroupIP	LocalPref	MED Label
i	Intra-Ad 65545:1 192.0.2.1 No As-Path	192.0.2.1 - - -	100	0 - - -
u*>i	Intra-Ad 65545:1 192.0.2.2 No As-Path	192.0.2.2 - - -	100	0 - - -
u*>i	Intra-Ad 65545:1 192.0.2.3 No As-Path	192.0.2.3 - - -	100	0 - - -
u*>i	Intra-Ad 65545:1 192.0.2.4 No As-Path	192.0.2.4 - - -	100	0 - - -
u*>i	Intra-Ad 65545:1 192.0.2.5 No As-Path	192.0.2.5 - - -	100	0 - - -

```
-----
Routes : 5
=====
*A:PE-1#
```

Instead of an I-PMSI being signaled, an S-PMSI AD route update is advertised by PE-1 to all receiver PEs within the MVPN. The NLRI encoding has a zero length field for both source and group addresses, so is seen to represent multicast group (C-*,C-*). This is wildcard nomenclature for both source and group addresses.

The BGP route as advertised by PE-1:

```
*A:PE-1# show router bgp routes mvpn-ipv4 type spmsi-ad rd 65545:1 hunt
=====
BGP Router ID:192.0.2.1          AS:65545          Local AS:65545
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked, x - stale, > - best, b - backup, p - purge
Origin codes  : i - IGP, e - EGP, ? - incomplete
=====
BGP MVPN-IPv4 Routes
=====
-----
RIB In Entries
-----
---snip---

-----
RIB Out Entries
-----
Route Type      : Spmsi-Ad
Route Dist.     : 65545:1
Originator IP   : 192.0.2.1
Source IP       : 0.0.0.0
Group IP        : 0.0.0.0
Nexthop         : 192.0.2.1
To              : 192.0.2.6
Res. Nexthop    : n/a
Local Pref.     : 100
Aggregator AS   : None
Atomic Aggr.    : Not Atomic
AIGP Metric     : None
Connector       : None
Community       : target:65545:1
Cluster         : No Cluster Members
Originator Id   : None
Origin          : IGP
AS-Path         : No As-Path
Route Tag       : 0
Neighbor-AS     : N/A
Orig Validation : N/A
Source Class    : 0
Interface Name  : NotAvailable
Aggregator      : None
MED             : 0
Peer Router Id  : 192.0.2.6
Dest Class     : 0
-----
PMSI Tunnel Attribute :
Tunnel-type      : LDP P2MP LSP
MPLS Label       : 0
Root-Node        : 192.0.2.1
Flags            : Leaf not required
LSP-ID           : 8193
-----
---snip---
```

The source IP and group IP address fields are advertised as 0.0.0.0, and the tunnel type attribute is now included as an LDP P2MP LSP.

The following output shows the MVPN status at PE-1, with the I-PMSI tunnel name containing a wildcard S-PMSI denoted by (W):

```
*A:PE-1# show router 1 mvpn
=====
MVPN 1 configuration data
=====
signaling          : Bgp                auto-discovery    : Default
UMH Selection      : Highest-Ip         SA withdrawn      : Disabled
intersite-shared   : Enabled            Persist SA        : Disabled
vrf-import         : N/A
vrf-export         : N/A
vrf-target         : unicast
C-Mcast Import RT  : target:192.0.2.1:2

ipmsi              : ldp
i-pmsi P2MP AdmSt  : Up
i-pmsi Tunnel Name : mpls-if-73728(W)
Mdt-type           : sender-receiver

BSR signalling     : none
wildcard s-pmsi    : true
s-pmsi             : none
data-delay-interval: 3 seconds
enable-asm-mdt     : N/A

=====
*A:PE-1#
```

At this point, there is no interested host and no customer multicast flow (c-flow), so there is no S-PMSI LDP P2MP LSP signaled.

Data Transmission at Source PE

Multicast traffic for a particular group will be forwarded between the sender and receiver PE over a provider tunnel (PMSI). Because there is no default I-PMSI present, the receiver PE has to choose an S-PMSI to be used for forwarding, based on the NLRI contained within the S-PMSI AD routes.

The provider tunnel is signaled using a P2MP LDP label mapping message toward the root address signaled in the wildcard S-PMSI AD BGP update message. As previously shown, the update message is based on whether traffic is being forwarded on the shared or source-based shortest path tree.

When joining the shared tree, a c-multicast shared-join is sent toward the appropriate PE, which represents the UMH toward the RP. The UMH is chosen from the unicast route that represents the RP address. When joining the shortest path tree, a source-join c-multicast route is sent toward the UMH chosen from the unicast route that represents the actual source address. In both cases, the VPN-IPv4 unicast route advertises a VRF route import community that contains the system address as a next hop. Upon receipt of these updates, the UMH PE will forward traffic along the signaled wildcard S-PMSI.

Each S-PMSI is bound to one or more flows, as determined by the NLRI contained within the S-PMSI BGP update. The use of the wildcard within the BGP update to replace the c-source and c-group allows multiple flows to be bound to a single provider tunnel.

Note: Traffic will only be forwarded upon reception of a BGP MVPN source-join or shared-join BGP route at the sender PE.

Data Reception at Receiver PE

When the sender PE has originated an S-PMSI AD route update, each receiver PE will install the route into its VRF. The S-PMSIs installed are used to select an appropriate upstream multicast router for a c-flow when an attached receiver is interested in receiving traffic from that c-flow.

The receiver PE will receive a flow based on the best match of the incoming (C-S,C-G) or (C-*,C-G) IGMP/MLD or PIM join.

If an IGMP/MLD group membership query or PIM join is received by the receiver PE over an attachment circuit for a group, the contained (C-S,C-G) or (C-*,C-G) must match the (C-S,C-G) contained within any installed S-PMSI AD route. In the case of the wildcard S-PMSI being the only installed NLRI, this will be a match; that is, incoming (C-*,C-G) or (C-S,C-G) will match the S-PMSI (C-*,C-*).

In this example, the c-group flow is 239.255.0.1.

Traffic Flow

A traffic stream representing a c-flow is enabled on PE-1: group address 239.255.0.1 with source address of 172.16.11.2. The RP for this group is found locally on PE-1. The outgoing interface list is empty:

```
*A:PE-1# show router 1 pim group detail
=====
PIM Source Group ipv4
=====
Group Address      : 239.255.0.1
Source Address     : 172.16.11.2
RP Address         : 10.0.0.1
Advt Router       : 192.0.2.1
Flags              :
Type               : (S,G)
MRIB Next Hop     : 172.16.11.2
MRIB Src Flags     : direct
Keepalive Timer Exp: 0d 00:03:03
Up Time           : 0d 00:49:27
Resolved By       : rtable-u

Up JP State       : Not Joined
Up JP Rpt         : Not Joined StarG
Up JP Expiry      : 0d 00:00:00
Up JP Rpt Override: 0d 00:00:00

Register State    : Pruned
Register Stop Exp : 0d 00:00:44
Reg From Anycast RP: No

Rpf Neighbor      : 172.16.11.2
Incoming Intf     : int-PE-1-S-1
Outgoing Intf List:
Outgoing Sap List :
Outgoing Host List:

Curr Fwding Rate  : 19717.1 kbps
Forwarded Packets : 158271505
Forwarded Octets  : 7280489230
Spt threshold     : 0 kbps
Admin bandwidth   : 1 kbps
Discarded Packets : 0
RPF Mismatches    : 0
ECMP opt threshold: 7
-----
Groups : 1
=====
*A:PE-1#
```

A host connected to PE-2 sends a (C-*,C-G) IGMP v2 group membership query for group 239.255.0.1.

PE-2 sends a BGP MVPN shared-join route update toward PE-1, where the RP address of the group 10.0.0.1 is found.

The following debug output shows the shared-join BGP route update transmitted by PE-2:

```
*A:PE-2# show log log-id 2
---snip---

1 2015/09/10 07:44:16.19 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.6
"Peer 1: 192.0.2.6: UPDATE
Peer 1: 192.0.2.6 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 69
  Flag: 0x90 Type: 14 Len: 33 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.2
    Type: Shared-Join Len:22 RD: 65545:1 SrcAS: 65545 Src: 10.0.0.1 Grp: 239.255.0.1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:192.0.2.1:2
"
```

Upon receipt of the shared-join, traffic flows on the shared tree toward the receiver PE. This will flow on the default wildcard S-PMSI, as shown in the outgoing interface list:

```
*A:PE-1# show router 1 pim group 239.255.0.1 type starg detail
=====
PIM Source Group ipv4
=====
Group Address      : 239.255.0.1
Source Address     : *
RP Address         : 10.0.0.1
Advt Router       : 192.0.2.1
Flags              :                               Type              : (*,G)
MRIB Next Hop     :
MRIB Src Flags    : self
Keepalive Timer   : Not Running
Up Time           : 0d 00:03:49      Resolved By             : rtable-u

Up JP State       : Joined           Up JP Expiry             : 0d 00:00:11
Up JP Rpt        : Not Joined StarG  Up JP Rpt Override      : 0d 00:00:00

Rpf Neighbor      :
Incoming Intf     :
Outgoing Intf List : mpls-if-73728(W)

Curr Fwding Rate  : 0.0 kbps
Forwarded Packets : 0                Discarded Packets       : 0
Forwarded Octets  : 0                RPF Mismatches          : 0
Spt threshold     : 0 kbps           ECMP opt threshold     : 7
Admin bandwidth   : 1 kbps
=====
Groups : 1
=====
*A:PE-1#
```

When traffic is received on the shared tree by PE-2, the source address is learned, so a source-join BGP route update is sent toward the UMH PE, which contains the source address of 172.16.11.2. The UMH is chosen from the unicast route-table using a lookup for the best route matching the source address.

The following debug output shows the BGP source-join route update toward the source of group 239.255.0.1, as transmitted by PE-2:

```
*A:PE-2# show log log-id 2
---snip---

3 2015/09/10 07:44:45.19 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.6
"Peer 1: 192.0.2.6: UPDATE
Peer 1: 192.0.2.6 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 69
  Flag: 0x90 Type: 14 Len: 33 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.2
    Type: Source-Join Len:22 RD: 65545:1 SrcAS: 65545 Src: 172.16.11.2 Grp: 239.255.0.1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:192.0.2.1:2
"
```

The c-flow toward host H-2 flows along the shortest path tree, and on PE-1 the outgoing interface list is populated with the wildcard S-PMSI:

```
*A:PE-1# show router 1 pim group detail 239.255.0.1 source 172.16.11.2
=====
PIM Source Group ipv4
=====
Group Address      : 239.255.0.1
Source Address     : 172.16.11.2
RP Address         : 10.0.0.1
Advt Router       : 192.0.2.1
Flags              : spt, rpt-prn-des   Type              : (S,G)
MRIB Next Hop     : 172.16.11.2
MRIB Src Flags    : direct
Keepalive Timer Exp: 0d 00:02:01
Up Time           : 0d 01:10:28          Resolved By       : rtable-u

Up JP State       : Joined               Up JP Expiry       : 0d 00:00:00
Up JP Rpt        : Pruned               Up JP Rpt Override : 0d 00:00:00

Register State    : Pruned               Register Stop Exp  : 0d 00:00:13
Reg From Anycast RP: No

Rpf Neighbor      : 172.16.11.2
Incoming Intf     : int-PE-1-S-1
Outgoing Intf List : mpls-if-73728(W)
```

Traffic Flow

```
Curr Fwding Rate   : 6915.5 kbps
Forwarded Packets  : 198813564      Discarded Packets : 0
Forwarded Octets   : 9145423944    RPF Mismatches    : 0
Spt threshold      : 0 kbps         ECMP opt threshold : 7
Admin bandwidth    : 1 kbps
```

```
-----
Groups : 1
=====
```

```
*A:PE-1#
```

The outgoing interface is the MPLS interface mpls-if-73728. This maps to a P2MP LDP label binding from which the p2mp-id can be derived:

```
*A:PE-1# show router ldp bindings active p2mp
```

```
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
              (IPv6 LSR ID ::[0])
=====
```

```
Legend: U - Label In Use,  N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
=====
```

```
LDP Generic IPv4 P2MP Bindings (Active)
=====
```

P2MP-Id	Interface		
RootAddr	Op	IngLbl	EgrLbl
EgrNH	EgrIf/LspId		
8193	73728		
192.0.2.1	Push	--	262137
192.168.16.2	1/1/1		

```
-----
No. of Generic IPv4 P2MP Active Bindings: 1
=====
```

```
---snip---
```

After the source-join is received, the sender PE will advertise a source-active AD BGP route to all PEs within the MVPN, to announce the presence of a (C-S,C-G) group. The following debug output shows the source-active AD route received on PE-2:

```
*A:PE-2# show log log-id 2
```

```
---snip---
```

```
4 2015/09/10 07:45:06.29 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.6
```

```
"Peer 1: 192.0.2.6: UPDATE
```

```
Peer 1: 192.0.2.6 - Received BGP UPDATE:
```

```
  Withdrawn Length = 0
```

```
  Total Path Attr Length = 79
```

```
  Flag: 0x90 Type: 14 Len: 29 Multiprotocol Reachable NLRI:
```

```
    Address Family MVPN_IPV4
```

```
    NextHop len 4 NextHop 192.0.2.1
```

```
    Type: Source-AD Len: 18 RD: 65545:1 Src: 172.16.11.2 Grp: 239.255.0.1
```

```
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
```

```
  Flag: 0x40 Type: 2 Len: 0 AS Path:
```

```
  Flag: 0x80 Type: 4 Len: 4 MED: 0
```

```
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
```

```

Flag: 0x80 Type: 9 Len: 4 Originator ID: 192.0.2.1
Flag: 0x80 Type: 10 Len: 4 Cluster ID:
      0.0.0.1
Flag: 0xc0 Type: 16 Len: 8 Extended Community:
      target:65545:1
"

```

The PIM status of the group on receiver PE-2 shows that the incoming interface is the wildcard S-PMSI originated on PE-1, as denoted by the (W):

```

*A:PE-2# show router 1 pim group 239.255.0.1 source 172.16.11.2 detail
=====
PIM Source Group ipv4
=====
Group Address       : 239.255.0.1
Source Address      : 172.16.11.2
RP Address          : 10.0.0.1
Advt Router         : 192.0.2.1
Flags               : spt                               Type           : (S,G)
MRIB Next Hop       : 192.0.2.1
MRIB Src Flags      : remote
Keepalive Timer Exp: 0d 00:02:21
Up Time             : 0d 00:12:43                       Resolved By        : rtable-u

Up JP State         : Joined                               Up JP Expiry       : 0d 00:00:16
Up JP Rpt           : Not Pruned                           Up JP Rpt Override : 0d 00:00:00

Register State      : No Info
Reg From Anycast RP: No

Rpf Neighbor        : 192.0.2.1
Incoming Intf      : mpls-if-73729(W)
Outgoing Intf List  : int-PE-2-H-2

Curr Fwding Rate    : 7683.3 kbps
Forwarded Packets   : 15692528                           Discarded Packets  : 0
Forwarded Octets    : 721856288                           RPF Mismatches     : 0
Spt threshold       : 0 kbps                               ECMP opt threshold : 7
Admin bandwidth     : 1 kbps
-----
Groups : 1
=====
*A:PE-2#

```

The S-PMSI is an LDP P2MP LSP. The LDP label binding for P2MP LSP-Id 8193 at PE-2 shows that the label operation is a label pop:

```

*A:PE-2# show router ldp bindings active p2mp p2mp-id 8193 root 192.0.2.1
=====
LDP Bindings (IPv4 LSR ID 192.0.2.2:0)
              (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use,  N - Label Not In Use,  W - Label Withdrawn
       WP - Label Withdraw Pending,  BU - Alternate For Fast Re-Route
=====
LDP Generic IPv4 P2MP Bindings (Active)

```

```

=====
P2MP-Id                               Interface
RootAddr                             Op           IngLbl    EgrLbl
EgrNH                               EgrIf/LspId
-----
8193                                 73729
192.0.2.1                           Pop           262136    --
--                                  --

-----
No. of Generic IPv4 P2MP Active Bindings: 1
=====
*A:PE-2#

```

PE-3 has no host joined to c-flow group 239.255.0.1. However, it contains the PIM state for this group due to the presence of the source-active AD route within the VRF. This route was received when the host connected to PE-2 joined the c-flow group.

The following output shows the source-active AD route within PE-3 for group 239.255.0.1 from source 172.16.11.2:

```

*A:PE-3# show router bgp routes mvpn-ipv4 type source-ad rd 65545:1
=====
BGP Router ID:192.0.2.3      AS:65545      Local AS:65545
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
               l - leaked, x - stale, > - best, b - backup, p - purge
Origin codes : i - IGP, e - EGP, ? - incomplete
=====
BGP MVPN-IPv4 Routes
=====
Flag  RouteType      OriginatorIP      LocalPref  MED
      RD           SourceAS          Label
      Nexthop      SourceIP
      As-Path       GroupIP
-----
u*>i Source-Ad      -                  100        0
      65545:1      -                  -
      192.0.2.1    172.16.11.2
      No As-Path   239.255.0.1
-----
Routes : 1
=====
*A:PE-3#

```

However, traffic is not received from the S-PMSI because there is no label binding for the LDP P2MP LSP. The following output shows that there is no label binding for the LSP Id 8193, which has its root on PE-1:

```

*A:PE-3# show router ldp bindings p2mp p2mp-id 8193 root 192.0.2.1
=====
LDP Bindings (IPv4 LSR ID 192.0.2.3:0)
              (IPv6 LSR ID ::[0])

```

```

=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
       WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
=====
LDP Generic IPv4 P2MP Bindings
=====
P2MP-Id
RootAddr                      Interface      IngLbl    EgrLbl
EgrNH                         EgrIf/LspId
Peer
-----
No Matching Entries Found
=====
*A:PE-3#

```

Receiver host H-3 sends an unsolicited IGMP v2 group membership query for group 239.255.0.1 toward PE-3. The following debug output shows the process.

IGMP Query Received on PE-3

```

*A:PE-3# show log log-id 2
---snip---

4 2015/09/10 08:25:30.13 UTC MINOR: DEBUG #2001 vprn1 IGMP[vprn1 inst 2]
"IGMP[vprn1 inst 2]: igmpIfSrcAdd
Adding i/f source entry for interface int-PE-3-H-3 [ifIndex 2] (*,239.255.0.1) to IGMP
fwdList Database, redir if N/A"

3 2015/09/10 08:25:30.13 UTC MINOR: DEBUG #2001 vprn1 IGMP[vprn1 inst 2]
"IGMP[vprn1 inst 2]: igmpProcessGroupRec
Process group rec CHG_TO_EXCL received on interface int-PE-3-H-3 [ifIndex 2] for group
239.255.0.1 in mode INCLUDE. Num srcs 0"

2 2015/09/10 08:25:30.13 UTC MINOR: DEBUG #2001 vprn1 IGMP[vprn1 inst 2]
"IGMP[vprn1 inst 2]: igmpIfGroupAdd
Adding 239.255.0.1 to IGMP interface int-PE-3-H-3 [ifIndex 2] database"

1 2015/09/10 08:25:08.60 UTC MINOR: DEBUG #2001 vprn1 IGMP[vprn1 inst 2]
"IGMP[vprn1 inst 2]: igmpSendQuery
Sending IGMP General query on interface int-PE-3-H-3 [ifIndex 2] to all-groups-address"

```

P2MP Label Mapping Message sent from PE-3 toward root node

```

*A:PE-3# show log log-id 2

8 2015/09/11 12:39:26.34 UTC MINOR: DEBUG #2001 Base LDP
"LDP: LDP
Send Label Mapping packet (msgId 185) to 192.0.2.4:0
Protocol version = 1
Label 262136 advertised for the following FECs
P2MP: root = 192.0.2.1, T: 1, L: 4, TunnelId: 8193

```

P2MP Label Mapping Message sent from PE-3 toward root node

```
"
7 2015/09/11 12:39:26.34 UTC MINOR: DEBUG #2001 Base LDP
"LDP: Binding
Sending Label mapping label 262136 for P2MP: root = 192.0.2.1, T: 1, L: 4, TunnelId: 8193
to peer 192.0.2.4:0."
```

BGP shared-join and source-join BGP route updates are sent via the RR toward the RP (source = 10.0.0.1) and the actual source (172.16.11.2), respectively:

```
*A:PE-3# show log log-id 2

9 2015/09/11 12:39:25.51 UTC MINOR: DEBUG #2001 Base Peer 1: 192.0.2.6
"Peer 1: 192.0.2.6: UPDATE
Peer 1: 192.0.2.6 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 93
  Flag: 0x90 Type: 14 Len: 57 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.3
    Type: Shared-Join Len:22 RD: 65545:1 SrcAS: 65545 Src: 10.0.0.1 Grp: 239.255.0.1
    Type: Source-Join Len:22 RD: 65545:1 SrcAS: 65545 Src: 172.16.11.2 Grp: 239.255.0.1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:192.0.2.1:2
"
```

The PIM status for the c-group 239.255.0.1 on PE-3 is now:

```
*A:PE-3# show router 1 pim group
=====
Legend:  A = Active    S = Standby
=====
PIM Groups ipv4
=====
Group Address          Type          Spt Bit   Inc Intf      No.Oifs
Source Address         RP            State      Inc Intf(S)
-----
239.255.0.1            (*,G)                mpls-if-73729* 1
*                      10.0.0.1
239.255.0.1            (S,G)          spt        mpls-if-73729* 1
172.16.11.2           10.0.0.1
-----
Groups : 2
=====
* indicates that the corresponding row element may have been truncated.
*A:PE-3#
```

Assume that a receiver on each of PE-4 and PE-5 needs to join group 239.255.0.1, as shown in [Figure 236](#).

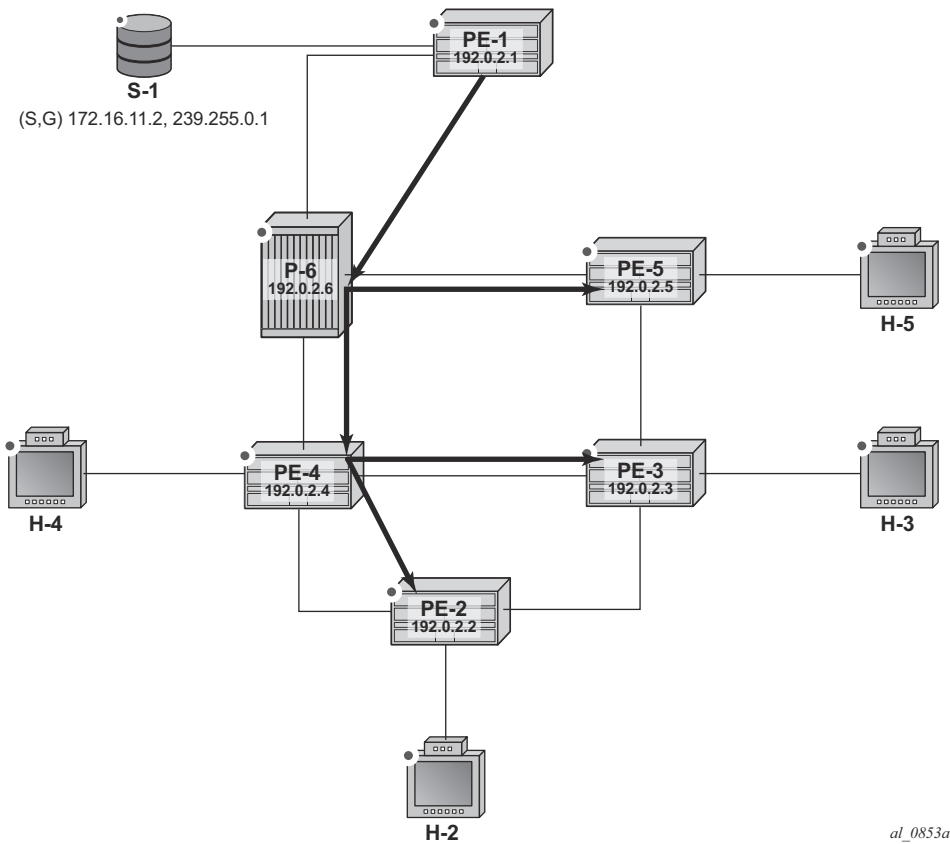


Figure 236: S-PMSI P2MP LSP Schematic

Figure 236 shows the S-PMSI P2MP LSP. The next set of outputs shows the P2MP label mapping of the LDP LSP between PE-1 and the receiver PEs.

The root of the S-PMSI is on PE-1:

```
*A:PE-1# show router ldp bindings active p2mp p2mp-id 8193 root 192.0.2.1
=====
LDP Bindings (IPv4 LSR ID 192.0.2.1:0)
(IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
      WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
=====
LDP Generic IPv4 P2MP Bindings (Active)
=====
P2MP-Id      Interface
RootAddr      Op      IngLbl  EgrLbl
EgrNH          EgrIf/LspId
-----
8193          73728
```

P2MP Label Mapping Message sent from PE-3 toward root node

```

192.0.2.1          Push          --          262137
192.168.16.2       1/1/1

```

```

-----
No. of Generic IPv4 P2MP Active Bindings: 1
=====

```

```

*A:PE-1#

```

The egress label on PE-1 becomes the ingress label on P-6. P-6 has two leaves: one toward PE-4 and one toward PE-5:

```

*A:P-6# show router ldp bindings active p2mp p2mp-id 8193 root 192.0.2.1

```

```

=====
LDP Bindings (IPv4 LSR ID 192.0.2.6:0)
              (IPv6 LSR ID ::[0])

```

```

-----
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route

```

```

=====
LDP Generic IPv4 P2MP Bindings (Active)

```

```

=====
P2MP-Id      Interface
RootAddr     Op          IngLbl    EgrLbl
EgrNH        EgrIf/LspId
-----
8193          Unknw
192.0.2.1     Swap          262137    262136
192.168.46.1  1/1/2
8193          Unknw
192.0.2.1     Swap          262137    262136
192.168.56.1  1/1/1

```

```

-----
No. of Generic IPv4 P2MP Active Bindings: 2
=====

```

On PE-5, the following output shows that the LSP terminates as a leaf, as the operation (Op) is shown as “pop”:

```

*A:PE-5# show router ldp bindings active p2mp p2mp-id 8193 root 192.0.2.1

```

```

=====
LDP Bindings (IPv4 LSR ID 192.0.2.5:0)
              (IPv6 LSR ID ::[0])

```

```

-----
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route

```

```

=====
LDP Generic IPv4 P2MP Bindings (Active)

```

```

=====
P2MP-Id      Interface
RootAddr     Op          IngLbl    EgrLbl
EgrNH        EgrIf/LspId
-----
8193          73728
192.0.2.1     Pop          262136    --

```

```

--
-----
No. of Generic IPv4 P2MP Active Bindings: 1
=====
*A:PE-5#

```

On PE-4, the P2MP LSP has 3 entries: a pop operation to receiver H-4, and two label swaps toward PE-3 and PE-2:

```

*A:PE-4# show router ldp bindings active p2mp p2mp-id 8193 root 192.0.2.1
=====
LDP Bindings (IPv4 LSR ID 192.0.2.4:0)
              (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use,  N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
=====
LDP Generic IPv4 P2MP Bindings (Active)
=====
P2MP-Id      Interface
RootAddr      Op          IngLbl    EgrLbl
EgrNH         EgrIf/LspId
-----
8193          73728
192.0.2.1     Pop          262136   --
--
8193          73728
192.0.2.1     Swap          262136   262136
192.168.24.1  1/1/2
8193          73728
192.0.2.1     Swap          262136   262136
192.168.34.1  1/1/3
-----
No. of Generic IPv4 P2MP Active Bindings: 3
=====
*A:PE-4#

```

PE-2 and PE-3 are termination PE's for P2MP leaf. On PE-2, the pop operation is shown:

```

*A:PE-2# show router ldp bindings active p2mp p2mp-id 8193 root 192.0.2.1
=====
LDP Bindings (IPv4 LSR ID 192.0.2.2:0)
              (IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use,  N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
=====
LDP Generic IPv4 P2MP Bindings (Active)
=====
P2MP-Id      Interface
RootAddr      Op          IngLbl    EgrLbl
EgrNH         EgrIf/LspId

```

P2MP Label Mapping Message sent from PE-3 toward root node

```
-----
8193                               73729
192.0.2.1                         Pop           262136      --
--                               --

-----
No. of Generic IPv4 P2MP Active Bindings: 1
=====
*A:PE-2#
```

On PE-3, the P2MP pop operation is shown:

```
*A:PE-3# show router ldp bindings active p2mp p2mp-id 8193 root 192.0.2.1
=====
LDP Bindings (IPv4 LSR ID 192.0.2.3:0)
(IPv6 LSR ID ::[0])
=====
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
        WP - Label Withdraw Pending, BU - Alternate For Fast Re-Route
=====
LDP Generic IPv4 P2MP Bindings (Active)
=====
P2MP-Id      Interface
RootAddr     Op           IngLbl      EgrLbl
EgrNH        EgrIf/LspId
-----
8193         73729
192.0.2.1    Pop           262136      --
--         --

-----
No. of Generic IPv4 P2MP Active Bindings: 1
=====
*A:PE-3#
```

Conclusion

MVPN wildcard Selective PMSI (S-PMSI), developed as per RFC 6625, provides an optimal solution for multicast routing in a VPRN. This protocol provides simple configuration, operation, and fast protection time in conjunction with MPLS and LDP fast-failover schemes. Wildcard S-PMSI can be used in a multicast network to avoid a large full mesh of an I-PMSI.

Source Redundancy in a Multicast VPN

In This Chapter

This section provides information about MVPN source redundancy.

Topics in this section include:

- [Applicability on page 1644](#)
- [Summary on page 1645](#)
- [Overview on page 1646](#)
- [Configuration on page 1649](#)
- [Conclusion on page 1681](#)

Applicability

This example is applicable to the 7750 SR-7/12, 7750 SR-c4/c12 and 7950 XRS. It can be configured on any supported IOM (IOM2 or IOM3-XP) or IMM type. It is also applicable to the 7450 ESS-7/12 with IOM3-XP or IMM. Chassis mode C or D is required for MPLS provider tunnels using multicast LDP or point to multi-point RSVP-TE Label Switched Paths.

The configuration was tested on release 12.0.R1, using multicast LDP as the provider tunnel signaling mechanism, for IPv4 multi-casting. The customer multicast signaling protocol within the VPN must be BGP.

Summary

Multicast source redundancy allows operators to provide multiple geo-redundant sources for the same multicast group in a multicast Virtual Private Network (MVPN). For instance, in an IPTV environment where a TV channel maps to a multicast group, the same TV channel can be provided from sources in a geographically diverse manner where a national broadcaster can have multiple sources from two or more regional distribution centers.

Knowledge of Multi-Protocol BGP (MP-BGP) and RFC 4364, *BGP/MPLS IP Virtual Private Networks (VPNs)*, is assumed throughout this example, as well as Protocol Independent Multicast (PIM), RFC 6513, *Multicast in MPLS/BGP IP VPNs*, and RFC 6514, *BGP Encodings and Procedures for Multicast in MPLS/BGP IP VPNs*.

Overview

Hosts connected to receiver PEs can receive TV channels from a specific source, with a regional backup source available in case of a failure.

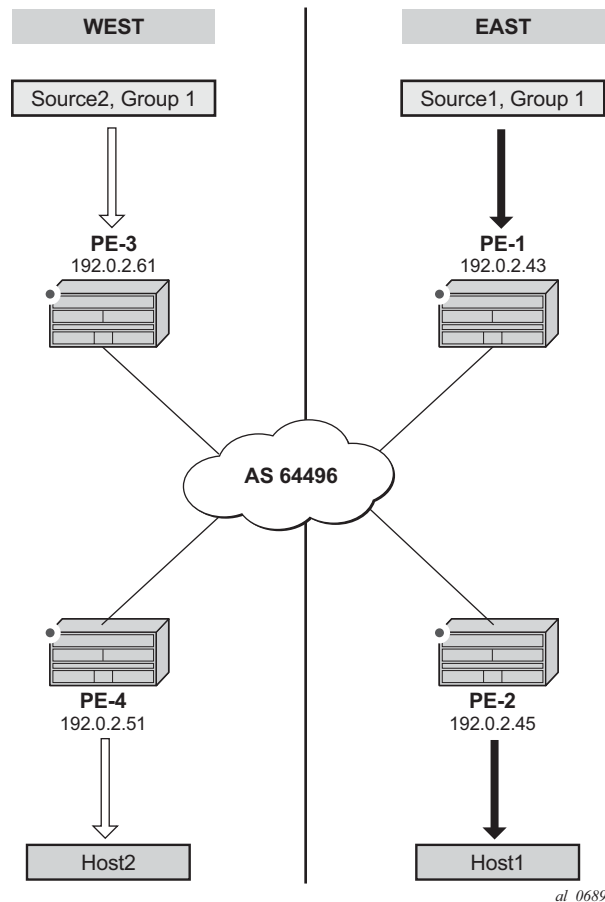


Figure 237: Source Redundancy Example.

Figure 237 shows the concept of source redundancy. PE-1 and PE-3 have directly connected multicast sources. For clarity, consider a single multicast group with two separate sources connected at different sites. The content of each group is identical at a given time (allowing for transmission delay), as is expected for an IPTV channel. PE-1 and PE-3 are referred to as sender PEs as they are closer to the source; PE2 and PE4 are referred to as receiver PEs as they are closer to hosts 1 and 2.

A multicast group, group 1 (G1) has two sources: Source 1 (S1) in the east region and Source (S2) in the west region which are connected to PE-1 and PE-3 respectively. Receivers connected to PE-2 in the east region will join group (S1,G1) and receivers connected to PE-4 in the west region will join group (S2,G1). The presence of each source is declared within the multicast VPN by the sender PE. When a multicast group becomes active, a BGP Source Active auto-discovery (SA) route is advertised to all PEs within the multicast VPN. This must occur even if no receiver indicates that it wishes to become a member of this group. In other words, the SA must be persistently present in the receiver PEs when the source is available.

Should either source fail or become unavailable, then the sender PE will notify the receiver PEs by sending an NLRI unreachable BGP SA Route that declares the absence of the source. All hosts that are members of this group will then switch to receive traffic from the remaining active source. Note that only customer multicast joins received as IGMP (*,G) queries or PIM (*,G) joins at the receiver PE are valid, as the source address is not specified.

Source redundancy is achieved by:

- Configuring a list of redundant sources within each receiver PE.
- Configuring the sender PEs to originate a BGP Source Active Auto Discovery for each detected active multicast source, regardless of whether a receiver is joined to the multicast group or not. As a result, a Source Active route is originated on a per (S,G) basis.

For multiple SAs to be persistently present in the receiver PEs, one of the following two conditions must be configured within the sender PEs:

- Either disable inter-site shared trees on the sender PEs, such that there is no c-tree with root at the RP. Any active source will announce its presence using a BGP SA to all receiver PEs so no shared joins are sent by receiver PEs to RP, or
- Leave inter-site shared trees as enabled, but configured so that the SA AD route for each multicast group is persistently present in the receiver PEs, even in the absence of requesting hosts for each group. Shared and Source Joins are sent by the receiver PEs.

Both of the above options are supported. The default behavior has inter-site shared trees enabled without persistency. In this example, inter-site shared trees at the sender PEs are enabled with Source Active routes set to be persistent.

- Ensuring that the preferred source is IP reachable within the VPRN from the receiver PE. This must be a remote source advertised from a remote PE within the VPRN.
- Receiver PEs will accept the Source Active route(s) into the appropriate Multicast VRF.
- Ensuring the preferred active source should have a higher BGP Local Preference. This is achieved using a route policy. Any other sources from the redundant list should exist as suppressed standby sources, but the (S,G) state should exist if the source is active – when a valid BGP MVPN Source Active route for that source has been received.

All of these conditions are achieved by configuration.

In order to allow each receiver PE to choose a preferred source, each SA route advertised by the sender PE will be tagged with a community value. Each receiver PE can then use the community value contained within each SA route update received to set the Local Preference BGP attribute to a value such that the receiver PE can choose the most preferred active source.

The objectives are:

- To configure multicast in a VPRN on PE-1 to PE-4 with inter-site-shared trees enabled on the receiver PEs and Source Active routes persistently present, for reasons previously described.
- To connect redundant sources to the sender PE-1 and PE-3, with each multicast source having the same group address. For ease of configuration a single redundant source is used.
- To advertise each source to the receiver PEs (PE-2 and PE-4), using appropriate route policies for adding community strings to the BGP Source Active Auto-Discovery routes.
- To configure appropriate route policies that allow each BGP SA route to have the correct Local Preference set, based on the community strings present.
- To allow receivers to connect to the appropriate source, using (*,G) joins.

The following configuration tasks should be completed as a pre-requisite:

- Full mesh ISIS or OSPF between each of the PE routers and the route reflector.
- Link-layer LDP between all PEs. (RSVP could also be used)
- Multicast LDP is used as the provider tunnel signaling protocol. This is enabled by default when link layer LDP is enabled.

For completeness, the command for all routers is as follows:

```
configure
router ldp
  interface-parameters
    interface int-PE-1-P-1
      multicast-traffic enable
    exit
  exit
exit
```

Note that RSVP and PIM SSM are also supported as provider tunnel signaling mechanisms and could be used.

Configuration

The test topology is shown in [Figure 238](#), containing the four PEs plus the route reflector at P-1.

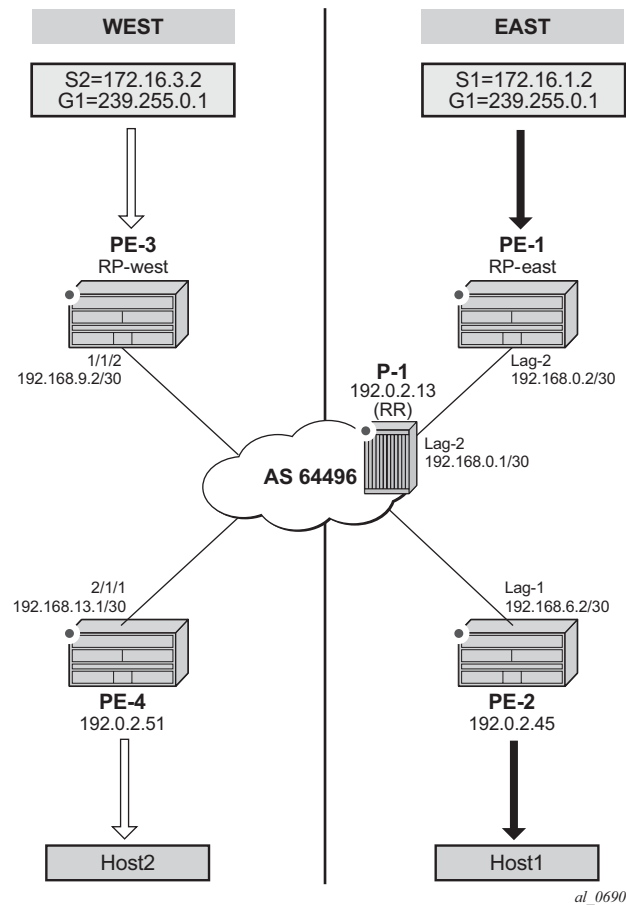


Figure 238: Schematic Topology

Global BGP Configuration

The first step is to configure an iBGP session between each of the PEs and the Route Reflector (RR) seen in [Figure 238](#). The address families negotiated between the iBGP peers are vpn-ipv4 (unicast routing) and mvpn-ipv4 (multicast routing).

The configuration for PE1 is:

```
configure router bgp
  group internal
    family vpn-ipv4 mvpn-ipv4
    type internal
    neighbor 192.0.2.13
  exit
exit
exit
```

The configuration for the other PE nodes is the same, apart from the IP addresses. The IP addresses can be derived from [Figure 238](#).

The configuration for the Route Reflector at P-1 is:

```
configure router bgp
  cluster 0.0.0.1
  group rr_clients
    type internal
    family vpn-ipv4 mvpn-ipv4
    neighbor 192.0.2.51
  exit
  neighbor 192.0.2.61
  exit
  neighbor 192.0.2.43
  exit
  neighbor 192.0.2.45
  exit
exit
exit all
```

On PE-1, verify that the BGP session with RR at P-1 is established with address families “vpn-ipv4” and “mvpn-ipv4” capabilities negotiated:

```
*B:PE-1>config>router>bgp>group# show router bgp summary
=====
BGP Router ID:192.0.2.43      AS:64496      Local AS:64496
=====
BGP Admin State      : Up      BGP Oper State      : Up
Total Peer Groups    : 1        Total Peers          : 1
Total BGP Paths       : 32      Total Path Memory    : 7864
Total IPv4 Remote Rts : 0        Total IPv4 Rem. Active Rts : 0
Total McIPv4 Remote Rts : 0      Total McIPv4 Rem. Active Rts : 0
Total McIPv6 Remote Rts : 0      Total McIPv6 Rem. Active Rts : 0
Total IPv6 Remote Rts  : 0        Total IPv6 Rem. Active Rts  : 0
```

Source Redundancy in a Multicast VPN

```

Total IPv4 Backup Rts      : 0          Total IPv6 Backup Rts      : 0
Total Supressed Rts       : 0          Total Hist. Rts           : 0
Total Decay Rts           : 0

Total VPN Peer Groups     : 0          Total VPN Peers           : 0
Total VPN Local Rts       : 10
Total VPN-IPv4 Rem. Rts   : 0          Total VPN-IPv4 Rem. Act. Rts: 0
Total VPN-IPv6 Rem. Rts   : 0          Total VPN-IPv6 Rem. Act. Rts: 0
Total VPN-IPv4 Bkup Rts   : 0          Total VPN-IPv6 Bkup Rts   : 0

Total VPN Supp. Rts       : 0          Total VPN Hist. Rts      : 0
Total VPN Decay Rts      : 0

Total L2-VPN Rem. Rts     : 0          Total L2VPN Rem. Act. Rts : 0
Total MVPN-IPv4 Rem Rts   : 0          Total MVPN-IPv4 Rem Act Rts : 0
Total MDT-SAFI Rem Rts    : 0          Total MDT-SAFI Rem Act Rts : 0
Total MSPW Rem Rts        : 0          Total MSPW Rem Act Rts     : 0
Total RouteTgt Rem Rts    : 0          Total RouteTgt Rem Act Rts : 0
Total McVpnIPv4 Rem Rts   : 0          Total McVpnIPv4 Rem Act Rts : 0
Total MVPN-IPv6 Rem Rts   : 0          Total MVPN-IPv6 Rem Act Rts : 0
Total EVPN Rem Rts        : 0          Total EVPN Rem Act Rts     : 0
Total FlowIpv4 Rem Rts    : 0          Total FlowIpv4 Rem Act Rts : 0
Total FlowIpv6 Rem Rts    : 0          Total FlowIpv6 Rem Act Rts : 0
=====
BGP Summary
=====
Neighbor
      AS PktRcvd InQ Up/Down State|Rcv/Act/Sent (Addr Family)
      PktSent OutQ
-----
192.0.2.13
      64496      616      0 00h00m05s 0/0/0 (VpnIPv4)
              12      0      0/0/0 (MvpnIPv4)
-----

```

The same command can be used on the other PEs to verify their BGP sessions to the RR.

Configuring VPRN on PEs

The following outputs show the VPRN configurations for each PE. The specific MVPN configuration is shown later.

PE-1

The VPRN configuration for PE-1 is as follows:

```
B:PE-1# configure service vprn 10
B:PE-1>config>service>vprn# info
-----
route-distinguisher 64496:10
auto-bind ldp
vrf-target target:64496:10
interface "int-PE-1-CE-1" create
    address 172.16.1.1/24
    sap 2/2/3:10.0 create
    exit
exit
interface "rp" create
    address 10.10.10.10/32
    loopback
exit
pim
    apply-to all
    rp
        static
            address 10.10.10.10
            group-prefix 239.255.0.0/8
        exit
    exit
    bsr-candidate
        shutdown
    exit
    rp-candidate
        shutdown
    exit
exit
no shutdown
exit
mvpn
    auto-discovery default
    c-mcast-signaling bgp
    provider-tunnel
        inclusive
        mldp
            no shutdown
        exit
    exit
exit
no shutdown
```


There is a single interface towards CE-1 from which the multicast group is generated.

If the customer signaling uses PIM ASM then the customer Rendezvous Point (RP) must be positioned on the sender PE as registration of the source with the RP causes the SA to be sent to the remote source PEs.

A loopback interface called **rp** acts as the RP for all group prefixes in the 239.255.0.0/16 range. This will be the RP for the East groups.

MVPN configuration enables BGP as both auto-discovery mechanism and the customer multicast signaling protocol across the VPRN. The provider tunnel between PEs within the MVPN is signaled using Multicast LDP.

PE-2

PE-2 contains an attached receiver so a single interface is configured to accommodate this. The RP configured is that of the East region and has a configuration as follows:

```
B:PE-2# configure service vprn 10
B:PE-2>config>service>vprn# info
-----
route-distinguisher 64496:10
auto-bind ldp
vrf-target target:64496:10
interface "int-PE-2-CE-2" create
    address 172.16.2.1/24
    sap 1/1/2:10.0 create
    exit
exit
igmp
    interface "int-PE-2-CE-2"
        no shutdown
    exit
    no shutdown
exit
pim
    rp
        static
            address 10.10.10.10
            group-prefix 239.255.0.0/8
        exit
    exit
    no shutdown
exit
no shutdown
```

PE-3

PE-3 acts as the RP for the West region and uses a different IP address for the Rendezvous Point interface.

```
*B:PE-3# configure service vprn 10
*B:PE-3>config>service>vprn# info
-----
route-distinguisher 64496:10
auto-bind ldp
vrf-target target:64496:10
interface "rp" create
    address 10.10.10.11/32
    loopback
exit
interface "int-PE3-CE-3" create
    address 172.16.3.1/24
    sap 2/1/3:10 create
    exit
exit
pim
    apply-to all
    rp
        static
            address 10.10.10.11
            group-prefix 239.255.0.0/8
        exit
    exit
exit
no shutdown
exit
no shutdown
-----
```

PE-4

PE-4 also contains a receiver, and uses the West Sender PE (PE-3) as the Rendezvous Point.

```
*B:PE-4>config>service>vpn# info
-----
route-distinguisher 64496:10
auto-bind ldp
vrf-target target:64496:10
interface "int-PE-4-CE-4" create
    address 172.16.4.1/24
    sap 2/2/2:10 create
    exit
exit
igmp
    interface "int-PE-4-CE-4"
        no shutdown
    exit
    no shutdown
exit
pim
    rp
        static
            address 10.10.10.11
            group-prefix 239.255.0.0/8
        exit
    exit
    exit
    no shutdown
exit
no shutdown
-----
```

MVPN Configuration for Source PEs

At the PEs closest to the sources (PE-1 and PE-3) Source Active auto-discovery BGP routes are generated when the source is active.

This applies for PIM-ASM (*,G) joins only, or IGMP (*,G) membership queries received by the provider domain. These are received by all PEs.

Inter-site trees must be disabled for this to occur. Alternatively, inter-site trees can be enabled such that when a source is discovered, a Source Active is advertised to each other PE in the MVPN. This occurs regardless of whether any receivers wish to become members of the multicast groups.

As previously stated, the presence of the SA in the receiver PEs means that no shared joins routes are generated towards the C-RPs.

The MVPN configuration for each PE should be as follows:

```
B:PE-1>config>service>vprn# info
-----
      mvpn
        auto-discovery default
        c-mcast-signaling bgp
        provider-tunnel
          inclusive
          mldp
            no shutdown
          exit
        exit
      exit
    exit
```

The VPRN MVPN configuration for PEs 2, 3 and 4 is identical.

Sender PE Route Policies

The choice of active and standby sources by the receiver PEs is determined by the “best route” policy. PE-1 and PE-3 advertises BGP Source Active Auto Discovery routes when a source is active. This is received by all PEs within the MVPN. As two different sources advertise the same group, it is necessary to differentiate between them.

Assuming that PE-2 receiver prefers the source from PE-1, and PE-4 prefers the source active on PE-3, then the export policy for MVPN routes on PE-1 requires the following steps:

1. Set a community value at PE-1 for the (S,G) multicast group – call this “blue” with value 64496:452.
2. Set the route target community for the VPRN – 64496:10.
3. Create a policy statement that becomes the export policy for MVPN routes within PE-1.
4. Create a policy statement entry (entry 10) that adds the community value “blue” along with the route target for Source Active AD BGP routes. Source Active AD routes are MVPN type 5 routes.
5. Create a policy statement entry default-action that adds the route target for all other MVPN AD BGP routes (e.g. Intra-AD (type 1)) that are exported to the MVPN PEs.

```
B:PE-1# configure router policy-options
begin
  community "blue" members "64496:452"
  community "mvpn_10_rt" members "target:64496:10"

  policy-statement "mvpn_10"
    entry 10
      description "match mvpn routes - type 5 Source AD - add RT and 'blue'
                  community"
      from
        mvpn-type 5
        family mvpn-ipv4
      exit
      action accept
        community add "blue" "mvpn_10_rt"
      exit
    exit
    default-action accept
      community add "mvpn_10_rt"
    exit
  exit
```

6. Apply as an export policy within the MVPN context.

Sender PE Route Policies

The import policy requires that all imported MVPN BGP routes have the correct route target extended community value, specifically 64496:10.

1. Create a policy statement that becomes the import policy for PE-1.
2. Create a policy statement entry (entry 10) that matches the community of the route target extended community for all MVPN BGP routes. These include the Intra-AD and Source-Join routes.

```
B:PE-1# configure router policy-options
begin
  community "blue" members "64496:452"
  community "mvpn_10_rt" members "target:64496:10"
  policy-statement "mvpn_10_import"
    entry 10
      from
        community "mvpn_10_rt"
      exit
      action accept
      exit
    exit
  exit
```

Enable the inter-site-shared type 5 advertisement persistency so that source ADs are advertised when multicast sources are active. Note that alternatively, inter-site shared trees can be disabled using the **no intersite shared** command. In this example only inter-site shared MVPN type 5 persistency is shown.

The MVPN configuration for PE-1 now looks as follows:

```
*B:PE-1>config>service>vprn>mvpn# info
-----
  auto-discovery default
  c-mcast-signaling bgp
  intersite-shared persistent-type5-adv
  provider-tunnel
    inclusive
      mldp
        no shutdown
      exit
    exit
  exit
  vrf-import "mvpn_10_import"
  vrf-export "mvpn_10_export"
```

For PE-3 (the other sender PE), similar import and export policies are required. In this case, the community will be called “red” and is added to the Source Active AD route generated when the source is active.

The requirements for the export policy for PE-3 are as follows:

```
B:PE-3# configure router policy-options
begin
  community "red" members "64496:332"
  community "mvpn_10_rt" members "target:64496:10"
  policy-statement "mvpn_10_export"
    entry 10
      from
        mvpn-type 5
        family mvpn-ipv4
      exit
      action accept
        community add "red" "mvpn_10_rt"
      exit
    exit
  default-action accept
    community add "mvpn_10_rt"
  exit
exit
```

The import policy is:

```
policy-statement "mvpn_10_import"
  entry 10
    from
      community "mvpn_10_rt"
    exit
    action accept
    exit
  exit
exit
```

Apply the import and export policies to the MVPN context of the sender PE (PE-3).

```
*B:PE-3>config>service>vprn>mvpn# info
-----
  auto-discovery default
  c-mcast-signaling bgp
  intersite-shared persistent-type5-adv
  provider-tunnel
    inclusive
      mldp
        no shutdown
      exit
    exit
  exit
  vrf-import "mvpn_10_import"
  vrf-export "mvpn_10_export"
```

Receiver PE Configuration

PE-2 and PE-4 are the receiver PEs. These will receive the Source Active AD routes and initiate Joins towards the preferred source.

When a Source-Active AD route is received, the community value is examined and the Local Preference value of the route is set using a Route Policy. The preferred source is determined by the SA AD route with the highest Local Preference value.

In the case of PE-2, the preferred source is that advertised by PE-1, the “blue” source as previously referenced. PE-2 sets the Local Preference to 200. The SA AD tagged with the “red” community has the Local Preference set to 50.

For PE-4, the reverse applies: SA AD routes tagged with the “red” community have the Local Preference set to 200, and “blue” SA AD routes have the Local Preference set to 50.

Once again, assuming that the PE-2 receiver prefers the source from PE-1 and PE-4 prefers the source active on PE-3, the import policy for MVPN routes on PE-2 requires the following steps:

1. Set a community value at PE-2 for the (S,G), call this “blue” with value 64496:452.
2. Set the route target community for the VPRN to 64496:10.
3. Create a prefix list that matches the multicast group address, in this case 239.255.0.0/24.
4. Create a policy statement that becomes the import policy for MVPN routes within PE-1.
5. Create a policy statement entry (entry 10) that matches the following attributes:
 - Source Active AD BGP routes type. Source Active AD routes are classed as MVPN type 5 routes, and
 - Community value “blue” AND Route Target extended community, and
 - Group address prefix 239.255.0.0/24

If the BGP route matches all three conditions then set the Local Preference to 200.

6. Create a policy statement default-action that accepts all other MVPN BGP routes, including SA routes tagged with the “red” community value.

The import policy statement looks like:

```
*B:PE-2>config>router>policy-options# info
-----
community "red" members "64496:332"
community "blue" members "64496:452"
community "mvpn_10_rt" members "target:64496:10"
policy-statement "mvpn_10_import"
  entry 10
    description "allow mvpn source-ad for 'red' group - set local-pref to
                200"
    from
```



```

        community expression "[blue] AND [mvpn_10_rt]"
        mvpn-type 5
    exit
    action accept
        local-preference 200
    exit
exit
entry 20
    from
        community expression "[red] AND [mvpn_10_rt]"
        mvpn-type 5
    exit
    action accept
        local-preference 50
    exit
    exit
default-action accept
exit
exit

```

The export policy for PE-2 MVPN routes requires each MVPN route to be tagged with the route target extended community for VPRN 10. The following policy statement is created:

```

*B:PE-2>config>router>policy-options# info
-----
begin
prefix-list "group_239.255.x.x"
    prefix 239.255.0.0/16 longer
exit
community "red" members "64496:332"
community "blue" members "64496:452"
community "mvpn_10_rt" members "target:64496:10"
policy-statement "mvpn_10_export"
    entry 10
        from
            family mvpn-ipv4
        exit
        action accept
            community add "mvpn_10_rt"
        exit
    exit
exit
exit
commit

```

7. Create a list of redundant sources. This is a list of prefixes that match the source addresses of redundant multicast groups. This is an important parameter as the receiver PEs only creates active and standby (S,G) states for groups with source address prefixes that are contained in this list.
8. Before any hosts attempt to join the multicast groups, the decision must be made to enable or disable inter-site shared trees at the receiver PEs. In this example, only the Inter-site shared trees disabled option will be considered. In order to make this change in configuration, it is necessary to shut the PIM protocol down before and re-enable when completed.

Receiver PE Configuration

The MVPN configuration for PE-2 is shown in the following output. Note the redundant source prefix list is included, and inter-site shared trees are disabled:

```
B:PE-2# configure service vprn 10
B:PE-2>config>service>vprn# info
-----
route-distinguisher 64496:10
auto-bind ldp
vrf-target target:64496:10
mvpn
  auto-discovery default
  c-mcast-signaling bgp
  no intersite-shared
  red-source-list
    src-prefix 172.16.1.0/24
    src-prefix 172.16.3.0/24
  exit
  provider-tunnel
    inclusive
    mldp
      no shutdown
    exit
  exit
  exit
  vrf-import "mvpn_10_import"
  vrf-export "mvpn_10_export"
exit
no shutdown
```

PE-4 requires a similar set of import and export policies. In this case, the “red” sources have the highest Local Preference value, based on the community string added by the export policy of PE-3.

```
B:PE-4>config>router>policy-options# info
-----
begin
community "red" members "64496:332"
community "mvpn_10_rt" members "target:64496:10"
policy-statement "mvpn_10_import"
  entry 10
    description "Match Source AD from 'red' group - set local preference to 200"
    from
      community expression "[red] AND [mvpn_10_rt]"
      mvpn-type 5
    exit
    action accept
      local-preference 200
    exit
  exit
  entry 20
    description "Match Source AD from 'blue' group - set local preference to 50"
    from
      community expression "[blue] AND [mvpn_10_rt]"
      mvpn-type 5
    exit
    action accept
```

```

        local-preference 50
    exit
exit
default-action accept
exit
exit
commit

```

The export policy for MVPN routes adds the route target extended community, as follows:

```

B:PE-4>config>router>policy-options# info
-----
begin
community "mvpn_10_rt" members "target:64496:10"
policy-statement "mvpn_10_export"
  description "export policy for mvpn routes"
  entry 10
    description "add RT for mvpn routes"
    from
      family mvpn-ipv4
    exit
    action accept
      community add "mvpn_10"
    exit
  exit
exit
commit

```

The MVPN configuration for VPRN 10 on PE-4 is now:

```

B:PE-4# configure service vprn 10
B:PE-4>config>service>vprn# info
-----
mvpn
  auto-discovery default
  c-mcast-signaling bgp
  no intersite-shared
  red-source-list
    src-prefix 172.16.1.0/24
    src-prefix 172.16.3.0/24
  exit
  provider-tunnel
    inclusive
    mldp
    no shutdown
  exit
  exit
  vrf-import "mvpn_10_import"
  vrf-export "mvpn_10_export"
exit
no shutdown

```

Receiver PE Configuration

Each PE within the MVPN originates an Intra-AD BGP route. This notifies the other PEs within the VPRN. This is used to create a set of Inclusive Provider Multicast Service Interfaces (I-PMSI) between each PE. In this case, I-PMSIs are signaled using mLDP.

Using PE-1 as an example, the set of Intra-AD routes can be seen using the following command:

```
*B:PE-1# show router bgp routes mvpn-ipv4
=====
BGP Router ID:192.0.2.43      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP MVPN-IPv4 Routes
=====
Flag  RouteType      OriginatorIP      LocalPref  MED
      RD          SourceAS          Label
      Nexthop      SourceIP
      As-Path      GroupIP
-----
i     Intra-Ad      192.0.2.43        100        0
      64496:10      -                 -
      192.0.2.43    -                 -
      No As-Path    -                 -
u*>i  Intra-Ad      192.0.2.45        100        0
      64496:10      -                 -
      192.0.2.45    -                 -
      No As-Path    -                 -
u*>i  Intra-Ad      192.0.2.51        100        0
      64496:10      -                 -
      192.0.2.51    -                 -
      No As-Path    -                 -
u*>i  Intra-Ad      192.0.2.61        100        0
      64496:10      -                 -
      192.0.2.61    -                 -
      No As-Path    -                 -
-----
Routes : 4
=====
```

At this moment, there are no connected sources detected and no receivers wishing to join any multicast sources.

Each I-PMSI is seen as a PIM tunnel interface. As there are four routers in the MVPN, there are four I-PMSIs.

```
*B:PE-1# show router 10 pim tunnel-interface
=====
PIM Interfaces ipv4
=====
Interface      Adm  Opr  DR Prty      Hello Intvl  Mcast Send
DR
-----
mpls-if-73734  Up   Up   N/A          N/A          N/A
```

```

    192.0.2.43
mpls-if-73735      Up    Up    N/A              N/A              N/A
    192.0.2.51
mpls-if-73736      Up    Up    N/A              N/A              N/A
    192.0.2.45
mpls-if-73737      Up    Up    N/A              N/A              N/A
    192.0.2.61
-----
Interfaces : 4
=====

```

In order to be able to reach the source, a route for each source is included in the VRF for VPRN 10.

For PE-2, this looks as follows:

```

*B:PE-2# show router 10 route-table
=====
Route Table (Service: 10)
=====
Dest Prefix[Flags]                                Type    Proto    Age          Pref
  Next Hop[Interface Name]                        Metric
-----
10.10.10.10/32                                     Remote  BGP VPN    03h30m32s   170
    192.0.2.61 (tunneled)                          0
10.10.10.11/32                                     Remote  BGP VPN    02h25m19s   170
    192.0.2.43 (tunneled)                          0
172.16.1.0/24                                       Remote  BGP VPN    02h25m19s   170
    192.0.2.43 (tunneled)                          0
172.16.2.0/24                                       Local   Local      02d02h51m   0
    int-PE-2-CE-2                                  0
172.16.3.0/24                                       Remote  BGP VPN    03h30m32s   170
    192.0.2.61 (tunneled)                          0
172.16.4.0/24                                       Remote  BGP VPN    03h18m27s   170
    192.0.2.51 (tunneled)                          0
-----
No. of Routes: 6
Flags: n = Number of times nexthop is repeated
      B = BGP backup route available
      L = LFA nexthop available
=====

```

The sources at 172.16.1.0/24 and 172.16.3.0/24 are learned as BGP VPN routes.

Examining the BGP routes for these prefixes. For 172.16.1.0/24 on PE-2:

```

*B:PE-2# show router bgp routes vpn-ipv4 172.16.1.0/24 hunt
=====
BGP Router ID:192.0.2.45      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP VPN-IPv4 Routes

```

Receiver PE Configuration

```
=====
-----
RIB In Entries
-----
Network      : 172.16.1.0/24
Nextthop     : 192.0.2.43
Route Dist.  : 64496:10          VPN Label      : 262072
Path Id      : None
From         : 192.0.2.13
Res. Nextthop : n/a
Local Pref.  : 100
Aggregator AS : None           Interface Name : int-PE-2-P-3
Atomic Aggr. : Not Atomic      Aggregator   : None
AIGP Metric  : None           MED          : None
Connector    : None
Community    : target:64496:10 l2-vpn/vrf-imp:192.0.2.43:2
               source-as:64496:0
Cluster      : 192.0.2.13
Originator Id : 192.0.2.43      Peer Router Id : 192.0.2.13
Fwd Class    : None           Priority      : None
Flags        : Used Valid Best IGP
Route Source  : Internal
AS-Path       : No As-Path
Neighbor-AS   : N/A
VPRN Imported : 10
-----
RIB Out Entries
-----
-----
Routes : 1
=====
*B:PE-2#
```

Note that this prefix is advertised with three communities:

- A Route Target extended community
- An l2-vpn/vrf-import extended community.
- A source-AS extended community (not used in Intra-AS context).

The l2-vpn/vrf-import extended community is significant as it is a unique value. It represents a specific MVPN on a specific PE and is comprised of a 32 bit value that identifies the PE plus an index identifying the VRF. The 32 bit value is the system address. The index (2) can be derived from the command:

```
*B:PE-1# admin display-config index | match vprn10
      virtual-router "vprn10" 2 0
```

Hence, the l2-vpn/vrf-import community for VPRN 10 on PE-1 is 192.0.2.43:2

This community attribute is included within the source-join BGP route that is sent in a BGP Update by a receiver PE as it tries to join a multicast group with a source address that matches the 172.16.1.0/24 prefix. This ensures that the source-join route is only accepted as a valid route and imported by the PE that originated the source address prefix. This is explained below.

Enable Redundant Sources

The redundant sources are now enabled so that multicast traffic flows into both PE-1 and PE-3, using groups (S1,G1) and (S2,G1), respectively.

On each of these PEs a source active AD route is generated. By examining each receiver PE, these can be clearly seen.

For PE-2, the source active AD routes can be seen using the following command.

```
*B:PE-2# show router bgp routes mvpn-ipv4 type source-ad
=====
BGP Router ID:192.0.2.45      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP MVPN-IPv4 Routes
=====
Flag  RouteType      OriginatorIP      LocalPref  MED
      RD          SourceAS          Label
      Nexthop      SourceIP
      As-Path      GroupIP
-----
u*>i  Source-Ad      -                  200        0
      64496:10      -                  -
      192.0.2.43    172.16.1.2
      No As-Path    239.255.0.1
u*>i  Source-Ad      -                  100        0
      64496:10      -                  -
      192.0.2.61    172.16.3.2
      No As-Path    239.255.0.1
-----
Routes : 2
=====
```

Note that there are two routes present, one from each source for the same group from PE-1 and PE-3.

The PIM groups can now be seen on PE-1 as follows:

```
B:PE-1# show router 10 pim group
=====
Legend:  A = Active   S = Standby
=====
PIM Groups ipv4
=====
Group Address      Type      Spt Bit  Inc Intf      No.Oifs
Source Address      RP        State    Inc Intf(S)
-----
239.255.0.1        (S,G)                    int-PE-1-CE-1  0
172.16.1.2         10.10.10.11
239.255.0.1        (S,G)                    mpls-if-73833  0
```



```

172.16.3.2          10.10.10.11
-----
Groups : 2
=====

```

There are two groups at PE-1. In addition to its locally connected source, PE-1 has also received a source active from PE-3 which has an incoming interface of the I-PMSI towards PE-3. Note that the outgoing interface list is empty as there is no host wishing to become a group member.

Similarly, on the other sender, PE-3.

```

*B:PE-3# show router 10 pim group
=====
Legend:  A = Active   S = Standby
=====
PIM Groups ipv4
=====
Group Address          Type          Spt Bit   Inc Intf      No.Oifs
  Source Address        RP           State      Inc Intf(S)
-----
239.255.0.1            (S,G)                mpls-if-73820  0
  172.16.1.2            10.10.10.10
239.255.0.1            (S,G)                int-PE3-CE-3   0
  172.16.3.2            10.10.10.10
-----
Groups : 2
=====

```

By examining the receiver PE-2, it can be seen that the Source AD route for (S,G) (172.16.1.2, 239.255.0.1) from PE-1 has a higher local preference so it is chosen as the preferred (active) source. Examining these routes in more detail shows that each route is tagged with two communities: the route target extended community and the “red” or “blue” community, as seen in the following output.

```

*B:PE-2# show router bgp routes mvpn-ipv4 type source-ad hunt
=====
BGP Router ID:192.0.2.45      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP MVPN-IPv4 Routes
=====
RIB In Entries
-----
Route Type      : Source-Ad
Route Dist.     : 64496:10
Source IP       : 172.16.1.2
Group IP        : 239.255.0.1
Nexthop        : 192.0.2.43
From            : 192.0.2.13
Res. Nexthop    : 0.0.0.0

```

Enable Redundant Sources

```
Local Pref.      : 200                      Interface Name : NotAvailable
Aggregator AS   : None                      Aggregator    : None
Atomic Aggr.    : Not Atomic                MED           : 0
AIGP Metric     : None
Connector       : None
Community       : 64496:452 target:64496:10
Cluster         : 192.0.2.13
Originator Id   : 192.0.2.43                Peer Router Id : 192.0.2.13
Flags           : Used Valid Best IGP
Route Source    : Internal
AS-Path         : No As-Path
Neighbor-AS     : N/A
VPRN Imported   : 10
```

```
Route Type      : Source-Ad
Route Dist.     : 64496:10
Source IP       : 172.16.3.2
Group IP        : 239.255.0.1
Nexthop         : 192.0.2.61
From           : 192.0.2.13
Res. Nexthop    : 0.0.0.0
Local Pref.     : 100                      Interface Name : NotAvailable
Aggregator AS   : None                      Aggregator    : None
Atomic Aggr.    : Not Atomic                MED           : 0
AIGP Metric     : None
Connector       : None
Community       : 64496:332 target:64496:10
Cluster         : 192.0.2.13
Originator Id   : 192.0.2.61                Peer Router Id : 192.0.2.13
Flags           : Used Valid Best IGP
Route Source    : Internal
AS-Path         : No As-Path
Neighbor-AS     : N/A
VPRN Imported   : 10
```

RIB Out Entries

Routes : 2
=====

The Local Preference is set based on these community values.

A debug of the received BGP Source AD routes is shown below for PE-2:

```
B:PE-2# 1 2014/03/05 13:51:51.31 GMT MINOR: DEBUG #2001 Base Peer 1: 192.0.2.13
"Peer 1: 192.0.2.13: UPDATE
Peer 1: 192.0.2.13 - Received BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 86
  Flag: 0x90 Type: 14 Len: 29 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.43
    Type: Source-AD Len: 18 RD: 64496:10 Src: 172.16.1.2 Grp: 239.255.0.1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
```

```

Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
Flag: 0xc0 Type: 8 Len: 4 Community:
    64496:452
Flag: 0x80 Type: 9 Len: 4 Originator ID: 192.0.2.43
Flag: 0x80 Type: 10 Len: 4 Cluster ID:
    192.0.2.13
Flag: 0xc0 Type: 16 Len: 8 Extended Community:
    target:64496:10
"

2 2014/03/05 13:51:51.31 GMT MINOR: DEBUG #2001 Base Peer 1: 192.0.2.13
"Peer 1: 192.0.2.13: UPDATE
Peer 1: 192.0.2.13 - Received BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 86
    Flag: 0x90 Type: 14 Len: 29 Multiprotocol Reachable NLRI:
        Address Family MVPN_IPV4
        NextHop len 4 NextHop 192.0.2.61
        Type: Source-AD Len: 18 RD: 64496:10 Src: 172.16.3.2 Grp: 239.255.0.1
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x80 Type: 4 Len: 4 MED: 0
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 8 Len: 4 Community:
        64496:332
    Flag: 0x80 Type: 9 Len: 4 Originator ID: 192.0.2.61
    Flag: 0x80 Type: 10 Len: 4 Cluster ID:
        192.0.2.13
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:64496:10
"

```

Similarly, the Source Active routes on receiver PE-4 show that the highest local preference value of 200 is set for the SA route received from PE-3 with an originator ID of 192.0.2.61, as shown below:

```

B:PE-4# show router bgp routes mvpn-ipv4 type source-ad hunt
=====
BGP Router ID:192.0.2.51      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP MVPN-IPv4 Routes
=====
-----
RIB In Entries
-----
Route Type      : Source-Ad
Route Dist.     : 64496:10
Source IP       : 172.16.1.2
Group IP        : 239.255.0.1
Nexthop         : 192.0.2.43
From            : 192.0.2.13
Res. Nexthop    : 0.0.0.0
Local Pref.     : 100
Interface Name  : NotAvailable

```

Enable Redundant Sources

```
Aggregator AS : None                      Aggregator : None
Atomic Aggr.  : Not Atomic                MED        : 0
AIGP Metric   : None
Connector     : None
Community     : 64496:452 target:64496:10
Cluster       : 192.0.2.13
Originator Id : 192.0.2.43                Peer Router Id : 192.0.2.13
Flags         : Used Valid Best IGP
Route Source  : Internal
AS-Path       : No As-Path
Neighbor-AS   : N/A
VPRN Imported : 10

Route Type    : Source-Ad
Route Dist.   : 64496:10
Source IP     : 172.16.3.2
Group IP      : 239.255.0.1
Nexthop       : 192.0.2.61
From          : 192.0.2.13
Res. Nexthop  : 0.0.0.0
Local Pref.   : 200                      Interface Name : NotAvailable
Aggregator AS : None                      Aggregator : None
Atomic Aggr.  : Not Atomic                MED        : 0
AIGP Metric   : None
Connector     : None
Community     : 64496:332 target:64496:10
Cluster       : 192.0.2.13
Originator Id : 192.0.2.61                Peer Router Id : 192.0.2.13
Flags         : Used Valid Best IGP
Route Source  : Internal
AS-Path       : No As-Path
Neighbor-AS   : N/A
VPRN Imported : 10

-----
RIB Out Entries
-----
-----
Routes : 2
=====
*B:PE-4#
```

Host Group Membership

If the hosts then send a (*,G) request to join the group, a source-join route is originated at each receiver PE towards the preferred source from the redundant list.

The following output shows a join originated by PE-2:

```
*B:PE-2# show debug
debug
  router "Base"
    bgp
      update neighbor 192.0.2.13
    exit
  exit
exit

*B:PE-2#
1 103 2014/05/27 11:15:19.29 BST MINOR: DEBUG #2001 Base Peer 1: 192.0.2.13
"Peer 1: 192.0.2.13: UPDATE
Peer 1: 192.0.2.13 - Send BGP UPDATE:
  Withdrawn Length = 0
  Total Path Attr Length = 101
  Flag: 0x90 Type: 14 Len: 57 Multiprotocol Reachable NLRI:
    Address Family MVPN_IPV4
    NextHop len 4 NextHop 192.0.2.45
    Type: Shared-Join Len:22 RD: 64496:101 SrcAS: 64496 Src: 10.10.10.11 Grp
: 239.255.0.1
    Type: Source-Join Len:22 RD: 64496:101 SrcAS: 64496 Src: 172.16.1.2 Grp:
239.255.0.1
  Flag: 0x40 Type: 1 Len: 1 Origin: 0
  Flag: 0x40 Type: 2 Len: 0 AS Path:
  Flag: 0x80 Type: 4 Len: 4 MED: 0
  Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
  Flag: 0xc0 Type: 16 Len: 16 Extended Community:
    target:64496:10
    target:192.0.2.43:2
"
```

Note that there is both a source-join and a shared join for the same group address (239.255.0.1).

The shared-join is sent to the Rendezvous Point 10.10.10.11 and the source-join is trying to become a member of group 239.255.0.1 with a source address of 172.16.1.2. As this is sent as a BGP routing update, this must be accepted by the MVPN VRF at the PE that originated the unicast route that represents the c-multicast source. As previously mentioned, there are two extended community values. The second of these is the l2-vpn/vrf-import route target for 192.0.2.43 (PE-1), so only PE-1 will accept this route.

Host Group Membership

Examining the PIM state table for PE-2 shows the presence of a group with multiple sources.

```
*B:PE-2# show router 10 pim group
=====
Legend:  A = Active   S = Standby
=====
PIM Groups ipv4
=====
Group Address          Type          Spt Bit   Inc Intf      No.Oifs
Source Address         RP           State      Inc Intf(S)
-----
239.255.0.1            (*,G)                mpls-if-73881  1
*                      10.10.10.11
239.255.0.1            (S,G)            spt        mpls-if-73881  1
172.16.1.2             10.10.10.11      A
239.255.0.1            (S,G)                mpls-if-73882  1
172.16.3.2             10.10.10.11      S
-----
Groups :3
=====
```

Note that each (S,G) has a state of either Active (A) or Standby (S), and the active group is chosen based on the Source Active AD with the highest Local Preference.

As a direct comparison, PE-4 also has the same two (S,G) states, but has a reversed active and standby source.

```
B:PE-4# show router 10 pim group
=====
Legend:  A = Active   S = Standby
=====
PIM Groups ipv4
=====
Group Address          Type          Spt Bit   Inc Intf      No.Oifs
Source Address         RP           State      Inc Intf(S)
-----
239.255.0.1            (*,G)                mpls-if-73889  1
*                      10.10.10.10
239.255.0.1            (S,G)                mpls-if-73888  1
172.16.1.2             10.10.10.10      S
239.255.0.1            (S,G)            spt        mpls-if-73889  1
172.16.3.2             10.10.10.10      A
-----
Groups : 3
=====
```

The Source Active ADs received on PE-4 have their Local Preference values based on the community string value.

```

B:PE-4# show router bgp routes mvpn-ipv4 type source-ad hunt
=====
BGP Router ID:192.0.2.51      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP MVPN-IPv4 Routes
=====
-----
RIB In Entries
-----
Route Type      : Source-Ad
Route Dist.     : 64496:10
Source IP       : 172.16.1.2
Group IP        : 239.255.0.1
Nexthop         : 192.0.2.43
From            : 192.0.2.13
Res. Nexthop    : 0.0.0.0
Local Pref.     : 100
Interface Name  : NotAvailable
Aggregator AS   : None
Aggregator      : None
Atomic Aggr.    : Not Atomic
MED             : 0
AIGP Metric     : None
Connector       : None
Community       : 64496:452 target:64496:10
Cluster         : 192.0.2.13
Originator Id   : 192.0.2.43      Peer Router Id : 192.0.2.13
Flags           : Used Valid Best IGP
Route Source    : Internal
AS-Path         : No As-Path
Neighbor-AS     : N/A
VPRN Imported   : 10

Route Type      : Source-Ad
Route Dist.     : 64496:10
Source IP       : 172.16.3.2
Group IP        : 239.255.0.1
Nexthop         : 192.0.2.61
From            : 192.0.2.13
Res. Nexthop    : 0.0.0.0
Local Pref.     : 200
Interface Name  : NotAvailable
Aggregator AS   : None
Aggregator      : None
Atomic Aggr.    : Not Atomic
MED             : 0
AIGP Metric     : None
Connector       : None
Community       : 64496:332 target:64496:10
Cluster         : 192.0.2.13
Originator Id   : 192.0.2.61      Peer Router Id : 192.0.2.13
Flags           : Used Valid Best IGP
Route Source    : Internal
AS-Path         : No As-Path
Neighbor-AS     : N/A
VPRN Imported   : 10

```

Host Group Membership

```
-----  
RIB Out Entries  
-----  
-----  
Routes : 2  
=====
```


Sender PE MVPN status

The MVPN status of the PE-1 sender PE is as follows:

```
*B:PE-1# show router 10 mvpn
=====
MVPN 10 configuration data
=====
signaling          : Bgp                auto-discovery      : Default
UMH Selection      : Highest-Ip
intersite-shared   : Disabled            Persist SA         : Disabled
vrf-import         : mvpn_10_import
vrf-export         : mvpn_10_export
vrf-target         : N/A
C-Mcast Import RT  : target:192.0.2.43:2

ipmsi              : ldp
i-pmsi P2MP AdmSt  : Up
i-pmsi Tunnel Name : mpls-if-73829
Mdt-type           : sender-receiver

s-pmsi             : none
data-delay-interval: 3 seconds
enable-asm-mdt     : N/A
=====
```

Note that the C-Mcast Import RT is set to <system-address>:<VPRN index>.

The VPRN index is derived from the command:

```
*B:PE-1# admin display-config index | match vprn10
virtual-router "vprn10" 2 0
```

Any Source Join received must include this attribute along with the route target extended community. As previously stated, this is advertised within the VPN-IPv4 routes as a BGP attribute.

A source join is shown below, received from PE-2, to join (S,G) (172.16.1.2, 239.255.0.1).

```
*B:PE-1# show router bgp routes mvpn-ipv4 type source-join hunt
=====
BGP Router ID:192.0.2.43      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
=====
BGP MVPN-IPv4 Routes
=====
-----
RIB In Entries
-----
```

Sender PE MVPN status

```
Route Type       : Source-Join
Route Dist.      : 64496:10
Source AS        : 64496
Source IP        : 172.16.1.2
Group IP         : 239.255.0.1
Nexthop          : 192.0.2.45
From             : 192.0.2.13
Res. Nexthop     : 0.0.0.0
Local Pref.      : 100
Aggregator AS    : None
Atomic Aggr.     : Not Atomic
AIGP Metric      : None
Connector        : None
Community        : target:64496:10 target:192.0.2.43:2
Cluster          : 192.0.2.13
Originator Id    : 192.0.2.45
Peer Router Id   : 192.0.2.13
Flags            : Used Valid Best IGP
Route Source     : Internal
AS-Path          : No As-Path
Neighbor-AS      : N/A
VPRN Imported    : 10
```

RIB Out Entries

Routes : 1
=====

The PIM status for this group on sender PE-1 is shown below:

```
*B:PE-1# show router 10 pim group 239.255.0.1 source 172.16.1.2 detail
```

=====

PIM Source Group ipv4

```
=====
Group Address      : 239.255.0.1
Source Address     : 172.16.1.2
RP Address         : 10.10.10.11
Advt Router        : 192.0.2.43
Flags              : spt
MRIB Next Hop      : 172.16.1.2
MRIB Src Flags     : direct
Keepalive Timer Exp: 0d 00:00:20
Up Time            : 0d 03:47:22
Type               : (S,G)
Resolved By        : rtable-u

Up JP State        : Joined
Up JP Rpt          : Not Joined StarG
Up JP Expiry       : 0d 00:00:00
Up JP Rpt Override : 0d 00:00:00

Register State     : Pruned
Register Stop Exp  : 0d 00:00:12
Reg From Anycast RP: No

Rpf Neighbor       : 172.16.1.2
Incoming Intf      : int-PE-1-CE-1
Outgoing Intf List : mpls-if-73829

Curr Fwding Rate   : 21504.0 kbps
Forwarded Packets  : 26603799
Forwarded Octets   : 35755505856
Spt threshold      : 0 kbps
Discarded Packets  : 0
RPF Mismatches     : 0
ECMP opt threshold : 7
```

```
Admin bandwidth      : 1 kbps
```

```
-----
Groups : 1
=====
```

The outgoing interface list is the I-PMSI, and traffic is seen to be flowing as the current forwarding rate is non-zero.

Similarly for sender PE-3, the MVPN status is:

```
*B:PE-3# show router 10 mvpn
```

```
=====
MVPN 10 configuration data
```

```
=====
signaling          : Bgp                auto-discovery      : Default
UMH Selection      : Highest-Ip
intersite-shared   : Enabled              Persist SA         : Enabled
vrf-import         : mvpn_10_import
vrf-export         : mvpn_10_export
vrf-target         : N/A
C-Mcast Import RT  : target:192.0.2.61:3
```

```
ipmsi              : ldp
i-pmsi P2MP AdmSt  : Up
i-pmsi Tunnel Name : mpls-if-73819
Mdt-type           : sender-receiver
```

```
s-pmsi             : none
data-delay-interval: 3 seconds
enable-asm-mdt     : N/A
```

The Source-Join route on PE-3 for this multicast group is:

```
*B:PE-3# show router bgp routes mvpn-ipv4 type source-ad detail
```

```
=====
BGP Router ID:192.0.2.61      AS:64496      Local AS:64496
```

```
=====
Legend -
```

```
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best, b - backup
```

```
=====
BGP MVPN-IPv4 Routes
```

```
=====
Route Type      : Source-Ad
Route Dist.     : 64496:10
Source IP       : 172.16.1.2
Group IP        : 239.255.0.1
Nexthop         : 192.0.2.43
From            : 192.0.2.13
Res. Nexthop    : 0.0.0.0
Local Pref.     : 100
Aggregator AS   : None
Atomic Aggr.    : Not Atomic
Interface Name  : NotAvailable
Aggregator      : None
MED             : 0
```

Sender PE MVPN status

```
AIGP Metric      : None
Connector        : None
Community        : 64496:452 target:64496:10
Cluster          : 192.0.2.13
Originator Id    : 192.0.2.43      Peer Router Id : 192.0.2.13
Flags            : Used Valid Best IGP
Route Source     : Internal
AS-Path          : No As-Path
Neighbor-AS      : N/A
VPRN Imported    : 10
```

```
-----
Routes : 1
=====
```

The PIM state for this group is seen below:

```
*B:PE-3# show router 10 pim group 239.255.0.1 source 172.16.3.2 detail
=====
PIM Source Group ipv4
=====
Group Address      : 239.255.0.1
Source Address     : 172.16.3.2
RP Address         : 10.10.10.10
Advt Router        : 192.0.2.61
Flags              : spt                      Type              : (S,G)
MRIB Next Hop      : 172.16.3.2
MRIB Src Flags     : direct
Keepalive Timer Exp: 0d 00:00:13
Up Time            : 0d 03:50:59      Resolved By           : rtable-u

Up JP State        : Joined              Up JP Expiry          : 0d 00:00:00
Up JP Rpt          : Not Joined StarG    Up JP Rpt Override    : 0d 00:00:00

Register State     : Pruned              Register Stop Exp     : 0d 00:00:29
Reg From Anycast RP: No

Rpf Neighbor       : 172.16.3.2
Incoming Intf      : int-PE3-CE-3
Outgoing Intf List : mpls-if-73819

Curr Fwding Rate   : 10703.9 kbps
Forwarded Packets  : 13518483            Discarded Packets     : 0
Forwarded Octets   : 18168841152         RPF Mismatches        : 0
Spt threshold      : 0 kbps              ECMP opt threshold    : 7
Admin bandwidth    : 1 kbps
-----
Groups : 1
=====
```

Note that the preferred source remains active unless:

- The multicast source ceases to exist, the source PE withdraws the Source Active AD route
- Or a Source Active AD is received with a higher local preference.

Conclusion

MVPN Source Redundancy provides an optimal solution for multicast routing in a VPRN. This protocol provides simple configuration, operation and guaranteed fast protection time. It could be utilized in a regionalized IPTV solution where multiple sources for the same TV channel are used.

Spoke Termination for IPv6-6VPE

In This Chapter

This section provides information about spoke termination for IPv6-6VPE.

Topics in this section include:

- [Applicability on page 1684](#)
- [Overview on page 1685](#)
- [Configuration on page 1688](#)
- [Conclusion on page 1713](#)

Applicability

Spoke termination for IPv6 is applicable to all of the 7750 SR family, plus 7450 ESS in mixed-mode. Chassis modes B (with mixed mode enabled), C or D need to be enabled to support IPv6 and VPNv6 address families. Spoke termination for IPv6 is supported for both IES and 6VPE services and has been tested for this note on 6VPE on 7750 SR-OS R9.0.

Overview

RFC 4659, *BGP-MPLS IP Virtual Private Network (VPN) Extension for IPv6 VPN*, standardized the use of an IPv6 over IPv4 tunneling scheme. The 7750 SR supports the standardized IPv6 over IPv4 tunneling scheme for VPRN services using Multi-Protocol Border Gateway Protocol (MP-BGP), also known as 6VPE. The 7750 SR also supports pseudowire termination by a VPRN from an Epipe Virtual Leased Line (VLL) or VPLS spoke SDP where the pseudowire can be given IPv6 addresses and run IPV6 protocols. In the example used in this section, any advertisements across the Multi-Protocol Labeled Service (MPLS) network between Virtual Private Routed Network (VPRN) Provider Edge (PE) devices will use 6VPE. The goal of this section is to list configuration guidelines for IPv6 spoke termination to a VPRN over an Epipe VLL and transporting IPv6 packets over 6VPE tunnels between PE devices.

This solution is to be used where a service provider is providing VPRN services built on a transport network whose Interior Gateway Protocol (IGP) is using IPv4 addressing on the network interfaces. The customer's CE and the service provider's PE must support IPv6 pseudowires, IPv6 interfaces and in addition, the service provider also be able to support the advertisements of IPv6 prefixes between CE-PE peerings and between the transport PE routers using MP-BGP. The advertisement of IPv6 prefixes across the MPLS network and the transport of IPv6 traffic is tunneled using 6VPE.

The VPRN PE has the ability to support spoke termination of Epipe VLL services on access with IPv6 addressing between the CE and VPRN PE. The functions of IPv6 spoke termination on VPRN services have the same functionality as VPRN IPv4 spoke termination prior to R8.0.

The example in [Figure 239](#) illustrates a CE device that connects to a VPRN PE on an IPv6 interface addressing using spoke termination.

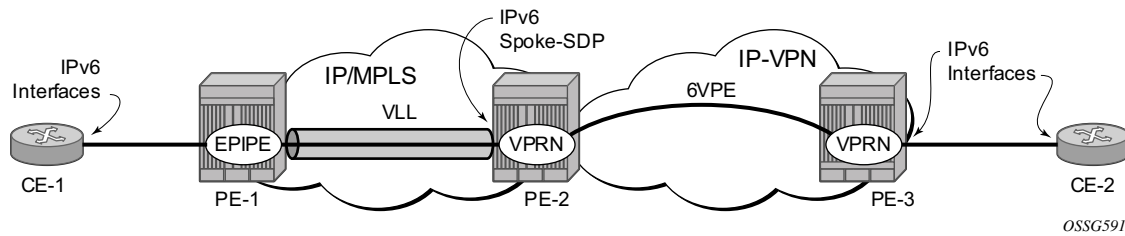


Figure 239: Spoke Termination for IPv6

CE-1 is connected to the VPRN service on PE-2, using IPv6 interfaces. CE-1 reaches PE-2 by connecting to PE-1. PE-1 uses an Epipe VLL for transport to the VPRN on the PE-2. The connectivity between PE-1's VLL service and PE-2's VPRN service is using spoke termination with IPv6 addressing on the PE-2's spoke-service distribution point (spoke SDP) interface.

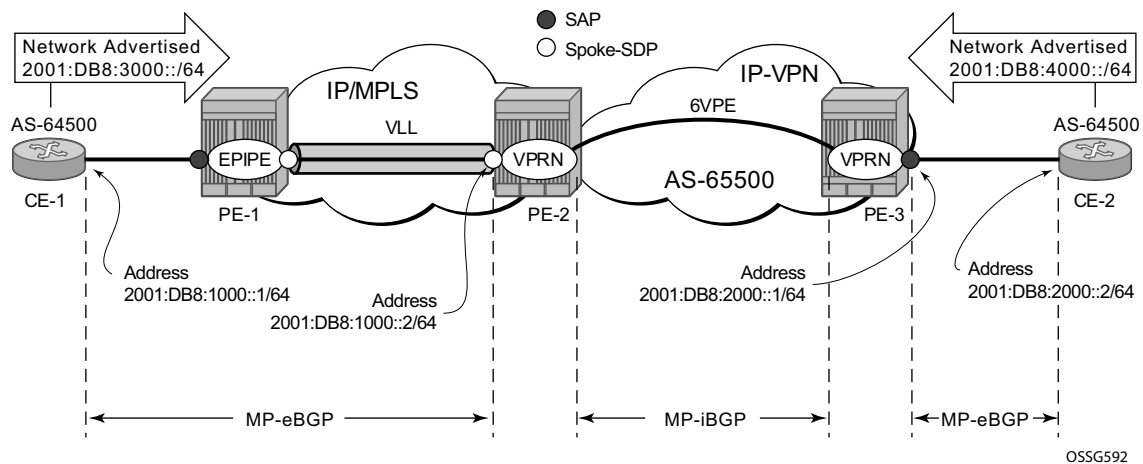


Figure 240: IPv6 Addressing and IPv6 Prefixes

Figure 240 shows the overall IPv6 addressing from interfaces to prefixes advertised from CE-1 and CE-2 across the VPRN network.

- Link between CE-1 and PE-2: 2001:DB8:1000::/64
- Link between CE-2 and PE-3: 2001:DB8:2000::/64
- Advertised Prefix from CE-1: 2001:DB8:3000::/64
- Advertised Prefix from CE-2: 2001:DB8:4000::/64

PE-2 has an MP-eBGP session with CE-1 to receive and advertise IPv6 routes. PE-2 also has an MP-iBGP peering session with PE-3 to use 6VPE to tunnel IPv6 routes and traffic to and from PE-3. PE-3 has an IPv6 SAP interface to CE-2 and uses MP-eBGP to advertise and receive routes to/from CE-2 (no spoke-termination). PE-3's configuration will also be included to provide examples of the end-to-end VPRN service using a 6VPE model.

This network topology will illustrate the use of spoke termination using IPv6 interfaces and the tunneling of IPv6 traffic over 6VPE MPLS network.

Configuration

First is to configure and establish an MPLS network where the VPRN service can use 6VPE to tunnel traffic across the IPv4 IGP.

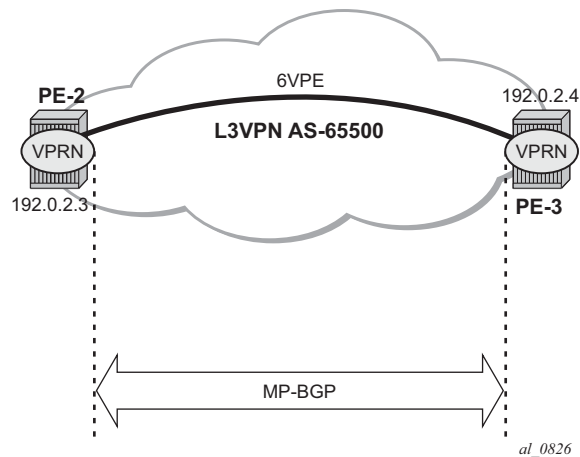


Figure 241: MP-BGP VPNv6

PE-2 and PE-3 in [Figure 241](#) are edge routers running VPRN Services on access with IPv6 Interfaces. The MPLS network is configured using IPv4 link addressing. Interior Border Gateway Protocol (iBGP) peerings need to be established with MP-BGP for VPN-IPv6 address families between PE-2 and PE-3.

```
A:PE-2>config>router>bgp# info
    hold-time 10
    router-id 192.0.2.3
    group "iBGP_AS65500"
        description "iBGP_peerings_AS65500"
        family vpn-ipv6
        peer-as 65500
        local-address 192.0.2.3
        neighbor 192.0.2.4
            description "PE-3"
        exit
    exit
exit
```

```
A:PE-3>config>router>bgp# info
    hold-time 10
    router-id 192.0.2.4
    group "iBGP_AS65500"
        description "iBGP_peerings_AS65500"
        family vpn-ipv6
        peer-as 65500
```

```

        local-address 192.0.2.4
        neighbor 192.0.2.3
            description "PE-2"
        exit
    exit

```

Configuring family vpn-ipv6 between VPRN PE edge routers in BGP turns on the functionality of MP-BGP for the Layer 3 VPNs supporting the customer's IPv6 Addressing (6VPE).

Verify BGP sessions for VPN-IPv6 address families between the PE-2 and PE-3.

```
A:PE-2# show router bgp neighbor 192.0.2.4
```

BGP Neighbor

```

Peer   : 192.0.2.4
Group  : iBGP_AS65500

```

Peer AS	: 65500	Peer Port	: 49521
Peer Address	: 192.0.2.4		
Local AS	: 65500	Local Port	: 179
Local Address	: 192.0.2.3		
Peer Type	: Internal		
State	: Established	Last State	: Established
Last Event	: recvKeepAlive		
Last Error	: Cease		
Local Family	: VPN-IPv6		
Remote Family	: VPN-IPv6		

```
A:PE-3# show router bgp neighbor 192.0.2.3
```

BGP Neighbor

```

Peer   : 192.0.2.3
Group  : iBGP_AS65500

```

Peer AS	: 65500	Peer Port	: 179
Peer Address	: 192.0.2.3		
Local AS	: 65500	Local Port	: 49521
Local Address	: 192.0.2.4		
Peer Type	: Internal		
State	: Established	Last State	: Active
Last Event	: recvKeepAlive		
Last Error	: Cease		
Local Family	: VPN-IPv6		
Remote Family	: VPN-IPv6		

Configuration

Now that you have established MP-BGP sessions for VPN-IPv6 address-family, 6VPE tunnel support is provided between PE-2 and PE-3.

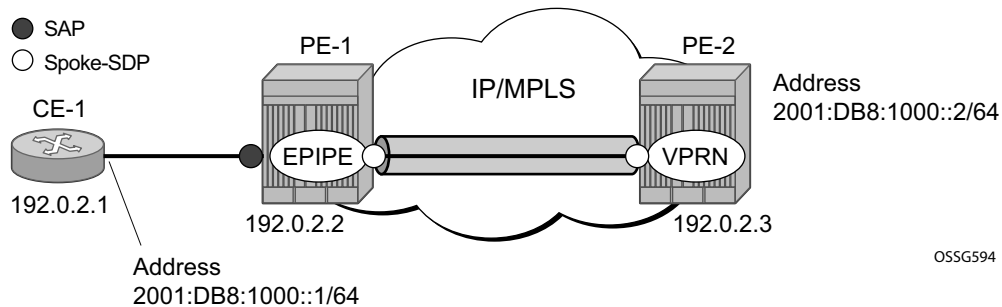


Figure 242: Spoke Termination for IPv6 Addressing

Figure 242 illustrates the model for spoke termination for IPv6 using VPRN services. CE-1 is configured with IPv6 addressing on the access interface facing the VPRN service. CE-1's access is backhauled to the VPRN service on PE-2 using Epipe VLL with spoke termination. Only Epipe VLL is supported for IPv6 spoke termination within the VPRN in R8.0. The configuration of the Epipe VLL on PE-1 is shown below:

```
A:PE-1>config>service# info
  customer 1 create
    description "Default customer"
  exit
  sdp 40 mpls create
    far-end 192.0.2.3
    ldp
    keep-alive
    shutdown
  exit
  no shutdown
exit
epipe 1 customer 1 create
  sap 1/1/4 create
  exit
  spoke-sdp 40:1 create
  exit
  no shutdown
exit
```

The example above is taken from PE-1 which has been configured using the Epipe VLL service with a SAP interface facing the customer and a spoke SDP facing PE-2. The spoke SDP is terminated into the customer's VPRN service on PE-2.

PE-2 is now ready to be configured for the IPv6 spoke SDP. Review the possible IPv6 options for spoke SDP interfaces on the CLI for VPRN Services (compliant to RFC 4213, *Basic Transition Mechanisms for IPv6 Hosts and Routers* <draft-ietf-v6ops-mech-v2-07.txt>):

- Interface spoke SDP (IPv6 options only)

```
A:PE-2>config>service>vprn$ interface <interface> create
A:PE-2>config>service>vprn>if$
[no] ipv6                + Enables/Configures IPv6 for a VPRN interface
A:PE-2>config>service>vprn>if$ ipv6
[no] address
[no] dhcp6-relay
[no] dhcp6-server
    icmp6
[no] link-local-address
[no] local-proxy-nd
[no] neighbor
[no] proxy-nd-policy
[no] vrrp
```

- IPv6 address

```
A:PE-2>config>service>vprn>if>ipv6# address
- address <ipv6-address/prefix-length> [eui-64] [preferred]
- no address <ipv6-address/prefix-length>

<ipv6-address/pref*> : ipv6-address  x:x:x:x:x:x:x:x (eight 16-bit pieces)
                                x:x:x:x:x:x:d.d.d.d
                                x [0..FFFF]H
                                d [0..255]D
                                prefix-length [1..128]
<eui-64>              : keyword
<preferred>          : keyword
```

- DHCPv6 relay parameters for the VPRN service

```
A:PE-2>config>service>vprn>if>ipv6>dhcp6-relay# info detail
shutdown
no description
no lease-populate
no neighbor-resolution
option
    no interface-id
    no remote-id
exit
no source-address
no server
```

- DHCPv6 server parameters for the VPRN service

```
A:PE-2>config>service>vprn>if>ipv6>dhcp6-server# info detail
prefix-delegation
shutdown
exit
max-nbr-of-leases 8000
```

- ICMPv6

```
A:PE-2>config>service>vprn>if>ipv6>icmp6# info detail
packet-too-big 100 10
param-problem 100 10
redirects 100 10
time-exceeded 100 10
unreachables 100 10
```

- Link-local-addressing, for the VPRN interface. Link-local addressing by default is assigned dynamically. Use this command if you would like to add a static link-local-address.

```
A:PE-2>config>service>vprn>if>ipv6# link-local-address
- link-local-address <ipv6-address> [preferred]
- no link-local-address

<ipv6-address>      : ipv6-address      - x:x:x:x:x:x:x:x
                                         x:x:x:x:x:x:d.d.d.d
                                         x [0..FFFF]H
                                         d [0..255]D

<preferred>        : keyword
```

- Neighbor

* IPv6 to MAC address mapping on the VRPN Interface

```
A:PE-2>config>service>vprn>if>ipv6# neighbor
- neighbor <ipv6-address> <mac-address>
- no neighbor <ipv6-address>

<ipv6-address>      : x:x:x:x:x:x:x:x   (eight 16-bit pieces)
                                         x:x:x:x:x:x:d.d.d.d
                                         x [0..FFFF]H
                                         d [0..255]D
                                         prefix-length [1..128]
<mac-address>       : xx:xx:xx:xx:xx:xx or xx-xx-xx-xx-xx-xx
```

- Enabling Local Proxy Neighbor Discovery

```
A:PE-2>config>service>vprn>if>ipv6# local-proxy-nd
- local-proxy-nd
- no local-proxy-nd
```

- VRRP

```
A:PE-2>config>service>vprn>if>ipv6# vrrp <virtual-router-id> [owner]
priority 100
no policy
preempt
master-int-inherit
no ping-reply
no telnet-reply
no traceroute-reply
no standby-forwarding
no mac
no init-delay
```



```

message-interval 1
no shutdown

```

PE-2 configuration for the VPRN service with IPv6 interface (spoke SDP) in reference to [Figure 242](#):

```

*A:PE-2>config>service>vprn# info
router-id 192.0.2.6
autonomous-system 65500
route-distinguisher 65500:1
auto-bind ldp
vrf-target target:65500:1
interface "loopback" create
    address 192.0.2.6/32
    loopback
exit
interface "Spoke_to_PE-1" create
    ipv6
        address 2001:DB8:1000::2/64
    exit
    spoke-sdp 40:1 create
    exit
exit
bgp
    router-id 192.0.2.6
    group "CE-1-PE-2-spoke"
        family ipv6
        local-as 65500
        peer-as 64500
        local-address 2001:DB8:1000::2
        neighbor 2001:DB8:1000::1
            as-override
            type external
            export "PE-2-BGP-CE-1"
        exit
    exit
exit
no shutdown

*A:PE-2>config>router>policy-options# info
policy-options
    begin
    prefix-list "PE-2-CE-1"
        prefix 2001:DB8:4000::/64 exact
    exit
    policy-statement "PE-2-BGP-CE-1"
        entry 10
            from
                prefix-list "PE-2-CE-1"
            exit
            action accept
            exit
        exit
    exit
commit
exit

```

Configuration

In the prior configuration example, PE-2 has been configured with an IPv6 spoke SDP (spoke termination) with interface spoke_to_PE-1. The VPRN configuration has also been set up for MP-eBGP peering to CE-1 through the IPv6 spoke interface. The MP-eBGP peering will be receiving and advertising IPv6 prefixes from/to CE-1. Route policy configuration has been included to show how IPv6 routes are advertised to CE-1 from PE-2 (policy-statement PE-2-BGP-CE-1).

On PE-1, verification that the Epipe VLL is established with the SAP facing CE-1 and spoke SDP facing VPRN on PE-2 can be seen as follows:

```
A:PE-1# show service id 1 all
```

Service Detailed Information

Service Id	: 1	Vpn Id	: 0
Service Type	: Epipe		
Name	: (Not Specified)		
Description	: (Not Specified)		
Customer Id	: 1		
Last Status Change	: 05/17/2010 21:45:56		
Last Mgmt Change	: 05/17/2010 21:38:10		
Admin State	: Up	Oper State	: Up
MTU	: 1514		
Vc Switching	: False		
SAP Count	: 1	SDP Bind Count	: 1
Per Svc Hashing	: Disabled		
Force Qtag Fwd	: Disabled		

Service Destination Points (SDPs)

Sdp Id 40:1 - (192.0.2.3)

Description	: (Not Specified)		
SDP Id	: 40:1	Type	: Spoke
Spoke Descr	: (Not Specified)		
VC Type	: Ether	VC Tag	: n/a
Admin Path MTU	: 1514	Oper Path MTU	: 1514
Far End	: 192.0.2.3	Delivery	: LDP
Hash Label	: Disabled		
Admin State	: Up	Oper State	: Up
Acct. Pol	: None	Collect Stats	: Disabled
Ingress Label	: 131069	Egress Label	: 131069
Ingr Mac Fltr-Id	: n/a	Egr Mac Fltr-Id	: n/a
Ingr IP Fltr-Id	: n/a	Egr IP Fltr-Id	: n/a
Ingr Ipv6 Fltr-Id	: n/a	Egr Ipv6 Fltr-Id	: n/a
Admin ControlWord	: Not Preferred	Oper ControlWord	: False
Admin BW(Kbps)	: 0	Oper BW(Kbps)	: 0
Last Status Change	: 05/17/2010 21:45:56	Signaling	: TLDP
Last Mgmt Change	: 05/17/2010 21:38:10	Force Vlan-Vc	: Disabled
Endpoint	: N/A	Precedence	: 4
Class Fwding State	: Down		
Flags	: None		
Peer Pw Bits	: None		
Peer Fault Ip	: None		
Peer Vccv CV Bits	: lspPing		
Peer Vccv CC Bits	: mplsRouterAlertLabel		
Application Profile	: None		

```

KeepAlive Information :
Admin State           : Disabled           Oper State           : Disabled
Hello Time            : 10                 Hello Msg Len        : 0
Max Drop Count        : 3                 Hold Down Time       : 10

Statistics            :
I. Fwd. Pkts.         : 231915             I. Dro. Pkts.        : 0
I. Fwd. Octs.         : 21994828           I. Dro. Octs.        : 0
E. Fwd. Pkts.         : 231914             E. Fwd. Octets       : 21992191
.
...
.
SAP 1/1/4

Service Id            : 1
SAP                   : 1/1/4               Encap                 : null
Description           : (Not Specified)     Oper State            : Up
Admin State           : Up
Flags                 : None
Multi Svc Site        : None
Last Status Change    : 05/17/2010 21:37:57
Last Mgmt Change      : 05/17/2010 21:07:20
Sub Type              : regular
Dot1Q Ethertype       : 0x8100             QinQ Ethertype        : 0x8100
Split Horizon Group   : (Not Specified)

LLF Admin State       : Down               LLF Oper State        : Clear
Admin MTU             : 9212               Oper MTU              : 9212
Ingr IP Fltr-Id       : n/a               Egr IP Fltr-Id        : n/a
Ingr Mac Fltr-Id      : n/a               Egr Mac Fltr-Id       : n/a
Ingr Ipv6 Fltr-Id     : n/a               Egr Ipv6 Fltr-Id      : n/a
tod-suite             : None               qinq-pbit-marking     : both
Ing Agg Rate Limit    : max                Egr Agg Rate Limit    : max
Endpoint              : N/A
Q Frame-Based Acct    : Disabled
Vlan-translation      : None

Acct. Pol             : None               Collect Stats         : Disabled
Application Profile    : None

Sap Statistics

Last Cleared Time     : N/A

                                Packets      Octets
Forwarding Engine Stats
Dropped               : 0                0
Off. HiPrio           : 0                0
Off. LowPrio          : 231919           22920335
Off. Uncolor          : 0                0

Queueing Stats(Ingress QoS Policy 1)
Dro. HiPrio           : 0                0
Dro. LowPrio          : 0                0
For. InProf           : 0                0
For. OutProf          : 231919           22920335

Queueing Stats(Egress QoS Policy 1)

```

Configuration

```
Dro. InProf      : 0                0
Dro. OutProf     : 0                0
For. InProf      : 231920           22922995
For. OutProf     : 0                0
```

Now, proceed to verify that the spoke termination on PE-2 in the VPRN using IPv6 addressing is established in an up/up state:

```
A:PE-2# show service id 1 all
Service Detailed Information
```

```
Service Id       : 1                Vpn Id           : 0
Service Type     : VPRN
Name             : (Not Specified)
Description      : (Not Specified)
Customer Id      : 1
Last Status Change: 05/17/2010 21:42:52
Last Mgmt Change : 05/17/2010 21:45:59
Admin State      : Up                Oper State      : Up

Route Dist.      : 65500:1           VPRN Type       : regular
AS Number        : 65500             Router Id       : 192.0.2.6
ECMP             : Enabled           ECMP Max Routes : 1
Max Ipv4 Routes  : No Limit          Auto Bind       : LDP
Max Ipv6 Routes  : No Limit
Ignore NH Metric : Disabled
Hash Label       : Disabled
Vrf Target       : target:65500:1
Vrf Import       : None
Vrf Export       : None
MVPN Vrf Target  : None
MVPN Vrf Import  : None
MVPN Vrf Export  : None

SAP Count        : 0                SDP Bind Count  : 1
```

```
Service Destination Points(SDPs)
```

```
Sdp Id 40:1 - (192.0.2.2)
```

```
Description      : (Not Specified)
SDP Id           : 40:1                Type           : Spoke
Spoke Descr      : (Not Specified)
VC Type          : n/a                VC Tag         : n/a
Admin Path MTU   : 1514               Oper Path MTU   : 1514
Far End          : 192.0.2.2          Delivery        : LDP

Admin State      : Up                Oper State      : Up
Acct. Pol        : None              Collect Stats    : Disabled
Ingress Label    : 131069            Egress Label    : 131069
Ingr Mac Fltr-Id : n/a                Egr Mac Fltr-Id : n/a
Ingr IP Fltr-Id  : n/a                Egr IP Fltr-Id  : n/a
Ingr Ipv6 Fltr-Id : n/a              Egr Ipv6 Fltr-Id : n/a
Admin ControlWord : Not Preferred     Oper ControlWord : False
Last Status Change : 05/17/2010 21:45:59
Last Mgmt Change  : 05/17/2010 21:45:59
Signaling        : n/a
```

Class Fwding State : Down
 Flags : None
 Peer Pw Bits : None
 Peer Fault Ip : None
 Peer Vccv CV Bits : lspPing
 Peer Vccv CC Bits : mplsRouterAlertLabel
 Application Profile: None

KeepAlive Information :

Admin State	: Disabled	Oper State	: Disabled
Hello Time	: 10	Hello Msg Len	: 0
Max Drop Count	: 3	Hold Down Time	: 10

Statistics :

I. Fwd. Pkts.	: 0	I. Dro. Pkts.	: 0
I. Fwd. Octs.	: 0	I. Dro. Octs.	: 0
E. Fwd. Pkts.	: 232421	E. Fwd. Octets	: 22042904

Number of SDPs : 1

Service Access Points

No Sap Associations

Service Interfaces

Interface

If Name	: loopback	Oper (v4/v6)	: Up/Down
Admin State	: Up		
Protocols	: None		
IP Addr/mask	: 192.0.2.6/32	Address Type	: Primary
IGP Inhibit	: Disabled	Broadcast Address	: Host-ones
Description	: N/A		

Details

Description	: (Not Specified)		
If Index	: 2	Virt. If Index	: 2
Last Oper Chg	: 05/17/2010 21:48:08	Global If Index	: 384
Port Id	: loopback		
TOS Marking	: Trusted	If Type	: VPRN
SNTP B.Cast	: False		
MAC Address	: 68:64:ff:00:00:00	Arp Timeout	: 14400
IP Oper MTU	: 1500	ICMP Mask Reply	: True
Arp Populate	: Disabled	Host Conn Verify	: Disabled
Cflowd	: None		
LdpSyncTimer	: None		
LSR Load Balance	: system		
uRPF Chk	: disabled		
uRPF Fail Bytes	: 0	uRPF Chk Fail Pkts:	0

Proxy ARP Details

Rem Proxy ARP	: Disabled	Local Proxy ARP	: Disabled
Policies	: none		

Proxy Neighbor Discovery Details

Local Pxy ND	: Disabled
Policies	: none

Configuration

DHCP no local server

DHCP Details

Description : (Not Specified)

Admin State : Down Lease Populate : 0
Gi-Addr : 192.0.2.6* Gi-Addr as Src Ip : Disabled
* = inferred gi-address from interface IP address

Action : Keep Trusted : Disabled

DHCP Proxy Details

Admin State : Down
Lease Time : N/A
Emul. Server : Not configured

Subscriber Authentication Details

Auth Policy : None

DHCP6 Relay Details

Description : (Not Specified)
Admin State : Down Lease Populate : 0
Oper State : Down Nbr Resolution : Disabled
If-Id Option : None Remote Id : Disabled
Src Addr : Not configured

DHCP6 Server Details

Admin State : Down Max. Lease States : 8000

ICMP Details

Redirects : Number - 100 Time (seconds) - 10
Unreachables : Number - 100 Time (seconds) - 10
TTL Expired : Number - 100 Time (seconds) - 10

IPCP Address Extension Details

Peer IP Addr : Not configured
Peer Pri DNS Addr : Not configured
Peer Sec DNS Addr : Not configured

Interface

If Name : Spoke_to_PE-1
Admin State : Up Oper (v4/v6) : Down/Up
Protocols : None
Ipv6 Addr : 2001:DB8:1000::2/64 PREFERRED
Ipv6 Addr : FE80::6A64:FFFF:FE00:0/64 PREFERRED
Description : N/A

Details

Description : (Not Specified)
If Index : 3 Virt. If Index : 3
Last Oper Chg : 05/17/2010 21:42:42 Global If Index : 383
SDP Id : spoke-40:1

Spoke-SDP Details

Admin State : Up Oper State : Up
Hash Label : Disabled
Peer Fault Ip : None

```

Peer Pw Bits      : None
Peer Vccv CV Bits : lspPing
Peer Vccv CC Bits : mplsRouterAlertLabel
Flags             : None

TOS Marking       : Trusted           If Type           : VPRN
SNTP B.Cast       : False
MAC Address       : 68:64:ff:00:00:00  Arp Timeout       : 14400
IP Oper MTU       : 1500               ICMP Mask Reply    : True
Arp Populate      : Disabled           Host Conn Verify   : Disabled
Cflowd           : None
LdpSyncTimer     : None
LSR Load Balance  : system
uRPF Chk         : disabled
uRPF Fail Bytes   : 0                  uRPF Chk Fail Pkts: 0

Proxy ARP Details
Rem Proxy ARP     : Disabled           Local Proxy ARP    : Disabled
Policies          : none

Proxy Neighbor Discovery Details
Local Pxy ND      : Disabled
Policies          : none

DHCP no local server

DHCP Details
Description       : (Not Specified)
Admin State       : Down               Lease Populate     : 0
Gi-Addr          : Not configured      Gi-Addr as Src Ip : Disabled
Action           : Keep                Trusted            : Disabled

DHCP Proxy Details
Admin State       : Down
Lease Time        : N/A
Emul. Server     : Not configured

Subscriber Authentication Details
Auth Policy       : None

DHCP6 Relay Details
Description       : (Not Specified)
Admin State       : Down               Lease Populate     : 0
Oper State       : Down               Nbr Resolution     : Disabled
If-Id Option     : None               Remote Id           : Disabled
Src Addr         : Not configured

DHCP6 Server Details
Admin State       : Down               Max. Lease States  : 8000

ICMP Details
Redirects         : Number - 100       Time (seconds)     - 10
Unreachables     : Number -100         Time (seconds)     - 10
TTL Expired      : Number -100         Time (seconds)     - 10

IPCP Address Extension Details
Peer IP Addr      : Not configured
Peer Pri DNS Addr : Not configured
Peer Sec DNS Addr : Not configured

```

Configuration

VPLS Sites

Site	Site-Id	Dest	Mesh-SDP	Admin	Oper	Fwdr
No Matching Entries						

It is important to note the following from PE-2 **show service** output above:

- VPRN service is in an admin/oper up/up state
- Spoke SDP is established to PE-1 (192.0.2.2) admin/oper up/up (IPv6) state.
 - IPv6 interface is established and its IPv6 address is preferred (2001:DB8:1000::2/64)
 - IPv6 link local address has been dynamically assigned and preferred (FE80::6A64:FFFF:FE00:0/64).
 - This output also lists other IPv6 fields that can be checked if configured: DHCP6-relay, DHCP6 server, etc.

After verification of the services (Epipe, VPRN), verify MP-eBGP peering connectivity (through IPv6 interfaces) on the VPRN between PE-2 and CE-1.

```
A:PE-2# show router 1 bgp neighbor
```

BGP Neighbor

```
Peer   : 2001:DB8:1000::1
Group  : CE-1-PE-2-spoke
```

Peer AS	: 64500	Peer Port	: 179
Peer Address	: 2001:DB8:1000::1		
Local AS	: 65500	Local Port	: 49723
Local Address	: 2001:DB8:1000::2		
Peer Type	: External		
State	: Established	Last State	: Active
Last Event	: recvKeepAlive		
Last Error	: Unrecognized Error		
Local Family	: Ipv6		
Remote Family	: Ipv6		
Hold Time	: 90	Keep Alive	: 30
Active Hold Time	: 10	Active Keep Alive	: 3
Cluster Id	: None		
Preference	: 170	Num of Update Flaps	: 0
Recd. Paths	: 1		
Ipv4 Recd. Prefixes	: 0	Ipv4 Active Prefixes	: 0
Ipv4 Suppressed Pfxs	: 0	VPN-Ipv4 Suppr. Pfxs	: 0
VPN-Ipv4 Recd. Pfxs	: 0	VPN-Ipv4 Active Pfxs	: 0
Mc Ipv4 Recd. Pfxs	: 0	Mc Ipv4 Active Pfxs	: 0
Mc Ipv4 Suppr. Pfxs	: 0	Ipv6 Suppressed Pfxs	: 0
Ipv6 Recd. Prefixes	: 1	Ipv6 Active Prefixes	: 1
VPN-Ipv6 Recd. Pfxs	: 0	VPN-Ipv6 Active Pfxs	: 0
VPN-Ipv6 Suppr. Pfxs	: 0	L2-VPN Suppr. Pfxs	: 0
L2-VPN Recd. Pfxs	: 0	L2-VPN Active Pfxs	: 0
MVPN-Ipv4 Suppr. Pfxs	: 0	MVPN-Ipv4 Recd. Pfxs	: 0


```

MVPN-Ipv4 Active Pfxs: 0
MDT-SAFI Recd. Pfxs : 0
Input Queue          : 0
i/p Messages         : 109157
i/p Octets            : 2074060
i/p Updates          : 1
TTL Security         : Disabled
Graceful Restart      : Disabled
Advertise Inactive    : Disabled
Advertise Label       : None
Auth key chain        : n/a
Bfd Enabled           : Disabled
Local Capability      : RtRefresh MPBGP 4byte ASN
Remote Capability     : RtRefresh MPBGP 4byte ASN
Import Policy         : None Specified / Inherited
Export Policy         : PE-2-BGP-CE-1

MDT-SAFI Suppr. Pfxs : 0
MDT-SAFI Active Pfxs : 0
Output Queue          : 0
o/p Messages          : 109704
o/p Octets             : 2084468
o/p Updates           : 1
Min TTL Value         : n/a
Stale Routes Time     : n/a
Peer Tracking         : Disabled

Neighbors : 1

```

It is important to note that not only is the MP-eBGP session on the VPRN established but that the MP-BGP capabilities are supported (locally and remotely).

Finally, the following configuration is an example of the VPRN service on PE-3 with SAP interfaces to CE-2, with MP-eBGP peering configured.

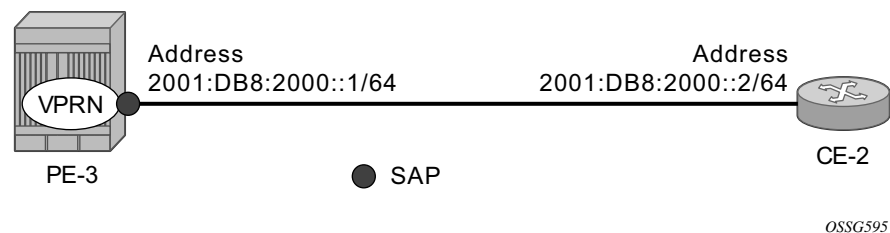


Figure 243: PE-3 VPRN with SAP to CE-2

Configuration

```
A:PE-3>config>service>vprn# info
    router-id 192.0.2.7
    autonomous-system 65500
    route-distinguisher 65500:1
    auto-bind ldp
    vrf-target target:65500:1
    interface "loopback" create
        address 192.0.2.7/32
        loopback
    exit
    interface "int-PE-3-CE-2" create
        ipv6
            address 2001:DB8:2000::1/64
        exit
        sap 1/1/2 create
        exit
    exit
    bgp
        router-id 192.0.2.7
        group "CE-2-PE-3"
            family ipv6
            local-as 65500
            peer-as 64500
            local-address 2001:DB8:2000::1
            neighbor 2001:DB8:2000::2
                as-override
                type external
                export "PE-3-BGP-CE-2"
            exit
        exit
    exit
no shutdown

A:PE3>config>router>policy-options# info
    policy-options
        begin
            prefix-list "PE-3-CE-2"
                prefix 2001:DB8:3000::/64 exact
            exit
            policy-statement "PE-3-BGP-CE-2"
                entry 10
                    from
                        prefix-list "PE-3-CE-2"
                    exit
                    action accept
                    exit
                exit
            exit
        exit
    commit
exit
```

The IPv6 configuration options for the SAP interface (int-PE-3-CE-2) are similar to those in the above example for the spoke SDP on PE-2. PE-3 BGP export policy (PE-3-BGP-CE-2) is also similar to the example for PE-2 in advertising the learned IPv6 route to CE-2.

Verification of the configuration of the VPRN service and MP-eBGP peering connectivity to CE-2 is shown below:

A:PE-3# show service id 1 all

Service Detailed Information

```

Service Id      : 1                      Vpn Id      : 0
Service Type    : VPRN
Name            : (Not Specified)
Description     : (Not Specified)
Customer Id     : 1
Last Status Change: 05/19/2010 14:10:48
Last Mgmt Change  : 05/19/2010 14:10:48
Admin State     : Up                    Oper State    : Up

Route Dist.     : 65500:1                VPRN Type     : regular
AS Number       : 65500                  Router Id     : 192.0.2.7
ECMP            : Enabled                 ECMP Max Routes : 1
Max Ipv4 Routes : No Limit               Auto Bind     : LDP
Max Ipv6 Routes : No Limit
Ignore NH Metric : Disabled
Hash Label      : Disabled
Vrf Target      : target:65500:1
Vrf Import      : None
Vrf Export      : None
MVPN Vrf Target : None
MVPN Vrf Import : None
MVPN Vrf Export : None

SAP Count       : 1                      SDP Bind Count : 0

```

Service Destination Points(SDPs)

No Matching Entries

Service Access Points

SAP 1/1/2

```

Service Id      : 1
SAP             : 1/1/2                  Encap         : null
Description     : (Not Specified)
Admin State     : Up                    Oper State    : Up
Flags          : None
Multi Svc Site  : None
Last Status Change : 05/19/2010 11:20:22
Last Mgmt Change  : 05/19/2010 14:16:54
Sub Type        : regular
Dot1Q Ethertype : 0x8100                QinQ Ethertype : 0x8100
Split Horizon Group: (Not Specified)

Admin MTU       : 9212                  Oper MTU      : 9212
Ingr IP Fltr-Id : n/a                   Egr IP Fltr-Id : n/a
Ingr Mac Fltr-Id : n/a                   Egr Mac Fltr-Id : n/a
Ingr Ipv6 Fltr-Id : n/a                  Egr Ipv6 Fltr-Id : n/a
tod-suite       : None                   qinq-pbit-marking : both
Ing Agg Rate Limit : max                  Egr Agg Rate Limit: max

```

Configuration

Q Frame-Based Acct : Disabled

Acct. Pol : None

Collect Stats : Disabled

Anti Spoofing : None

Avl Static Hosts : 0

Tot Static Hosts : 0

Calling-Station-Id : n/a

Application Profile: None

Sap Statistics

Last Cleared Time : N/A

	Packets	Octets
Forwarding Engine Stats		
Dropped	: 0	0
Off. HiPrio	: 0	0
Off. LowPrio	: 0	0
Off. Uncolor	: 0	0
Queueing Stats(Ingress QoS Policy 1)		
Dro. HiPrio	: 0	0
Dro. LowPrio	: 0	0
For. InProf	: 0	0
For. OutProf	: 0	0
Queueing Stats(Egress QoS Policy 1)		
Dro. InProf	: 0	0
Dro. OutProf	: 0	0
For. InProf	: 141983	14042605
For. OutProf	: 0	0

Service Interfaces

Interface

If Name	: loopback		
Admin State	: Up	Oper (v4/v6)	: Up/Down
Protocols	: None		
IP Addr/mask	: 192.0.2.7/32	Address Type	: Primary
IGP Inhibit	: Disabled	Broadcast Address	: Host-ones
Description	: N/A		

Details

Description	: (Not Specified)		
If Index	: 2	Virt. If Index	: 2
Last Oper Chg	: 05/19/2010 14:12:24	Global If Index	: 384
Port Id	: loopback		
TOS Marking	: Trusted	If Type	: VPRN
SNTP B.Cast	: False		
MAC Address	: 68:67:ff:00:00:00	Arp Timeout	: 14400
IP Oper MTU	: 1500	ICMP Mask Reply	: True
Arp Populate	: Disabled	Host Conn Verify	: Disabled
Cflowd	: None		
LdpSyncTimer	: None		
LSR Load Balance	: system		
uRPF Chk	: disabled		
uRPF Fail Bytes	: 0	uRPF Chk Fail Pkts:	0

```

Proxy ARP Details
Rem Proxy ARP      : Disabled          Local Proxy ARP   : Disabled
Policies           : none

Proxy Neighbor Discovery Details
Local Pxy ND       : Disabled
Policies           : none

DHCP no local server

DHCP Details
Description        : (Not Specified)
Admin State        : Down              Lease Populate     : 0
Gi-Addr           : 192.0.2.7*        Gi-Addr as Src Ip : Disabled
* = inferred gi-address from interface IP address

Action            : Keep              Trusted           : Disabled

DHCP Proxy Details
Admin State        : Down
Lease Time         : N/A
Emul. Server       : Not configured

Subscriber Authentication Details
Auth Policy        : None

DHCP6 Relay Details
Description        : (Not Specified)
Admin State        : Down              Lease Populate     : 0
Oper State         : Down              Nbr Resolution    : Disabled
If-Id Option       : None              Remote Id         : Disabled
Src Addr           : Not configured

DHCP6 Server Details
Admin State        : Down              Max. Lease States : 8000

ICMP Details
Redirects          : Number -100        Time (seconds)    - 10
Unreachables       : Number -100        Time (seconds)    - 10
TTL Expired        : Number -100        Time (seconds)    - 10

IPCP Address Extension Details
Peer IP Addr       : Not configured
Peer Pri DNS Addr  : Not configured
Peer Sec DNS Addr  : Not configured

Interface

If Name            : int-PE-3-CE-2
Admin State        : Up                Oper (v4/v6)       : Down/Up
Protocols          : None
IPv6 Addr          : 2001:DB8:2000::1/64      PREFERRED
IPv6 Addr          : FE80::6A67:FFFF:FE00:0/64 PREFERRED
Description        : N/A

Details
Description        : (Not Specified)

```

Configuration

```
If Index          : 3                      Virt. If Index    : 3
Last Oper Chg     : 05/19/2010 14:13:01  Global If Index   : 383
SAP Id            : 1/1/2
TOS Marking       : Trusted                If Type          : VPRN
SNTP B.Cast       : False
MAC Address       : 68:67:01:01:00:02     Arp Timeout      : 14400
IP Oper MTU       : 9198                  ICMP Mask Reply   : True
Arp Populate      : Disabled              Host Conn Verify  : Disabled
Cflowd           : None
LdpSyncTimer      : None
LSR Load Balance  : system
uRPF Chk          : disabled
uRPF Fail Bytes   : 0                    uRPF Chk Fail Pkts: 0

Proxy ARP Details
Rem Proxy ARP     : Disabled              Local Proxy ARP   : Disabled
Policies          : none

Proxy Neighbor Discovery Details
Local Pxy ND      : Disabled
Policies          : none

DHCP no local server

DHCP Details
Description       : (Not Specified)
Admin State       : Down                  Lease Populate    : 0
Gi-Addr          : Not configured         Gi-Addr as Src Ip : Disabled
Action           : Keep                  Trusted           : Disabled

DHCP Proxy Details
Admin State       : Down
Lease Time        : N/A
Emul. Server      : Not configured

Subscriber Authentication Details
Auth Policy       : None

DHCP6 Relay Details
Description       : (Not Specified)
Admin State       : Down                  Lease Populate    : 0
Oper State        : Down                  Nbr Resolution    : Disabled
If-Id Option      : None                  Remote Id         : Disabled
Src Addr          : Not configured

DHCP6 Server Details
Admin State       : Down                  Max. Lease States : 8000

ICMP Details
Redirects         : Number -100           Time (seconds)    - 10
Unreachables     : Number -100           Time (seconds)    - 10
TTL Expired      : Number -100           Time (seconds)    - 10

IPCP Address Extension Details
Peer IP Addr      : Not configured
Peer Pri DNS Addr : Not configured
Peer Sec DNS Addr : Not configured
```

At this point, VPRN access using spoke termination for IPv6 with MP-eBGP peering on PE-2, towards CE-1, has been established. VPRN access using a SAP with MP-eBGP peering on PE-3, towards CE-1, has also been established. MP-iBGP, providing 6VPE, has been configured and built between PE-2 and PE-3 across the MPLS Network. Now, propagate the advertisements of the IPv6 prefixes learned on PE-2 from CE-1 (2001:DB8:3000::/64) and on PE-3 from CE-2 (2001:DB8:4000::/64) across the MPLS network using MP-iBGP (6VPE).

Perform verification on PE-2 of routes learned and advertised to CE-1.

```
A:PE-2# show router bgp summary
BGP Router ID:192.0.2.3      AS:65500      Local AS:65500

BGP Admin State      : Up          BGP Oper State      : Up
Total Peer Groups    : 1           Total Peers          : 1
Total BGP Paths       : 6           Total Path Memory    : 752
Total Ipv4 Remote Rts : 0           Total Ipv4 Rem. Active Rts : 0
Total Ipv6 Remote Rts : 0           Total Ipv6 Rem. Active Rts : 0
Total Suppressed Rts  : 0           Total Hist. Rts      : 0
Total Decay Rts       : 0

Total VPN Peer Groups : 1           Total VPN Peers      : 1
Total VPN Local Rts   : 1
Total VPN-Ipv4 Rem. Rts : 0         Total VPN-Ipv4 Rem. Act. Rts: 0
Total VPN-Ipv6 Rem. Rts : 1         Total VPN-Ipv6 Rem. Act. Rts: 1
Total L2-VPN Rem. Rts  : 0           Total L2VPN Rem. Act. Rts : 0
Total VPN Supp. Rts    : 0           Total VPN Hist. Rts    : 0
Total VPN Decay Rts    : 0
Total MVPN-Ipv4 Rem Rts : 0         Total MVPN-Ipv4 Rem Act Rts : 0
Total MDT-SAFI Rem Rts : 0           Total MDT-SAFI Rem Act Rts : 0

BGP Summary

Neighbor
      AS PktRcvd InQ Up/Down State|Rcv/Act/Sent (Addr Family)
      PktSent OutQ

192.0.2.4
      65500 72529 0 01d23h54m 1/1/1 (VpnIPv6)
      72550 0
```

The above output shows the BGP Neighbor of CE-1 (192.0.2.4) and that PE-2 has received and learned an IPv6 prefix.

```
A:PE-2# show router 1 bgp routes ipv6
BGP Router ID:192.0.2.6      AS:65500      Local AS:65500

Legend
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : I - IGP, e - EGP, ? - incomplete, > - best

BGP Ipv6 Routes

Flag Network LocalPref MED
NextHop VPNLabel
```

Configuration

```
As-Path

u*>?  2001:DB8:3000::/64          None      None
      2001:DB8:1000::1
      64500

Routes : 1
```

The output above of the VPRN's BGP route-table for the IPv6 address-family lists the valid and best route for 2001:DB8:3000::/64 with a BGP next hop of 2001:DB8:1000::1 (CE-1).

```
A:PE-2# show router 1 bgp neighbor 2001:DB8:1000::1 advertised-routes ipv6

BGP Router ID:192.0.2.6      AS:65500      Local AS:65500

Legend
Status codes  : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes  : I - IGP, e - EGP, ? - incomplete, > - best

BGP Ipv6 Routes

Flag  Network                      LocalPref  MED
      Nexthop                      VPNLabel
      As-Path

?     2001:DB8:4000::/64          n/a        None
      2001:DB8:1000::2
      65500 65500

Routes : 1
```

The above output taken from PE-2 shows the advertised IPv6 prefix of 2001:DB8:4000::/64, originated and advertised from CE-2 to PE-3.

```
A:PE-2# show router bgp routes vpn-ipv6
BGP Router ID:192.0.2.3      AS:65500      Local AS:65500

Legend
Status codes  : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes  : I - IGP, e - EGP, ? - incomplete, > - best

BGP VPN-Ipv6 Routes

Flag  Network                      LocalPref  MED
      Nexthop                      VPNLabel
      As-Path

u*>?  65500:1:2001:DB8:4000::/64  100        None
      ::FFFF:C000:204
      64500                      131068

Routes : 1
```


PE-2 in the previous output has learned of prefix 2001:DB8:4000::/64 as an MP-BGP VPN-IPv6 route, with the VRF route-target of 65500:1 from PE-3.

In the output below PE-2 is advertising the VPN-IPv6 route of 2001:DB8:3000::/64 that was learned from CE-1 across the MP-iBGP Session to PE-3.

```
A:PE-2# show router bgp neighbor 192.0.2.4 advertised-routes vpn-ipv6
BGP Router ID:192.0.2.3      AS:65500      Local AS:65500
```

Legend

Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : I - IGP, e - EGP, ? - incomplete, > - best

BGP VPN-IPv6 Routes

Flag	Network	LocalPref	MED
	Nexthop		VPNLabel
	As-Path		
?	65500:1:2001:DB8:3000::/64	100	None
	::FFFF:C000:203		131068
	64500		

Routes : 1

Verify VPN-IPv6 routes on PE-3.

```
A:PE-3# show router bgp routes vpn-ipv6
BGP Router ID:192.0.2.4      AS:65500      Local AS:65500

Legend
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : I - IGP, e - EGP, ? - incomplete, > - best

BGP VPN-IPv6 Routes

Flag  Network                               LocalPref  MED
      Nexthop                             VPNLabel
      As-Path

u*>?  65500:1:2001:DB8:3000::/64          100        None
      ::FFFF:C000:203                    131068
      64500

Routes : 1
```

The output above, taken from PE-3, lists the VPN-IPv6 route that is being learned from PE-2, namely 2001:DB8:3000::/64.

The output show below lists the advertised VPN-IPv6 route of 2001:DB8:4000::/64 from PE-3 to PE-2.

```
A:PE-3# show router bgp neighbor 192.0.2.3 advertised-routes vpn-ipv6
BGP Router ID:192.0.2.4      AS:65500      Local AS:65500

Legend
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : I - IGP, e - EGP, ? - incomplete, > - best

BGP VPN-IPv6 Routes

Flag  Network                               LocalPref  MED
      Nexthop                             VPNLabel
      As-Path

?      65500:1:2001:DB8:4000::/64          100        None
      ::FFFF:C000:204                    131068
      64500

Routes : 1
```

Verify IPv6 prefixes learned and advertised between PE-3 and CE-2.

```
A:PE-3# show router 1 bgp routes ipv6
BGP Router ID:192.0.2.7      AS:65500      Local AS:65500

Legend
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : I - IGP, e - EGP, ? - incomplete, > - best
```

BGP Ipv6 Routes

Flag	Network	LocalPref	MED
	Nexthop		VPNLabel
	As-Path		
u*>	2001:DB8:4000::/64	None	None
	2001:DB8:2000::2		-
	64500		

Routes : 1

From the PE-3 output above shows the IPv6 prefix of 2001:DB8:4000::/64 learned from CE-2.

The output below verifies the advertisement of IPv6 prefix 2001:DB8:3000::/64 to CE-2.

```
A:PE-3# show router 1 bgp neighbor 2001:DB8:2000::2 advertised-routes ipv6
BGP Router ID:192.0.2.7          AS:65500          Local AS:65500
```

Legend

Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
 Origin codes : I - IGP, e - EGP, ? - incomplete, > - best

BGP Ipv6 Routes

Flag	Network	LocalPref	MED
	Nexthop		VPNLabel
	As-Path		
?	2001:DB8:3000::/64	n/a	None
	2001:DB8:2000::1		-
	65500 65500		

Routes : 1

Perform the final verification of CE-1 and CE-2 showing that IPv6 routes for AS-64500 have been received and are valid across the VPRN service.

```
A:CE-1# show router bgp routes ipv6
BGP Router ID:192.0.2.1          AS:64500          Local AS:64500
```

Legend

Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
 Origin codes : I - IGP, e - EGP, ? - incomplete, > - best

BGP Ipv6 Routes

Flag	Network	LocalPref	MED
	Nexthop		VPNLabel
	As-Path		
u*>?	2001:DB8:4000::/64	None	None
	2001:DB8:1000::2		-
	65500 65500		

Configuration

Routes : 1

A:CE-2# show router bgp routes ipv6

BGP Router ID:192.0.2.5 AS:64500 Local AS:64500

Legend

Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid

Origin codes : I - IGP, e - EGP, ? - incomplete, > - best

BGP Ipv6 Routes

Flag	Network	LocalPref	MED
	Nexthop		VPNLabel
	As-Path		
u*>?	2001:DB8:3000::/64	None	None
	2001:DB8:2000::1		-
	65500 65500		

Routes : 1

Conclusion

Spoke termination for IPv6-6VPE extends the use of spoke terminated interfaces from an EPIPE VLL into a VPRN service using IPv6 interfaces on the access. Supporting the requirement of IPv6 interfaces, routing of IPv6 prefixes and the use of 6VPE for IPv6 tunnelling over an IPv4 network allows the 7750 SR to provide capabilities to support the growth of IPv6 architectures. This chapter provides examples of this feature with **show** commands for guidance.

VPRN Inter-AS VPN Model C

In This Chapter

This section provides following information.

Topics in this section include:

- [Applicability on page 1716](#)
- [Overview on page 1717](#)
- [Configuration on page 1720](#)
- [Conclusion on page 1729](#)

Applicability

This example is applicable to all of the 7750 and 7710 SR series and was tested on release 7.0R5. There are no pre-requisites for this configuration. This is supported on 7450 ESS-7 or ESS-12 in mixed-mode since 8.0R1. The 7750 SR-c4 is supported from 8.0R4 and higher.

Overview

Introduction

Section 10 of RFC 4364, *BGP/MPLS IP Virtual Private Networks (VPNs)*, describes three potential methods for service providers to interconnect their IP-VPN (Internet Protocol — Virtual Private Network) backbones in order to provide an end-to-end MPLS-VPN where one or more sites of the VPN are connected to different service provider autonomous systems. The purpose of this section is to describe the configuration and troubleshooting for inter-AS VPN model C.

In this architecture, VPN prefixes are neither held, nor re-advertised by the Autonomous System Border Router — Provider Edge (ASBR-PE) routers. The ASBR-PE does however maintain labeled IPv4 /32 routes to other PE routers within its own AS. It then redistributes these /32 IPv4 prefixes in external Border Gateway Protocol (eBGP) to the ASBR-PE in other service providers ASs. Using this methodology, it is possible for PE routers in different ASs to establish multi-hop Multi Protocol — external Border Gateway Protocol (MP-eBGP) sessions to each other in order to exchange customer VPN prefixes over those connections.

To be more specific, the /32 IPv4 routes for the PE routers in the other service providers AS will need to be redistributed into the interior Gateway Protocol (IGP) in the local AS together with an assigned label. As most service providers do not like redistribution of loop-back addresses from another service provider into the local IGP, a potential solution can be found by imposing a three-level label stack on the ingress PE. The bottom-level label would be assigned by the egress PE (advertised in multi-hop MP-eBGP without next-hop override) and is commonly referred to as the VPN-label. The middle label would be assigned by the local ASBR-PE and would correspond to the /32 route of the egress PE (in a different AS) using BGP-LBL (RFC 3107, *Carrying Label Information in BGP-4*). The top level label would then be the label assigned by the local ASBR-PE(s) /32 loop-back address, which would be assigned by the IGP next-hop of the ingress PE. This label is referred to as the LDP-LBL. [Figure 244](#) reflects this mechanism. The VPN-LBL is assigned by PE-5, the BGP-LBL is assigned by PE-4 and the LDP-LBL is also assigned by PE-4.

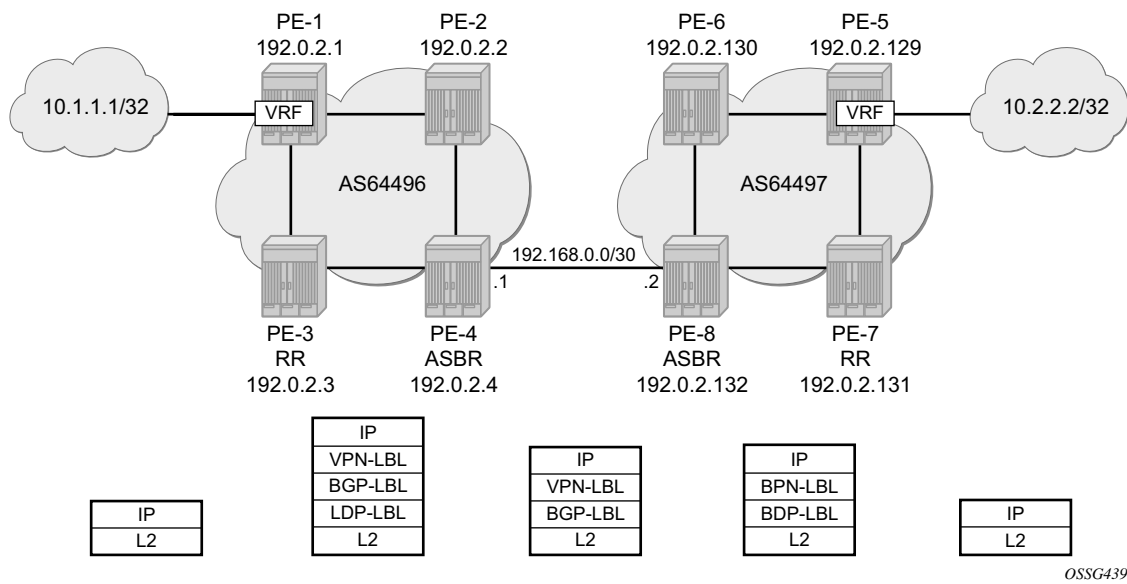


Figure 244: Inter-AS VPN Model C

The VPN connectivity is established using Labeled VPN route exchange using MP-eBGP without next-hop override. The PE connectivity will be established as described below.

EBGP PE /32 loopback leaking routing exchange using eBGP LBL (RFC 3107) at the ASBR-PE. The /32 PE routes learned from the other AS through the ASBR-PE are further distributed into the local AS using iBGP and optionally Route Reflectors (RRs). This model uses a three label stack and is referred to as Model C. Resilience for ASBR-PE failures is dependent on BGP.

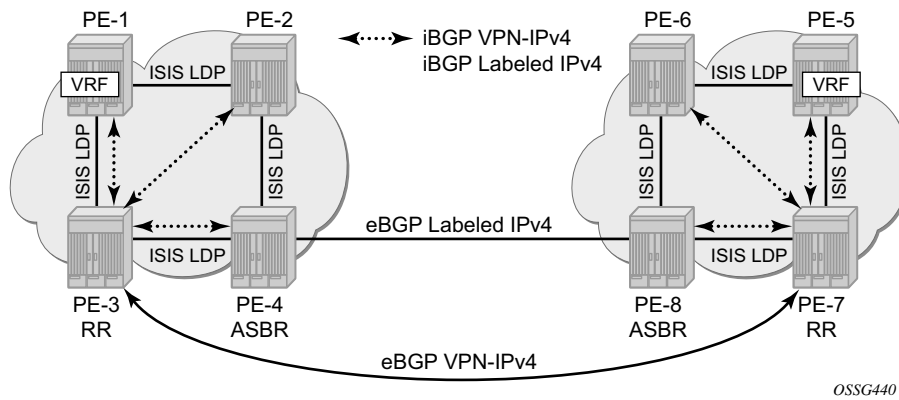


Figure 245: Protocol Overview

[Figure 245](#) gives an overview of all protocols used when implementing Inter-AS Model C. Inside each AS there is an ISIS adjacency and a link LDP session between each pair of adjacent nodes. As an alternative, OSPF can be used as IGP. Also there is an iBGP session between each PE and the RR. The address family is both VPN-IPv4 for the exchange of customer VPN prefixes and Labeled IPv4 for the exchange of labeled IPv4 prefixes. Note that as an alternative, a full mesh of iBGP sessions can be used in each AS.

Between the ASBRs there is an eBGP sessions for the exchange of labeled IPv4 prefixes. The ASBRs will override the next-hop for those prefixes. Between the RRs in the different ASs there is an eBGP session for the exchange of VPN customer prefixes. The RRs will not override the next-hop for those prefixes.

The big advantage of this model is that no VPN routes need to be held on the ASBR-PEs and as such it scales the best among all the three Inter-AS IP-VPN models. However, leaking /32 PE addresses between service providers creates some security concerns. As such we see Model C typically deployed within a service provider network.

The network topology is displayed in [Figure 244](#). The setup consists of two times four (2 x 4) 7750/7710 nodes located in different autonomous systems. There is an AS interconnection from ASBR PE-4 to ASBR PE-8. PE-3 and PE-7 will act as RRs for their AS. It is assumed that an IP-VPN is already configured in each AS. Following configuration tasks should be done first:

- ISIS or OSPF on all interfaces within each of the ASs.
- LDP on all interfaces within each of the ASs.
- MP-iBGP sessions between the PE routers and the RRs in each of the ASs.
- IP-VPN on PE-1 and on PE-5 with identical route targets.
- A loopback interface in the VRF on PE-1 and PE-5.

Configuration

The first step is to configure a MP-eBGP session between the ASBRs in both ASs. This session will be used to redistribute labelled IPv4 routes for the /32 system IP addresses between the AS?. These MP-BGP extensions are described in RFC 3107.

The configuration for ASBR PE-4 is displayed below. The **advertise-label ipv4** command is required to enable the advertising of labelled IPv4 routes. Note that this command is also required on the RR neighbor in order to propagate the labelled IPv4 routes towards the other PEs in the AS. The address family for labelled IPv4 routes is IPv4 so this family must be enabled for the peering with the RR.

```
configure router bgp
  group "rr"
    family ipv4 vpn-ipv4
    neighbor 192.0.2.3
      advertise-label ipv4
    exit
  exit
  group "remote-as"
    family ipv4
    type external
    peer-as 64497
    neighbor 192.168.0.2
      advertise-label ipv4
    exit
  exit
exit all
```

Note that address **family vpn-ipv4** is also required to advertise IPv4 customer routes within the AS. On the RR, the **advertise-label ipv4** command must be specified for each PE neighbor. Also note that address family IPv4 must be enabled. The configuration for RR PE-3 is displayed below.

```
configure router bgp
  group "rr-clients"
    family ipv4 vpn-ipv4
    neighbor 192.0.2.1
      advertise-label ipv4
    exit
    neighbor 192.0.2.2
      advertise-label ipv4
    exit
    neighbor 192.0.2.4
      advertise-label ipv4
    exit
  exit
exit all
```

On the remaining PE nodes in AS 64496, the **advertise-label ipv4** command must be specified on the RR neighbor. Also the IPv4 family must be enabled.

```
configure router bgp
  group rr
    family ipv4 vpn-ipv4
    neighbor 192.0.2.3
      advertise-label ipv4
    exit
  exit
exit all
```

The configuration for the nodes in AS64497 is very similar. The IP addresses can be derived from [Figure 244](#).

On ASBR PE-4, verify that the BGP session with ASBR PE-8 is up:

```
A:PE-4# show router bgp neighbor 192.168.0.2
=====
BGP Neighbor
=====
-----
Peer   : 192.168.0.2
Group  : remote-as
-----
-----
Peer AS           : 64497           Peer Port        : 179
Peer Address      : 192.168.0.2
Local AS          : 64496           Local Port       : 51262
Local Address     : 192.168.0.1
Peer Type         : External
State             : Established     Last State       : Active
Last Event        : recvKeepAlive
Last Error        : Cease
Local Family      : IPv4
Remote Family     : IPv4
Hold Time         : 90              Keep Alive      : 30
Active Hold Time  : 90              Active Keep Alive : 30
Cluster Id        : None
Preference        : 170             Num of Flaps     : 4
Recd. Paths       : 0
IPv4 Recd. Prefixes : 0             IPv4 Active Prefixes : 0
IPv4 Suppressed Pfxs : 0             VPN-IPv4 Suppr. Pfxs : 0
VPN-IPv4 Recd. Pfxs : 0             VPN-IPv4 Active Pfxs : 0
Mc IPv4 Recd. Pfxs. : 0             Mc IPv4 Active Pfxs. : 0
Mc IPv4 Suppr. Pfxs : 0             IPv6 Suppressed Pfxs : 0
IPv6 Recd. Prefixes : 0             IPv6 Active Prefixes : 0
VPN-IPv6 Recd. Pfxs : 0             VPN-IPv6 Active Pfxs : 0
VPN-IPv6 Suppr. Pfxs : 0             L2-VPN Suppr. Pfxs  : 0
L2-VPN Recd. Pfxs  : 0             L2-VPN Active Pfxs  : 0
MVPN-IPv4 Suppr. Pfxs : 0           MVPN-IPv4 Recd. Pfxs : 0
MVPN-IPv4 Active Pfxs : 0
Input Queue       : 0               Output Queue     : 0
i/p Messages      : 37              o/p Messages     : 39
i/p Octets        : 891             o/p Octets       : 891
i/p Updates       : 4               o/p Updates      : 4
TTL Security      : Disabled        Min TTL Value    : n/a
```

Configuration

```
Graceful Restart      : Disabled          Stale Routes Time    : n/a
Advertise Inactive    : Disabled          Peer Tracking         : Disabled
Advertise Label       : ipv4
Auth key chain        : n/a
Bfd Enabled           : Disabled
Local Capability       : RtRefresh MPBGP 4byte ASN
Remote Capability     : RtRefresh MPBGP 4byte ASN
Import Policy         : None Specified / Inherited
Export Policy         : None Specified / Inherited
```

```
-----
Neighbors : 1
=====
```

```
A:PE-4#
```

Note that both ASBRs have MPBGP capabilities. At this time, no prefixes have been received from the remote ASBR. To enable the advertising of labelled IPv4 routes for the system loopback interfaces, an export policy must be created and applied to the BGP session on both ASBRs. The policy configuration is displayed below for ASBR PE-4. Note that the configuration for ASBR PE-8 is very similar, the IP addresses can be derived from [Figure 244](#).

```
configure router policy-options
  prefix-list "pe_sys"
    prefix 192.0.2.128/25 longer
  exit
  policy-statement "pe-sys-to-bgp"
    entry 10
      from
        prefix-list "pe-sys"
      exit
      to
        protocol bgp
      exit
      action accept
      exit
    exit
  exit
exit all
configure router bgp
  group remote-as
    neighbor 192.168.0.2
      export "pe-sys-to-bgp"
    exit
  exit
exit all
```

After creating and applying the export policies on both ASBRs, labelled IPv4 routes will be advertised towards the remote AS for system IP addresses of the PE nodes in the local AS.

On ASBR PE-4, verify if labelled IPv4 routes have been received from ASBR PE-8:

```
A:PE-4# show router bgp routes
=====
BGP Router ID:192.0.2.4          AS:64496          Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best
=====
BGP IPv4 Routes
=====
Flag   Network                               LocalPref  MED
      Nexthop                               VPNLabel
      As-Path
-----
u*>i  192.0.2.129/32                          None       20
      192.168.0.2
      64497
u*>i  192.0.2.130/32                          None       10
      192.168.0.2
      64497
u*>i  192.0.2.131/32                          None       10
      192.168.0.2
      64497
u*>?  192.0.2.132/32                          None       None
      192.168.0.2
      64497
-----
Routes : 4
=====
A:PE-4#
```

As can be seen from the output above, 4 labelled IPv4 routes have been received. One route for every system IP address in the remote AS with a label attached.

The actual labels can be seen with following command:

```
A:PE-4# show router bgp routes 192.0.2.129/32 hunt
=====
BGP Router ID:192.0.2.4          AS:64496          Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best
=====
BGP IPv4 Routes
=====
RIB In Entries
-----
Network       : 192.0.2.129/32
Nexthop       : 192.168.0.2
From          : 192.168.0.2
Res. Nexthop  : 192.168.0.2
Local Pref.   : None
Aggregator AS : None
Interface Name : int-PE-4-PE-8
Aggregator    : None
```

Configuration

```
Atomic Aggr.   : Not Atomic           MED           : 20
Community     : No Community Members
Cluster       : No Cluster Members
Originator Id : None                  Peer Router Id : 192.0.2.132
IPv4 Label    : 131065
Flags         : Used Valid Best IGP
AS-Path       : 64497
```

RIB Out Entries

```
-----
Network       : 192.0.2.129/32
Nexthop       : 192.0.2.4
To            : 192.0.2.3
Res. Nexthop  : n/a
Local Pref.   : 100
Aggregator AS : None                  Interface Name : NotAvailable
Atomic Aggr.  : Not Atomic           Aggregator    : None
Community     : No Community Members MED           : 20
Cluster       : No Cluster Members
Originator Id : None                  Peer Router Id : 192.0.2.3
IPv4 Label    : 131062
Origin        : IGP
AS-Path       : 64497
-----
```

Routes : 2

=====

A:PE-4#

Note that in the RIB In entries, the received label from PE-8 can be seen (131065). In the RIB Out entries, the locally assigned label for this prefix can be seen (131062). The label mapping can also be seen with following command:

A:PE-4# show router bgp inter-as-label

BGP Inter-AS labels

```
=====
NextHop                Received      Advertised      Label
                        Label          Label          Origin
-----
192.0.2.1              0              131065          Internal
192.168.0.2            131064         131061          External
192.168.0.2            131065         131062          External
192.168.0.2            131066         131060          External
192.168.0.2            131067         131063          External
192.0.2.2              0              131064          Internal
192.0.2.3              0              131066          Internal
192.0.2.4              0              131067          Edge
=====
```

A:PE-4#

Verify that the routes have been installed in the routing table:

```
A:PE-4# show router route-table
=====
Route Table (Router: Base)
=====
Dest Prefix                                Type   Proto   Age           Pref
      Next Hop[Interface Name]                                Metric
-----
192.0.2.1/32                               Remote  ISIS    02h24m15s     18
      192.168.3.1                                           20
192.0.2.2/32                               Remote  ISIS    02h24m15s     18
      192.168.3.1                                           10
192.0.2.3/32                               Remote  ISIS    02h27m29s     18
      192.168.4.1                                           10
192.0.2.4/32                               Local   Local   02h27m35s      0
      system                                                0
192.0.2.129/32                             Remote  BGP      00h03m54s    170
      192.168.0.2                                           0
192.0.2.130/32                             Remote  BGP      00h03m54s    170
      192.168.0.2                                           0
192.0.2.131/32                             Remote  BGP      00h03m54s    170
      192.168.0.2                                           0
192.0.2.132/32                             Remote  BGP      00h03m54s    170
      192.168.0.2
...
=====
A:PE-4#
```

Verify that the BGP routes are further advertised towards all the PEs in the AS (through the RR) and are installed in the routing table on all PEs by using the above command on the other PEs.

At this point, all PEs in one AS have the /32 system IPs of the remote PEs in their routing table. All PEs in one AS have also received labels for all /32 system IPs of the remote PEs. Now a MP-eBGP session can be created between the RRs in the different ASs to exchange VPN-IPv4 routes.

The configuration for RR PE-3 is displayed below. The configuration for RR PE-7 is very similar. The IP addresses can be derived from [Figure 245](#).

```
configure router bgp
  group "remote-as-rr"
    family vpn-ipv4
    multihop 10
    peer-as 64497
    neighbor 192.0.2.131
  exit
exit
exit all
```

On the RRs, verify that the MP-eBGP session is up:

```
A:PE-3# show router bgp neighbor 192.0.2.131
=====
BGP Neighbor
=====
-----
Peer      : 192.0.2.131
Group     : remote-as-rr
-----
Peer AS           : 64497           Peer Port           : 179
Peer Address      : 192.0.2.131
Local AS          : 64496           Local Port           : 49714
Local Address     : 192.0.2.3
Peer Type         : External
State            : Established      Last State           : Active
Last Event        : recvKeepAlive
Last Error        : Unrecognized Error
Local Family      : VPN-IPv4
Remote Family     : VPN-IPv4
Hold Time         : 90              Keep Alive           : 30
Active Hold Time  : 90              Active Keep Alive    : 30
Cluster Id        : None
Preference        : 170             Num of Flaps         : 0
Recd. Paths       : 1
IPv4 Recd. Prefixes : 0             IPv4 Active Prefixes : 0
IPv4 Suppressed Pfxs : 0             VPN-IPv4 Suppr. Pfxs : 0
VPN-IPv4 Recd. Pfxs : 1             VPN-IPv4 Active Pfxs : 0
Mc IPv4 Recd. Pfxs. : 0             Mc IPv4 Active Pfxs. : 0
Mc IPv4 Suppr. Pfxs : 0             IPv6 Suppressed Pfxs : 0
IPv6 Recd. Prefixes : 0             IPv6 Active Prefixes : 0
VPN-IPv6 Recd. Pfxs : 0             VPN-IPv6 Active Pfxs : 0
VPN-IPv6 Suppr. Pfxs : 0            L2-VPN Suppr. Pfxs   : 0
L2-VPN Recd. Pfxs   : 0            L2-VPN Active Pfxs   : 0
MVPN-IPv4 Suppr. Pfxs : 0           MVPN-IPv4 Recd. Pfxs : 0
MVPN-IPv4 Active Pfxs : 0
Input Queue        : 0              Output Queue         : 0
i/p Messages       : 14             o/p Messages         : 14
i/p Octets         : 370            o/p Octets           : 370
i/p Updates        : 1              o/p Updates          : 1
TTL Security       : Disabled        Min TTL Value         : n/a
Graceful Restart   : Disabled        Stale Routes Time     : n/a
Advertise Inactive : Disabled        Peer Tracking         : Disabled
Advertise Label    : None
Auth key chain     : n/a
Bfd Enabled        : Disabled
Local Capability    : RtRefresh MPBGP ORFSendExComm ORFRecvExComm 4byte ASN
Remote Capability   : RtRefresh MPBGP ORFSendExComm ORFRecvExComm 4byte ASN
Import Policy       : None Specified / Inherited
Export Policy       : None Specified / Inherited
-----
Neighbors : 1
=====
A:PE-3#
```

The BGP session is established. Note that 1 VPN-IPv4 prefix has been received for the remote AS.

Now the VPRNs on PE-1 in AS64496 and PE-5 in AS64497 are interconnected. Packets originating in AS 64496 with a destination in AS 64497 will have 3 labels in AS 64496. Originate a VPRN ping on PE-1 towards the VPRN loopback IP address on PE-5:

```
A:PE-1# ping router 1 10.2.2.2
PING 10.2.2.2 56 data bytes
64 bytes from 10.2.2.2: icmp_seq=1 ttl=64 time=7.50ms.
64 bytes from 10.2.2.2: icmp_seq=2 ttl=64 time=3.77ms.
64 bytes from 10.2.2.2: icmp_seq=3 ttl=64 time=3.80ms.
64 bytes from 10.2.2.2: icmp_seq=4 ttl=64 time=3.77ms.
64 bytes from 10.2.2.2: icmp_seq=5 ttl=64 time=3.78ms.

---- 10.2.2.2 PING Statistics ----
5 packets transmitted, 5 packets received, 0.00% packet loss
round-trip min = 3.77ms, avg = 4.52ms, max = 7.50ms, stddev = 1.49ms
```

The top label is the LDP label to reach the exit point of the AS (PE-4). This label can be seen with following command on PE-1:

```
A:PE-1# show router ldp bindings prefix 192.0.2.4/32 active
=====
Legend: (S) - Static
=====
LDP Prefix Bindings (Active)
=====
Prefix                Op    IngLbl    EgrLbl    EgrIntf/LspId  EgrNextHop
-----
192.0.2.4/32          Push   --        131069    1/1/2          192.168.1.2
192.0.2.4/32          Swap  131068    131069    1/1/2          192.168.1.2
-----
No. of Prefix Bindings: 2
=====
A:PE-1#
```

The middle label is the label assigned by MP-BGP on the local ASBR-PE to reach the remote PE in the remote AS. This label can be seen with following command on PE-1:

```
A:PE-1# show router bgp routes 192.0.2.129/32 hunt
=====
BGP Router ID:192.0.2.1      AS:64496      Local AS:64496
=====
Legend -
Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes  : i - IGP, e - EGP, ? - incomplete, > - best
=====
BGP IPv4 Routes
=====
RIB In Entries
-----
Network      : 192.0.2.129/32
Nexthop      : 192.0.2.4
From         : 192.0.2.3
Res. Nexthop  : 192.168.1.2
Local Pref.   : 100
Interface Name : int-PE-1-PE-2
```

Configuration

```
Aggregator AS   : None                      Aggregator      : None
Atomic Aggr.    : Not Atomic                MED             : 20
Community       : No Community Members
Cluster         : 1.1.1.1
Originator Id   : 192.0.2.4                  Peer Router Id  : 192.0.2.3
IPv4 Label      : 131062
Flags           : Used Valid Best IGP
AS-Path         : 64497
```

RIB Out Entries

Routes : 1

=====

```
A:PE-1#
```

The bottom label is the VPN label assigned by the remote PE in the remote AS for the destination network. This label can be seen with following command on PE-1:

```
A:PE-1# show router bgp routes vpn-ipv4 10.2.2.2/32 hunt
```

```
=====
```

BGP Router ID:192.0.2.1	AS:64496	Local AS:64496
-------------------------	----------	----------------

```
=====
```

Legend -

Status codes : u - used, s - suppressed, h - history, d - decayed, * - valid
Origin codes : i - IGP, e - EGP, ? - incomplete, > - best

=====

```
BGP VPN-IPv4 Routes
```

RIB In Entries

```
-----
```

Network	: 10.2.2.2/32	
Nexthop	: 192.0.2.129	
Route Dist.	: 64497:1	VPN Label : 131070
From	: 192.0.2.3	
Res. Nexthop	: n/a	
Local Pref.	: 100	Interface Name : NotAvailable
Aggregator AS	: None	Aggregator : None
Atomic Aggr.	: Not Atomic	MED : None
Community	: target:64496:1	
Cluster	: No Cluster Members	
Originator Id	: None	Peer Router Id : 192.0.2.3
Flags	: Used Valid Best IGP	
AS-Path	: 64497	
VRPN Imported	: 1	

```
-----
```

RIB Out Entries

Routes : 1

=====

```
A:PE-1#
```

Conclusion

Inter-AS option C allows the delivery of Layer 3 VPN services to customers who have sites connected multiple ASs. This example shows the configuration of inter-AS option C (specific to this feature) together with the associated show output which can be used verify and troubleshoot it.

Quality of Service

In This Section

This section provides configuration information for the following topics:

- [Class Fair Hierarchical Policing for SAPs on page 1733](#)
- [Pseudowire QoS on page 1777](#)
- [QoS Architecture and Basic Operation on page 1803](#)

Class Fair Hierarchical Policing for SAPs

In This Chapter

This section provides information to configure Class Fair Hierarchical Policing for SAPs.

Topics in this section include:

- [Applicability on page 1734](#)
- [Summary on page 1735](#)
- [Overview on page 1736](#)
- [Configuration on page 1747](#)
- [Conclusion on page 1776](#)

Applicability

The information in this note is applicable to all of the Alcatel-Lucent 7x50 platforms and is focused on the FP2 chipset, which is used in the IOM3-XP/IMMs and in the 7750 SRc-12/4. The configuration was tested on release 9.0R1. There are no specific pre-requisites for this configuration.

Summary

The Quality of Service (QoS) features of the 7x50 platforms provide traffic control with both shaping and policing.

Shaping is achieved using a queue; packets are placed on the queue and a scheduler removes packets from the queue at a given rate. This provides an upper bound to the traffic rate sent, thereby protecting downstream devices from bursts. However, shaping can introduce latency and jitter as packets are delayed in the queue. Packets can be dropped when the queue is full or statistically when weighted random early discard is applied. Configuration of shaping on the 7x50 is described in [QoS Architecture and Basic Operation on page 1803](#).

Policing is another mechanism for controlling traffic rates but it does not introduce latency/jitter. This is achieved using a token bucket mechanism which drops certain packets from the traffic. A common disadvantage of policing implementations is that they are usually applicable to a single level of traffic priority and have no way to fairly share capacity between multiple streams at the same priority level. Alcatel-Lucent's Class Fair Hierarchical Policing (CFHP) addresses these problems by implementing a four level prioritized policing hierarchy which also provides weighted fairness for traffic at a given priority.

Regardless of whether shaping or policing is being used, the preceding QoS classification and subsequent packet marking functionality is similar for both and is covered in more detail in [QoS Architecture and Basic Operation on page 1803](#).

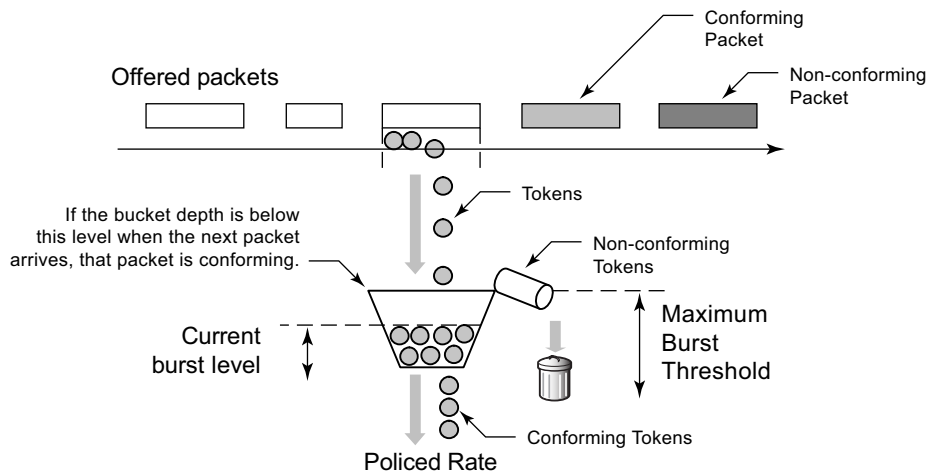
This note describes the configuration and operation of CFHP when applied to Service Access Points (SAPs). It is also possible to use CFHP for subscribers in a Triple Play Service Delivery Architecture (TPSDA) environment but it is beyond the scope of this note.

Overview

Policers

CFHP can be used both for ingress and egress QoS. The basic element is a policer which can apply both a committed information rate (CIR) and peak information rate (PIR) to a traffic flow (determined by the ingress classification). Traffic is directed to a policer by assigning a forwarding class (FC) to the policer.

To describe the operation of a policer we will use a token bucket model, this is shown in [Figure 246](#).



OSSG513

Figure 246: Policer Token Bucket Model

The policer is modeled by a bucket being filling with tokens which represent the bytes in the packets passing through the policer. The bucket drains at a given rate (the policed rate) and if the token (byte) arrival rate exceeds the drain rate then the bucket will fill. The bucket has a maximum depth, defined by a maximum burst threshold. If tokens for a packet arrive in the bucket when the current burst level of tokens is below the maximum burst threshold then the packet is considered to be conforming and all of its tokens are accepted into the bucket. If a packet's tokens arrive when the current burst level has exceeded the maximum burst threshold then none its tokens are accepted into the bucket and the packet is considered to be non-conforming (in the representation, these tokens over-flow into a waste bin).

Table 12 shows an example of the two possibilities.

Table 12: Burst Levels

Maximum burst threshold = 2000 tokens (bytes) Policed rate 2 Mbps = 250000 bytes/sec (250 tokens/ms)				
Arrival Time	Packet Size	Current Burst Level	Conforming Packet	New Burst Level
T_0	1024	1500	Yes	$1500 + 1024 = 2524$
$T_0 + 1\text{ms}$	128	$2524 - 250 = 2274$	No	2274

When the first packet arrives the current burst level is below the maximum burst threshold so it is conforming, however, when the second packet arrives the current burst level is above the maximum burst threshold so it is non-conforming.

An important aspect of the implementation of hierarchical policing is the ability of a policer bucket to have multiple burst thresholds. The tokens for each arriving packet are only compared against a single threshold relating to the characteristics of packet. These burst thresholds allow specific granular QoS control.

Policer Buckets

A policer uses up to 3 buckets depending on its configuration. A PIR bucket to control the traffic rate which is always used though its rate could be max, there can be an optional CIR bucket if a CIR rate is defined for dynamically profiling (in-profile/out-of-profile) packets, finally there may be a fair information rate (FIR) bucket used to maintain traffic fairness in a hierarchical policing scenario when multiple child policers are configured at the same parent priority level.

The PIR bucket is drained at the PIR rate and has two burst thresholds, one for high burst priority traffic (defined by the maximum burst size (MBS)) and a second for low burst priority traffic (defined by the MBS minus high-prio-only), see [Figure 247](#). The traffic burst priority is determined at ingress by the configured priority of either high or low, and at the egress by the profile state of the packets (in-profile=high, out-of-profile=low). Note that by default all FCs are low burst priority. If a packet conforms at the PIR bucket (its tokens enter the bucket) then the packet is forwarded, otherwise the packet is discarded. Discarding logically results in the packet's tokens not being placed into the CIR, FIR or parent policer buckets.

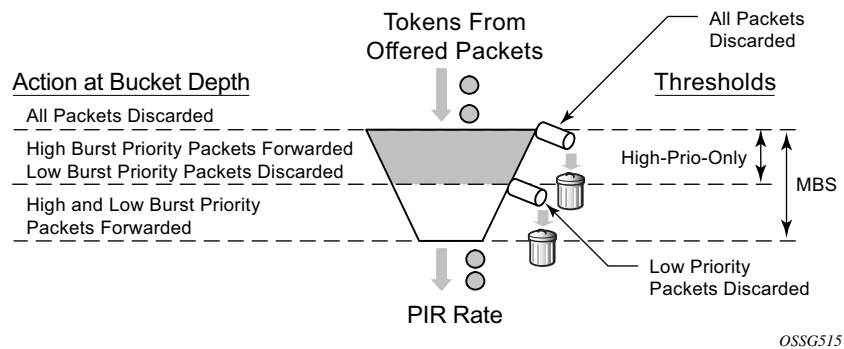


Figure 247: Peak Information Rate (PIR) Bucket

The CIR bucket is drained at the CIR rate and has one configurable burst threshold (defined by the committed burst size (CBS)). At the ingress, if the bucket level is below this threshold traffic is determined to be in-profile so the only action of the CIR bucket is to set the state of dynamically profiled packets to be either in-profile or out-of-profile. At the egress, re-profiling only affects Dot1P and DEI (Layer 2) egress marking (if the frame is double tagged, only the outer VLAN tag is remarked).

The CBS threshold is used when operating in color-blind mode, the profile of incoming packets is undefined and dynamically set based on the current burst level in the CIR bucket compared to the CBS threshold. It is also possible to operate (simultaneously) in color-aware mode, where the classification of incoming packets is used to explicitly determine whether a packet is in-profile or out-of-profile. For color-aware mode, the CIR bucket does not change the packet profile state.

In order to ensure that the overall amount of in-profile traffic takes into account both the explicit and dynamic in-profile packets, tokens from the explicit in-profile packets are allowed to fill the bucket above the CBS threshold. By doing this, dynamically profiled packets are only marked as in-profile after the token level representing dynamically in-profile and explicit in-profile packets have fallen below the CBS threshold (as the bucket drains). Note that explicitly marked out-of-profile packets remain out-of-profile, so the bottom of the bucket can be considered to be an implicit burst threshold for these packets. This is shown in [Figure 248](#).

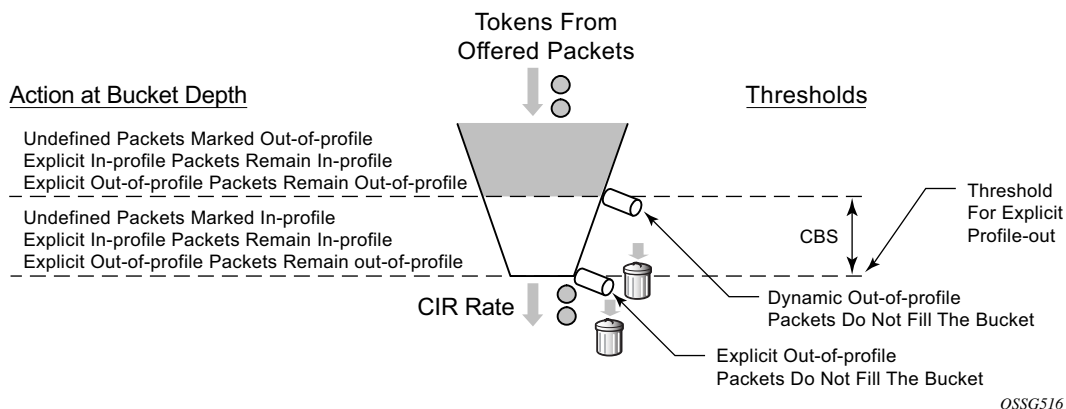


Figure 248: Committed Information Rate (CIR) Bucket

As the depths of the PIR and CIR buckets (MBS and CBS, respectively) are configured independently it is possible to have, for example, the CBS to be larger than the MBS (which is not possible for a queue). This could result in traffic being discarded because it is non-conforming at the PIR bucket but would have been conforming at the CIR bucket. Conversely, if the CBS is smaller than the MBS and the PIR=CIR traffic can be forwarded as out-of-profile, which would not be the case with a queue.

The FIR bucket is controlled by the system and is only used in hierarchical policing scenarios to determine a child's fair access to the available capacity at a parent priority level relative to other children at the same level. This bucket is only used when there is more than one child policer assigned to a given parent policer priority level. The drain rate of the FIR bucket is dynamically set proportionally to the weight configured for the child. This is shown in [Figure 249](#).

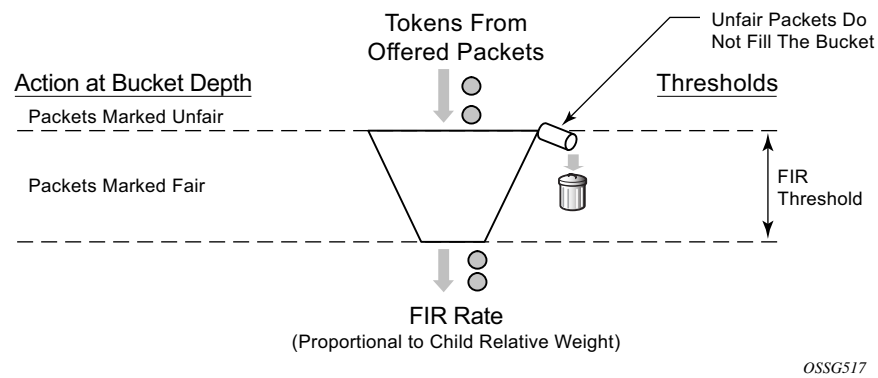


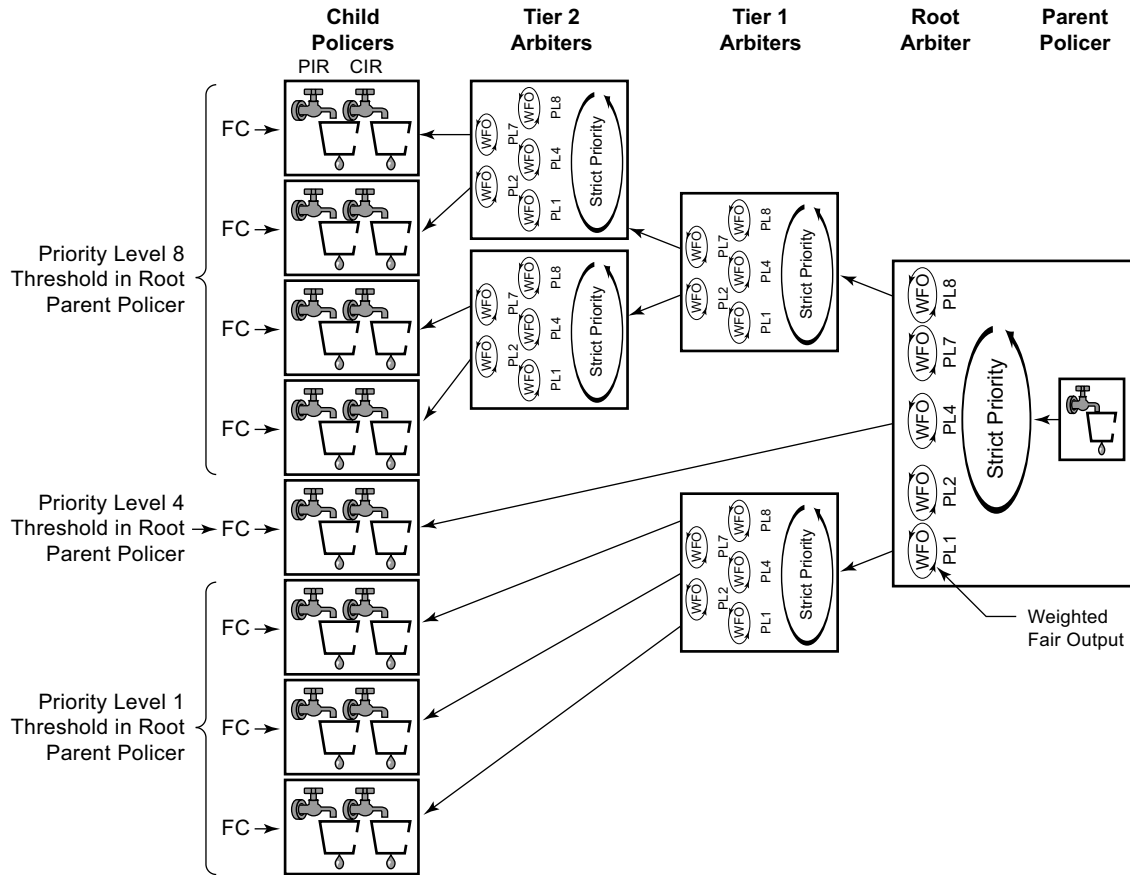
Figure 249: Fair Information Rate (FIR) Bucket

Hierarchical Policing

Policers can be used standalone or with a parent policer to provide hierarchical policing. Up to four stages can be configured in the hierarchy: the child policer, tier 1 and 2 intermediate arbiters, and a root arbiter (which is associated with the parent policer). The arbiters are logical entities that distribute bandwidth at a particular tier to their children in a priority level order, see [Figure 250](#).

This may result in the drain rates for the child policer buckets being modified, so each child policer PIR and CIR bucket has an administrative rate value (what it is configured to) and an operational rate value (the current operating rate) based on the bandwidth distribution by the parent arbiters.

Each stage in the hierarchy connects to its parent at a priority level and a weight. There are eight available priorities which are serviced in a strict order (8 to 1, highest to lowest, respectively). The weight is used to define relative fairness when multiple children are configured in the same priority level. Note that the child access to parent policer burst capacity is governed by the level at which the child ultimately connects into the root arbiter, not by its connection level at any intermediate arbiters.



OSSG518

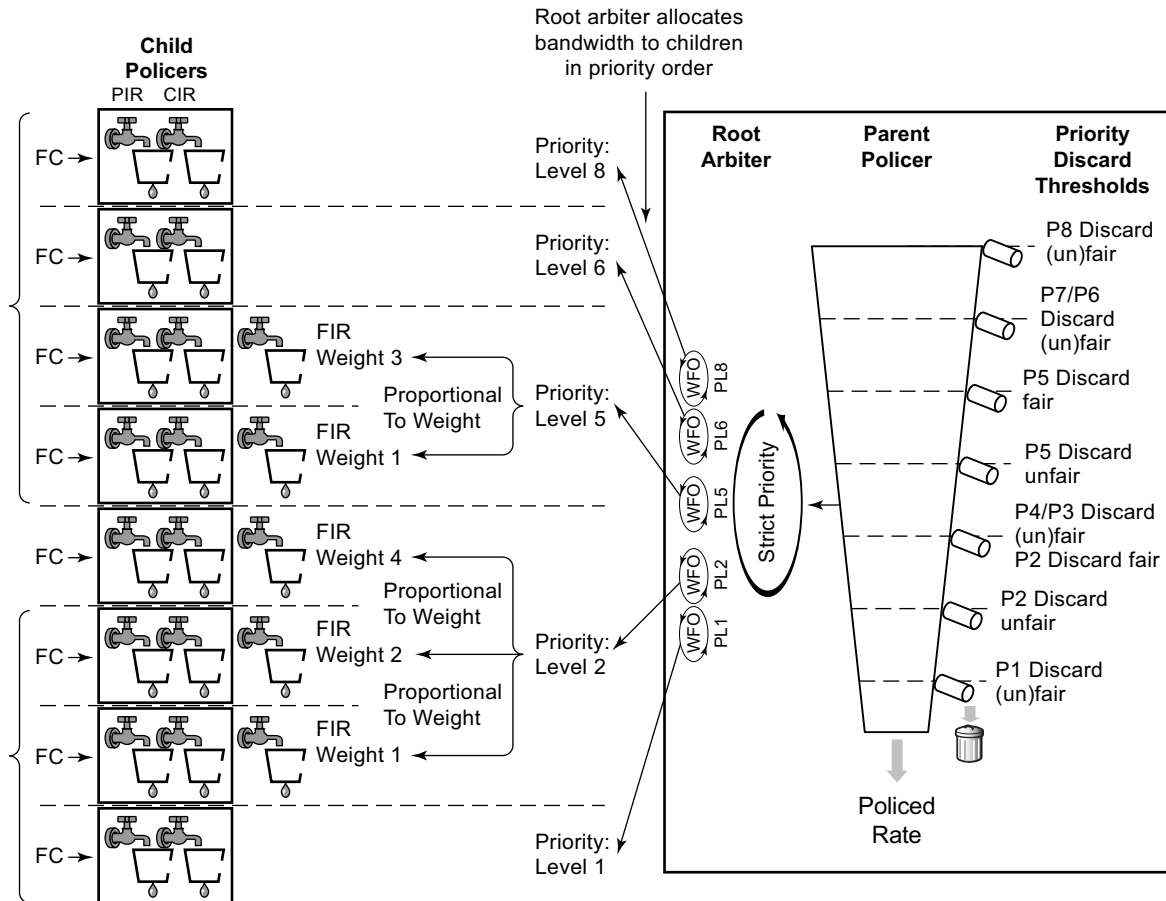
Figure 250: Policer and Arbiter Hierarchy

The final configuration aspect to consider is the parent policer, specifically its multiple thresholds and how they relate to the child policers. See [Figure 251](#).

There are 8 priority levels at the parent policer, each having an associated discard-fair and discard-unfair threshold.

The discard-fair threshold is the upper burst limit for all tokens (consequently, all packets) at the given priority, all traffic at a given priority level is discarded when its tokens arrive with this threshold being exceeded. The discard-fair thresholds enable prioritization at the parent policer by having the burst capacity for each priority threshold be larger (or equal) to those of lower priorities. For example, referring to [Figure 251](#), the priority 6 (P6) discard-fair threshold is larger than the priority 5 (P5) discard-fair threshold with the result that even if the priority 5 and below traffic is overloading the parent policer, the priority 6 traffic has burst capacity available in order to allow some of its packets to conform and get forwarded through the parent policer.

Note that if a packet is discarded at the parent policer, the discard needs to be reflected in the associated child policer, this is achieved by also logically removing the related tokens from the child policer buckets.



OSSG519

Figure 251: Parent Policer and Root Arbiter

Each priority also has a discard-unfair threshold which discards only unfair traffic of that priority, remembering that fair and unfair are determined by the FIR bucket based on the relative weights of the children.

By default, if there are no children configured at a given priority level then both its discard-fair and discard-unfair thresholds are set to zero bytes above the previous priority's discard-fair threshold.

If there is only a single child at a priority level, the discard-fair will be greater than the previous priority's discard-fair value (by an amount equal to the maximum of the min-thresh-separation and the mbs-contribution, see below) but the discard-unfair will be the same as the previous priority's discard-fair threshold (there is no need for a fairness function when there is only a single child at that priority).

If there is more than one child at a priority level, the discard-unfair threshold will be greater than the previous priority's discard-fair threshold by min-thresh-separation (see below) and the discard-fair threshold will be adjusted upwards by an amount equal to mbs-contribution minus min-thresh-separation.

The result can be summarized as follows:

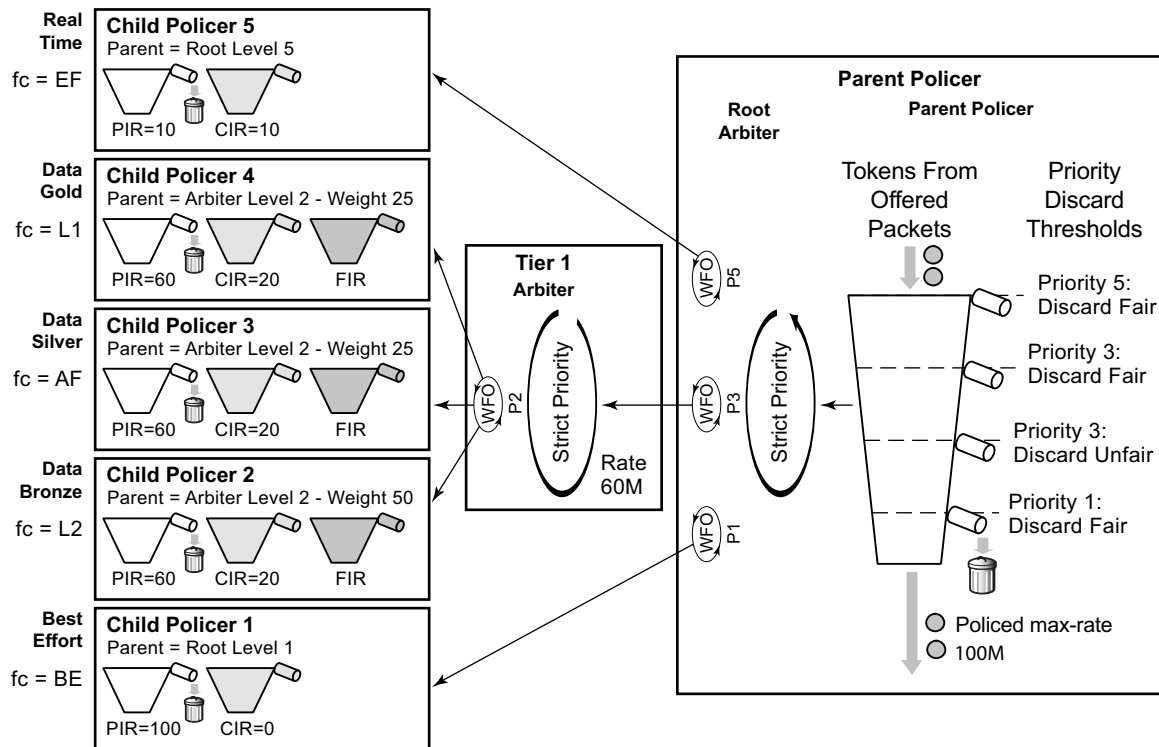
- With no children at a priority level, the discard fair and unfair thresholds match the values of the previous priority.
- If there are at least two children at a priority level, the discard-unfair burst capacity equals min-thresh-separation.
- The burst capacity for a given priority level with at least one child equals the mbs-contribution, unless this is less than min-thresh-separation in which case the min-thresh-separation is used.

The burst tolerance for each threshold is its own burst capacity plus the sum of the burst capacities of all lower thresholds. Referring to [Figure 251](#), the total burst capacity for priority 6 is the sum of the burst capacities for priorities 1 to 6. Note that the burst for a given FC is normally controlled by the burst allowed at the child PIR threshold, not by the parent policer.

As the burst capacity at the parent policer for a given priority level can change when adding or removing children at lower priority levels, a parameter (fixed) is available per priority threshold which causes the discard-fair and discard-unfair thresholds to be non-zero and so greater than the previous priority's thresholds, calculated as above, even when there are no children at that priority level. An exception to this is when the mbs-contribution is set to zero with the fixed parameter configured, in which case both the discard unfair and fair for that priority level are set to zero bytes above the previous level's thresholds (which results in the corresponding traffic being dropped).

A specific configuration and associated show output is included below to highlight the different threshold options described above.

The QoS example shown in [Figure 252](#) is used to describe the configuration of CFHP.



OSSG520

Figure 252: Configuration Example

Five classes of services are accepted, each with a specific CIR and PIR. The data classes, bronze, silver and gold (L2/AF/L1), have a relative weighting of 50/25/25 at priority Level 2 of an intermediate arbiter which is constrained to 60Mbps. At the parent policer, the real time traffic (EF) is defined at level 5, with the data classes at Level 3 and a best effort class (BE) at Level 1. The overall traffic is constrained to 100Mbps at the parent policer. Only unicast traffic is policed in this example.

This example focuses on ingress policing, however, the configuration of policers, arbiters and the parent policer at the egress is almost identical to that at the ingress, the only difference being the particular statistics that can be collected.

There is a difference between ingress and egress policing in terms of how the ingress traffic accesses the switch fabric and the egress traffic access the port after it has been policed. In both cases, unicast access is enabled through a set of policer-output-queues, which are shared-queues at the ingress and queue-groups at the egress (at the egress, user defined queue-groups can be used). It is also possible to use a single service queue to access the egress port. Ingress multipoint traffic accesses the switch fabric using the Ingress Multicast Path Management (IMPM) queues.

Hierarchical Policing

This is shown in [Figure 253](#) on an IOM3-XP (other line cards have the same logic).

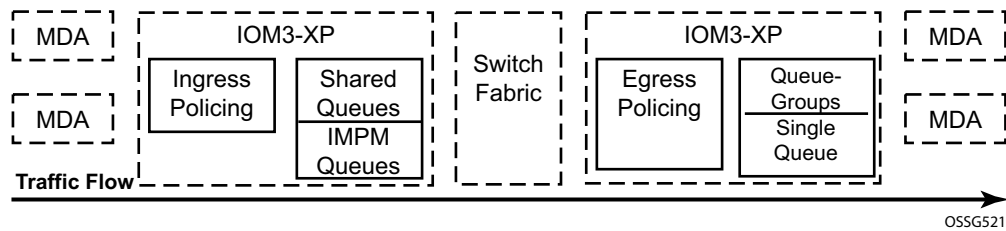


Figure 253: Post Policing Queues

The differences between the ingress and egress policing configuration will be high-lighted in the associated sections.

Configuration

To achieve the QoS shown in [Figure 252](#), configure a SAP-ingress QoS policy to define the child policers and a policer-control-policy to define the intermediate arbiter and the root arbiter/parent policer. As this example is for ingress, the unicast traffic will pass through a set of shared queues called policer-output-queues, which could be modified if required.

Policers

Policers control the CIR and PIR rates for each of the traffic classes and are defined in a SAP-ingress QoS policy. The focus here are parameters related to policing.

The configuration of a child (or standalone) policer is similar to that of a queue.

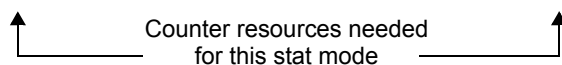
```
config>qos>sap-ingress# policer policer-id [create]
  description "description-string"
  adaptation-rule [pir {max | min | closest}] [cir {max | min | closest}]
  stat-mode {no-stats|minimal|offered-profile-no-cir|
             offered-priority-no-cir|offered-profile-cir|offered-priority-cir|
             offered-total-cir|offered-limited-profile-cir}
  rate {max | kilobits-per-second} [cir {max | kilobits-per-second}]
  percent-rate pir-percent [cir cir-percent]
  mbs size [bytes | kilobytes]
  cbs size [bytes | kilobytes]
  high-prio-only [default | percent-of-mbs]
  parent {root | arbiter-name} [level level] [weight weight-within-level]
  packet-byte-offset {add bytes | subtract bytes}
```

Parameters:

- **description** — This configures a text string, up to 80 characters, which can be used to describe the use of the policy.
- **adaptation-rule** — The hardware supports distinct values for the rates. This parameter tells the system how the rate configured should be mapped onto the possible hardware values. min results in the next higher hardware value being used, max results in the next lower hardware value being used and closest results in the closest available hardware value being used. As can be seen, it is possible to set the adaptation-rule independently for the CIR and PIR.
Default: closest
- **stat-mode** — This defines the traffic statistics collected by the policer, summarized in [Table 13](#).

Table 13: Policer stat-mode

stat-mode	Ingress		Egress	
no-stats	0	Neither policer nor parent arbiter account are required.	0	Neither policer nor parent arbiter accounting are required.
minimal (default)	1	Basic policer accounting (default).	1	Basic policer accounting (default).
offered-profile-no-cir	2	All ingress packets are either in-profile or out-of-profile.	2	Accounting for egress offered profile is required. No visibility for CIR profile state output.
offered-priority-no-cir	2	Ingress packet burst priority accounting is the primary requirement.	N/A	
offered-limited-profile-cir	3	Ingress color-aware profiling is in use but packets are not being classified as in-profile.	N/A	
offered-profile-cir	4	Ingress color-aware profiling is in use and packets are undefined or classified as out-of-profile or in-profile.	4	Egress profile reclassification is performed.
offered-priority-cir	4	Ingress policer is used in color-blind mode and ingress packet priority and CIR state output accounting is needed.	N/A	
offered-total-cir	2	Ingress priority and ingress profile accounting is not needed. CIR profiling is in use.	2	Offered profile visibility is not required (such as, all offered packets have the same profile) and CIR profiling is in use.



Counter resources needed
for this stat mode

- **rate and cir** — The rate defines the PIR and the cir defines the CIR, both are in Kbps. The parameters rate and percent-rate are mutually exclusive and will overwrite each other when configured in the same policy.
Range: PIR=1 to 20,000,000 Kbps or max ; CIR=0 to 20,000,000 Kbps or max
Default: rate(PIR)=max ; cir=0
- **percent-rate and cir** — The percent-rate defines the PIR and the cir defines the CIR with their values being a percentage of the maximum policer rate of 20Gbps. The parameters rate and percent-rate are mutually exclusive and will overwrite each other when

configured in the same policy.

Range: pir-percent = [0.01..100.00]; cir-percent = [0.00..100.00]

Default: pir-percent = 100; cir-percent = 0.00

- mbs and cbs — The mbs defines the MBS for the PIR bucket and the cbs defines the CBS for the CIR bucket, both can be configured in bytes or kilobytes.
Note that the PIR MBS applies to high burst priority packets (these are packets whose classification match criteria is configured with priority high at the ingress and are in-profile packets at the egress).
Range: mbs=0 to 4194304 bytes; cbs=0 to 4194304 bytes
Note: mbs=0 prevents any traffic from being forwarded.
Default: mbs=10ms of traffic or 64KB if PIR=max; cbs=10ms of traffic or 64KB if CIR=max
- high-prio-only — This defines a second burst threshold within the PIR bucket to give a maximum burst size for low burst priority packets (these are packets whose classification match criteria is configured with priority low at the ingress and are out-of-profile packets at the egress). It is configured as a percentage of the MBS.
Default: 10%
- parent — This parameter is used when hierarchical policing is being performed and points to the parent arbiter (which could be the root arbiter or an intermediate arbiter), giving the level to which this policer connects to its parent arbiter and its relative weight compared to other children at the same level. Note that for a child policer to be associated with a parent, its stat-mode cannot be no-stats.
Range: level=1 to 8; weight=1 to 100
Default: level=1; weight=1
- packet-byte-offset — This changes the packet size used for accounting purposes, both in terms of the CIR and PIR rates and what is reported in the statistics. The change can either add or subtract a number of bytes. For example:
 - To have the policer work on Layer 2 frame size including inter-frame gap and preamble, add 20 bytes.
 - To have the policer work on IP packet size instead of the default layer 2 frame size, subtract the encapsulation overhead:
14 bytes L2 + 4bytes VLAN ID + 4 bytes FCS = 22 bytes
Range: add-bytes=0 to 31; sub-bytes=1 to 32
Default: add-bytes=0; sub-bytes=0

A FC must be assigned to the policer in order for the policer to be instantiated (allocating a hardware policer).

By default, any unicast traffic assigned to the FC at the ingress will be processed by the policer, non-unicast traffic would continue to use the multipoint queue. At the egress all traffic assigned to the FC is processed by the policer (as there is no distinction between unicast and non-unicast traffic at the egress).

If required, non-unicast traffic can be policed in IES/VP RN and VPLS services at the ingress (note: all Epipe traffic is treated as unicast). Within an IES/VP RN service, multicast traffic can be assigned to a specific ingress policer on a PIM enabled IP interface. When the service is VPLS, broadcast, unknown unicast and multicast traffic can be individually assigned to ingress policers. In each of these cases, the policers used could be separate from the unicast policer, resulting in the instantiation of additional hardware policers, or a single policer could be used for multiple traffic types (this differs from the queuing implementation where separate queue types are used for unicast and non-unicast traffic).

```
config>qos>sap-ingress>fc#  
    broadcast-policer <policer-id>  
    unknown-policer <policer-id>  
    multicast-policer <policer-id>
```

As mentioned above, the ingress policed unicast traffic passes through a set of shared-queues (policer-output-queues) to access the switch fabric with the multipoint traffic using the IMPM queues.

When policers are required at the egress, a SAP-egress policy is used. The configuration of the policers is almost identical to that used in the SAP-ingress policy, the only difference being the available stat-modes (as shown above).

At the egress, the policed traffic can also be directed to a specific queue-group (instead of the default policer-output-queues) and to a specific queue within that queue-group, as follows:

```
config>qos>sap-egress>fc# policer <policer-id> [group <queue-group-name> [queue  
<queueid>]]
```

It is also possible to direct the egress policed traffic to a single service queue if specific egress queuing is required, as follows:

```
config>qos>sap-egress>fc# policer <policer-id> queue <queue-id>
```

Multiple egress policers in a SAP-egress policy can use the same local queue and other forwarding classes can directly use the same local queue that is being used by policers.

Parent Policer and Arbiters

The parent policer and its associated root arbiter, together with the tier 1 and 2 arbiters, are configured within a policer-control-policy.

```
config>qos# policer-control-policy policy-name [create]
      description description-string
      root
        max-rate {kilobits-per-second | max}
        priority-mbs-thresholds
          min-thresh-separation size [bytes|kilobytes]
          priority level
          mbs-contribution size [bytes|kilobytes] [fixed]
      tier 1
        arbiter arbiter-name [create]
          description description-string
          rate {kilobits-per-second|max}
          parent root [level priority-level] [weight weight-within-level]
      tier 2
        arbiter arbiter-name [create]
          description description-string
          rate {kilobits-per-second | max}
          parent {root|arbiter-name} [level priority-level] [weight weight-within-level]
```

Parameters:

- **description** — This configures a text string, up to 80 characters, which can be used to describe the use of the policy.
- **root** — This section defines the configuration of the parent policer and the root arbiter.
 - **max-rate** — This defines the policed rate of the parent policer, the rate at which the bucket is drained. It is defined in Kbps with an option to use max, in which case the maximum possible rate is used.
Range: 1 to 20,000,000Kbps or max
Default: max
 - **priority-mbs-thresholds**
This section defines the thresholds used for the 8 priorities available in the parent policer.
 - **min-thresh-separation** — This defines the minimum separation between any two active thresholds in the parent policer in units of bytes or kilobytes.
It should be set to a value greater than the maximum packet size used for traffic passing through the policer. This ensures that a single packet arriving in the parent policer will not cause the depth of tokens to cross two burst thresholds, if this did happen it would result in the prioritization failing as a given priority level could be starved of burst capacity by a lower priority traffic.

This parameter is also used as the burst capacity for each priority level's unfair packets.

Range: 0 to 4194304 bytes

Default: 1536 bytes

- mbs-contribution — This is normally used to define the amount of packet burst capacity required at the parent policer for a particular priority level with at least one child, keeping in mind that the total capacity is the sum of this plus that of all lower thresholds. The actual burst capacity used depends also on the setting of min-thresh-separation, as described earlier. This permits the tuning of the burst capacity at the parent for any children at a given priority level. A conservative setting would ensure that the burst at the parent policer for a given priority is the sum of the bursts of all children at that priority. Less conservative settings could use a lower value and assume some level of oversubscription.

The use of the fixed parameter causes both the fair and unfair discard thresholds to be non-zero even when there are no children assigned to this priority level (unless the mbs-contribution is set to zero).

Range: 0 to 4194304

Default: 8192 bytes

The relationship between these two parameters is shown in [Figure 254](#).

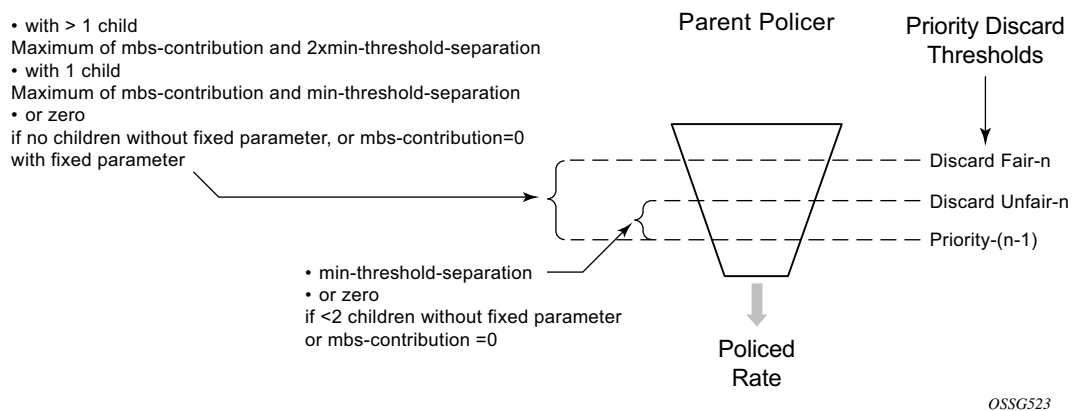


Figure 254: Parent Policer Thresholds

- tier

This section defines the configuration of any intermediate tier 1 or 2 arbiters.

→ arbiter

This specifies the name of the arbiter.

- description — This configures a text string, up to 80 characters, which can be used to describe the use of the policy.
- rate — This defines the rate of the arbiter, it is the maximum rate at which the arbiter will distribute burst capacity to its children. It is defined in Kbps with an option to use max, in which case the maximum possible rate is used.
Range: 1 to 20,000,000Kbps or max
Default: max
- parent — This parameter is used when hierarchical policing is being performed and points to the parent arbiter (which could be the root arbiter or a tier 1 arbiter), giving the level to which this arbiter connects to its parent arbiter and its relative weight compared to other children at the same level.
Range: level=1 to 8; weight=1 to 100
Default: level=1; weight=1

Access to Switch Fabric and Egress Port

After the traffic has been processed by the policers it must pass through a set of queues in order to access the switch fabric at the ingress or the port at the egress.

For the ingress unicast traffic, there is a set of shared-queues (one queue per FC for each possible switch fabric destination) called policer output queues. Note that only their queue characteristics can be configured, the FC to queue mapping is fixed. Also, the PIR/CIR rates only affect the packet scheduling, they do not alter the packet profile state. The details of shared-queues are beyond the scope of this note.

```
config>qos# shared-queue "policer-output-queues"
description description-string
fc <fc-name> [create]
  broadcast-queue <queue-id>
  multicast-queue <queue-id>
  queue <queue-id>
  unknown-queue <queue-id>
queue queue-id [queue-type] [multipoint] [create]
  cbs percent
  mbs percent
  high-prio-only percent
  pool pool-name
  rate percent [cir percent]
```

Multipoint traffic uses the IMPM queues to access the switch fabric. For the egress to the port, either a queue-group or a single service queue is used. There is a default queue-group called policer-output-queues or a user configured queue-group can also be used.

As mentioned above, when a policer is assigned to a specific queue-group (default or user defined) it is optionally possible to configure explicitly the queue to be used. Within the queue-group it is also possible to redirect a FC for policed traffic to a specific queue, using the FC parameter. The preference of the FC to queue mapping is (in order, highest to lowest):

1. Explicitly configured in SAP-egress FC definition
2. Mapped using FC parameter within queue-group definition
3. Default is to use queue 1

```
config>qos>qgrps>egr# queue-group queue-group-name [create]
description description-string
queue queue-id [queue-type] [create]
  adaptation-rule [pir adaptation-rule] [cir adaptation-rule]
  burst-limit size [bytes|kilobytes]
  cbs size-in-kbytes
  high-prio-only percent
  mbs size [bytes|kilobytes]
  parent scheduler-name [weight weight] [level level] [cir-weight cir-weight]
  [cir-level cir-level]
  percent-rate pir-percent [cir cir-percent]
```

```

pool pool-name
port-parent [weight weight] [level level] [cir-weight cir-weight]
             [cir-level cir-level]
rate pir-rate [cir cir-rate]
xp-specific
wred-queue [policy slope-policy-name]
fc fc-name [create]
queue queue-id

```

The default policer-output-queues queue-group consists of two queues; queue 1 being best-effort and queue 2 being expedite. The lowest four FCs (BE, L2, AF, L1) are assigned to queue 1 and the highest four queues (H2, EF, H1, NC) are assigned to queue 2. It may be important to change the queue 2 definition in the queue-group to have CIR=PIR when there are other best-effort queues using a non-zero CIR on the same egress port. This ensures that the policed traffic using queue 2 will be scheduled before any other best-effort within CIR traffic. It also results in the queue CBS being non-zero, allowing the queue 2 traffic access to reserved buffer space.

```

A:PE-1>config>qos# queue-group-templates egress queue-group "policer-output-queues"
A:PE-1>cfg>qos>qgrps>egr>qgrp# info detail
-----
description "Default egress policer output queues."
queue 1 best-effort create
    no parent
    no port-parent
    adaptation-rule pir closest cir closest
    rate max cir 0
    cbs default
    mbs default
    high-prio-only default
    no pool
    xp-specific
        no wred-queue
    exit
    no burst-limit
exit
queue 2 expedite create
    no parent
    no port-parent
    adaptation-rule pir closest cir closest
    rate max cir 0
    cbs default
    mbs default
    high-prio-only default
    no pool
    xp-specific
        no wred-queue
    exit
    no burst-limit
exit
fc af create
    queue 1
exit
fc be create
    queue 1
exit

```

Access to Switch Fabric and Egress Port

```
fc ef create
    queue 2
exit
fc h1 create
    queue 2
exit
fc h2 create
    queue 2
exit
fc l1 create
    queue 1
exit
fc l2 create
    queue 1
exit
fc nc create
    queue 2
exit
```

The remaining details of queue-groups are beyond the scope of this section.

Applying the SAP Ingress and Policer Control Policy

The SAP ingress policy and policer control policy are both applied under the associated SAP. After applying these, it is possible to override the configuration of specific policers and/or the policer control policy. This is shown below. The parameter values are the same as detailed for the policies, as above.

```
config>service><service>#
  sap sap-id [create]
    [ingress|egress]
      qos policy-id
        policer-control-policy policy-name
        policer-override
          policer policer-id [create]
            cbs size [bytes|kilobytes]
            mbs size [bytes|kilobytes]
            packet-byte-offset {add add-bytes | subtract sub-bytes}
            rate {rate | max} [cir {max | rate}
            percent-rate <pir-percent> [cir <cir-percent>]
            stat-mode stat-mode
          policer-control-override [create]
            max-rate {rate | max}
            priority-mbs-thresholds
              min-thresh-separation size [bytes | kilobytes]
              priority level
              mbs-contribution size [bytes | kilobytes]
```

The SAP ingress policy and policer control policy required for the configuration example in [Figure 252](#) is shown below.

```
#-----
echo "QoS Policy Configuration"
#-----
  qos
    policer-control-policy "cfhp-1" create
      root
        max-rate 100000
      exit
      tier 1
        arbiter "a3" create
          parent "root" level 3
          rate 60000
        exit
      exit
    exit
  sap-ingress 10 create
    queue 1 create
    exit
    queue 11 multipoint create
    exit
    policer 1 create
      stat-mode offered-total-cir
      parent "root"
      rate 100000
```

Applying the SAP Ingress and Policer Control Policy

```
        high-prio-only 0
    exit
    policer 2 create
        stat-mode offered-total-cir
        parent "a3" level 2 weight 50
        rate 60000 cir 20000
        high-prio-only 0
    exit
    policer 3 create
        stat-mode offered-total-cir
        parent "a3" level 2 weight 25
        rate 60000 cir 20000
        high-prio-only 0
    exit
    policer 4 create
        stat-mode offered-total-cir
        parent "a3" level 2 weight 25
        rate 60000 cir 20000
        high-prio-only 0
    exit
    policer 5 create
        stat-mode offered-total-cir
        parent "root" level 5
        rate 10000 cir 10000
        high-prio-only 0
    exit
    fc "af" create
        policer 3
    exit
    fc "be" create
        policer 1
    exit
    fc "ef" create
        policer 5
    exit
    fc "l1" create
        policer 4
    exit
    fc "l2" create
        policer 2
    exit
    dot1p 1 fc "be"
    dot1p 2 fc "l2"
    dot1p 3 fc "af"
    dot1p 4 fc "l1"
    dot1p 5 fc "ef"
exit
```

Traffic is classified based on dot1p values, each of which is assigned to an individual FC which in turn is assigned to a policer. The policer rates are configured as required for the example with an appropriate stat-mode. Default values are used for the policer burst thresholds. As all FCs are low burst priority by default, the high-prio-only has been set to zero in order to allow the traffic to use all of the MBS available at the PIR bucket.

Policers 2, 3 and 4 are parented to the arbiter “a3” with the required weights and at a single level (Level 2). In this example it does not matter which level of “a3” is used to parent these policers,

the important aspect is the level at which “a3” is parented to the root. Consequently, these policers use the Level 3 parent policer thresholds (not the level they are parented on a “a3” not Level 2). Arbiter “a3” has a rate of 60Mbps so that its children cannot exceed this rate (except up to the burst tolerances).

Policers 1 and 5 are directly parented to the root arbiter, together with tier 1 arbiter “a3”.

The total capacity for the 5 traffic streams is constrained to 100Mbps by the parent policer, again with the default burst tolerances at the root arbiter.

The SAP-ingress and policer-control-policies are applied to a SAP within an Epipe.

```
#-----
echo "Service Configuration"
#-----
service
  epipe 1 customer 1 create
    sap 1/1/3:1 create
      ingress
        policer-control-policy "cfhp-1"
        qos 10
      exit
    exit
  sap 1/1/4:1 create
  exit
no shutdown
exit
exit
```

Applying the SAP Ingress and Policer Control Policy

The following configuration is used to highlight the relative thresholds in the parent policer when a priority level has 0, 1 or 2 associated children, both with and without using the fixed parameter.

```
-----
echo "QoS Policy Configuration"
#-----
qos
  policer-control-policy "cfhp-2" create
    root
      max-rate 100000
      priority-mbs-thresholds
        min-thresh-separation 256 bytes
        priority 1
          mbs-contribution 1 kilobytes
        exit
        priority 2
          mbs-contribution 1 kilobytes
        exit
        priority 3
          mbs-contribution 1 kilobytes
        exit
        priority 4
          mbs-contribution 1 kilobytes fixed
        exit
        priority 5
          mbs-contribution 1 kilobytes fixed
        exit
        priority 6
          mbs-contribution 1 kilobytes fixed
        exit
      exit
    exit
  exit
  sap-ingress 20 create
    queue 1 create
    exit
    queue 11 multipoint create
    exit
    policer 1 create
      parent "root" level 2
    exit
    policer 2 create
      parent "root" level 3
    exit
    policer 3 create
      parent "root" level 3
    exit
    policer 4 create
      parent "root" level 5
    exit
    policer 5 create
      parent "root" level 6
    exit
    policer 6 create
      parent "root" level 6
    exit
  fc "af" create
```

```

        policer 3
    exit
    fc "be" create
        policer 1
    exit
    fc "ef" create
        policer 6
    exit
    fc "h2" create
        policer 5
    exit
    fc "l1" create
        policer 4
    exit
    fc "l2" create
        policer 2
    exit
exit
#-----
echo "Service Configuration"
#-----
    service
        epipe 2 customer 1 create
            sap 1/1/3:2 create
                ingress
                    policer-control-policy "cfhp-2"
                    qos 20
                exit
            exit
            sap 1/1/4:2 create
            exit
            no shutdown
        exit

```

A policer-control-policy can also be applied under a multi-service site (MSS) so that the hierarchical policing applies to traffic on multiple SAPs, potentially from different services. The MSS can only be assigned to a port, which could be a LAG, but it is not possible to assign an MSS to a card. When MSS are used, policer overrides are not supported.

```

config>service><service>#
    service
        customer customer-id [create]
            multi-service-site customer-site-name [create]
                assignment port port-id
                egress
                    policer-control-policy name
                ingress
                    policer-control-policy name
            service-type
                sap sap-id
                multi-service-site customer-site-name
                ingress
                    qos policy-id
                egress
                    qos policy-id

```

Show Output

After configuring the example as described in the previous section, steady state traffic was sent through the Epipe to overload each of the policers and the show output below was collected. This output focuses on the policer and arbiter details.

The following shows the policers on the SAP and their current state.

```
A:PE-1# show qos policer sap 1/1/3:1
=====
Policer Information (Summary), Slot 1
=====
-----
Name          FC-Maps    MBS      HP-Only  A.PIR    A.CIR
Direction     CBS        Depth    O.PIR    O.CIR    O.FIR
-----
1->1/1/3:1->1
Ingress       be          124 KB   0 KB     100000   0
              0 KB       82      30000    0        30000
1->1/1/3:1->2
Ingress       l2          76 KB    0 KB     60000    20000
              25 KB     77846   30000    20000    30000
1->1/1/3:1->3
Ingress       af          76 KB    0 KB     60000    20000
              25 KB     77824   15000    15000    15000
1->1/1/3:1->4
Ingress       l1          76 KB    0 KB     60000    20000
              25 KB     77868   15000    15000    15000
1->1/1/3:1->5
Ingress       ef          12800 B   0 KB     10000    10000
              12800 B  12834   10000    10000    10000
=====
A:PE-1#
```

The output above shows the configured values for the policers, e.g. PIR and CIR, together with their operational (current) state, such as PIR, CIR and FIR. The depth of each of the PIR buckets is also shown.

The detailed state of each policer can be seen by adding the parameter detail. The following is the output for policer 3.

```
A:PE-1# show qos policer sap 1/1/3:1 ingress detail
...
=====
Policer Info (1->1/1/3:1->3), Slot 1
=====
Policer Name      : 1->1/1/3:1->3
Direction         : Ingress          Fwding Plane      : 1
FC-Map            : af
Depth PIR         : 77842 Bytes      Depth CIR         : 25618 Bytes
Depth FIR         : 77842 Bytes
MBS               : 76 KB             CBS               : 25 KB
Hi Prio Only      : 0 KB             Pkt Byte Offset   : 0
```

Class Fair Hierarchical Policing for SAPs

```

Admin PIR      : 60000 Kbps      Admin CIR      : 20000 Kbps
Oper PIR       : 15000 Kbps      Oper CIR       : 15000 Kbps
Oper FIR       : 15000 Kbps
Stat Mode      : offered-total-cir
PIR Adaption   : closest         CIR Adaption     : closest
Parent Arbiter Name: a3
-----
Arbiter Member Information
-----
Offered Rate   : 45800 Kbps
Level          : 2               Weight         : 25
Parent PIR     : 15000 Kbps      Parent FIR     : 15000 Kbps
Consumed       : 15000 Kbps
-----
=====...
A:PE-1#

```

Notice that the above output shows the depth of the PIR, CIR and FIR buckets together with their operational rates. This can be used to explain the operation of the policers in this example and is discussed later in this section.

The stat-mode of offered-total-cir configured on policer 3 results in these statistics being collected.

```

A:PE-1# show service id 1 sap 1/1/3:1 stats
=====
...
-----
Sap per Policer stats
-----
                Packets                Octets

Ingress Policer 1 (Stats mode: offered-total-cir)
Off. All        : 2690893              172217152
Dro. InProf     : 0                    0
Dro. OutProf    : 967465              61917760
For. InProf     : 0                    0
For. OutProf    : 1723428             110299392

Ingress Policer 2 (Stats mode: offered-total-cir)
Off. All        : 2690988              172223232
Dro. InProf     : 0                    0
Dro. OutProf    : 909492              58207488
For. InProf     : 1178507             75424448
For. OutProf    : 602989              38591296
...

```

Show Output

The following output is included for reference and shows the statistics which are collected for each of the ingress and egress stat-modes.

```
PE-1# show service id 2 sap 1/1/1:2 stats
...
-----
Sap per Policer stats
-----
                Packets                Octets

Ingress Policer 1 (Stats mode: no-stats)

Ingress Policer 2 (Stats mode: minimal)
Off. All          : 0                  0
For. All          : 0                  0
Dro. All          : 0                  0

Ingress Policer 3 (Stats mode: offered-profile-no-cir)
Off. InProf       : 0                  0
Off. OutProf      : 0                  0
For. InProf       : 0                  0
For. OutProf      : 0                  0
Dro. InProf       : 0                  0
Dro. OutProf      : 0                  0

Ingress Policer 4 (Stats mode: offered-priority-no-cir)
Off. HiPrio       : 0                  0
Off. LowPrio      : 0                  0
For. HiPrio       : 0                  0
For. LoPrio       : 0                  0
Dro. HiPrio       : 0                  0
Dro. LowPrio      : 0                  0

Ingress Policer 5 (Stats mode: offered-profile-cir)
Off. InProf       : 0                  0
Off. OutProf      : 0                  0
Off. Uncolor      : 0                  0
For. InProf       : 0                  0
For. OutProf      : 0                  0
Dro. InProf       : 0                  0
Dro. OutProf      : 0                  0

Ingress Policer 6 (Stats mode: offered-priority-cir)
Off. HiPrio       : 0                  0
Off. LowPrio      : 0                  0
For. InProf       : 0                  0
For. OutProf      : 0                  0
Dro. InProf       : 0                  0
Dro. OutProf      : 0                  0

Ingress Policer 7 (Stats mode: offered-total-cir)
Off. All          : 0                  0
For. InProf       : 0                  0
For. OutProf      : 0                  0
Dro. InProf       : 0                  0
Dro. OutProf      : 0                  0

Ingress Policer 8 (Stats mode: offered-limited-profile-cir)
```


Off. OutProf	: 0	0
Off. Uncolor	: 0	0
For. InProf	: 0	0
For. OutProf	: 0	0
Dro. InProf	: 0	0
Dro. OutProf	: 0	0

Egress Policer 1 (Stats mode: no-stats)

Egress Policer 2 (Stats mode: minimal)

Off. All	: 0	0
For. All	: 0	0
Dro. All	: 0	0

Egress Policer 3 (Stats mode: offered-profile-no-cir)

Off. InProf	: 0	0
Off. OutProf	: 0	0
For. InProf	: 0	0
For. OutProf	: 0	0
Dro. InProf	: 0	0
Dro. OutProf	: 0	0

Egress Policer 4 (Stats mode: offered-profile-cir)

Off. InProf	: 0	0
Off. OutProf	: 0	0
Off. Uncolor	: 0	0
For. InProf	: 0	0
For. OutProf	: 0	0
Dro. InProf	: 0	0
Dro. OutProf	: 0	0

Egress Policer 5 (Stats mode: offered-total-cir)

Off. All	: 0	0
For. InProf	: 0	0
For. OutProf	: 0	0
Dro. InProf	: 0	0
Dro. OutProf	: 0	0

=====

Show Output

It is possible to show the policer-control-policy details and the SAPs with which it is associated, as shown here.

```
A:PE-1# show qos policer-control-policy cfhp-1
=====
QoS Policer Control Policy
=====
Policy-Name       : cfhp-1
Description       : (Not Specified)
Min Threshold Sep : Def

-----
Priority MBS Thresholds
-----
Priority          MBS Contribution
-----
1                 none
2                 none
3                 none
4                 none
5                 none
6                 none
7                 none
8                 none

-----
Tier/Arbiter          Lvl/Wt      Rate      Parent
-----
    root              N/A         100000    None
1 a3                  3/1         60000     root

=====
A:PE-1# show qos policer-control-policy "cfhp-1" association
=====
QoS Policer Control Policy
=====
Policy-Name       : cfhp-1
Description       : (Not Specified)

-----
Associations
-----
Service-Id         : 1 (Epipe)      Customer-Id       : 1
- SAP : 1/1/3:1 (Ing)

=====
A:PE-1
```

The following command shows the policer hierarchy, including the child policers and their relationship to the intermediate arbiter (a3) and the root arbiter. It can be used to monitor the status of the child policers in the hierarchy. The output shows the assigned, offered and consumed capacity for each policer.

```
A:PE-1# show qos policer-hierarchy sap 1/1/3:1
=====
Policer Hierarchy - Sap 1/1/3:1
=====
Ingress Policer Control Policy : cfhp-1
Egress Policer Control Policy :
-----
root (Ing)
|
| slot(1)
|
|--(A) : a3 (Sap 1/1/3:1)
|
|   |--(P) : Policer 1->1/1/3:1->4
|   |
|   |   [Level 2 Weight 25]
|   |   Assigned PIR:15000      Offered:45800
|   |   Consumed:15000
|   |
|   |   Assigned FIR:15000
|   |
|   |--(P) : Policer 1->1/1/3:1->3
|   |
|   |   [Level 2 Weight 25]
|   |   Assigned PIR:15000      Offered:45800
|   |   Consumed:15000
|   |
|   |   Assigned FIR:15000
|   |
|   |--(P) : Policer 1->1/1/3:1->2
|   |
|   |   [Level 2 Weight 50]
|   |   Assigned PIR:30000      Offered:45800
|   |   Consumed:30000
|   |
|   |   Assigned FIR:30000
|   |
|   |--(P) : Policer 1->1/1/3:1->5
|   |
|   |   [Level 5 Weight 1]
|   |   Assigned PIR:10000      Offered:10000
|   |   Consumed:10000
|   |
|   |   Assigned FIR:10000
|   |
|   |--(P) : Policer 1->1/1/3:1->1
|   |
|   |   [Level 1 Weight 1]
|   |   Assigned PIR:30000      Offered:45800
|   |   Consumed:30000
|   |
|   |   Assigned FIR:30000
```

Show Output

```
root (Egr)
|
No Active Members Found on slot 1
=====
A:PE-1#
```

The complete information about the policer hierarchy can be seen by adding the detail parameter, as shown below, with alternative parameters to select more specific information.

- root-detail — Rates, depth and thresholds for the root arbiter.
- thresholds — CBS, MBS and high-prio-only thresholds with associated rates of child policers.
- priority-info — Discard-fair and discard-unfair thresholds, with number of associated children, for each of the root priority levels.
- depth — Parent policer and child PIR buckets depth, with PIR and FIR rate information.
- arbiter — Specific information of a given arbiter.
- port — For use with LAGs in different line cards or using adapt-qos link.

The output adds a good representation of the root arbiter thresholds, indicating the priority levels, discard-unfair and discard-fair thresholds, and how many child policers are associated with each level. It also includes the current depth of the child policer PIR buckets and the parent policer bucket.

```
A:PE-1# show qos policer-hierarchy sap 1/1/3:1 detail
=====
Policer Hierarchy - Sap 1/1/3:1
=====
Ingress Policer Control Policy : cfhp-1
Egress Policer Control Policy :
-----
Legend :
(*) real-time dynamic value
(w) Wire rates
-----
root (Ing)
|
| slot(1)
|   MaxPIR:100000
|   ConsumedByChildren:100000
|   OperPIR:100000      OperFIR:100000
|
|   DepthPIR:8111 bytes
| Priority 8
|   Oper Thresh Unfair:17408      Oper Thresh Fair:25600
|   Association count:0
| Priority 7
|   Oper Thresh Unfair:17408      Oper Thresh Fair:25600
|   Association count:0
| Priority 6
|   Oper Thresh Unfair:17408      Oper Thresh Fair:25600
```

```

Association count:0
Priority 5
  Oper Thresh Unfair:17408      Oper Thresh Fair:25600
  Association count:1
Priority 4
  Oper Thresh Unfair:9728      Oper Thresh Fair:17408
  Association count:0
Priority 3
  Oper Thresh Unfair:9728      Oper Thresh Fair:17408
  Association count:3
Priority 2
  Oper Thresh Unfair:0         Oper Thresh Fair:8192
  Association count:0
Priority 1
  Oper Thresh Unfair:0         Oper Thresh Fair:8192
  Association count:1

--(A) : a3 (Sap 1/1/3:1)
      MaxPIR:60000
      ConsumedByChildren:60000
      OperPIR:60000      OperFIR:60000

      [Level 3 Weight 1]
      Assigned PIR:60000      Offered:60000
      Consumed:60000

      Assigned FIR:60000

--(P) : Policer 1->1/1/3:1->4
      MaxPIR:60000      MaxCIR:20000
      CBS:25600      MBS:77824
      HiPrio:0
      Depth:77876

      OperPIR:15000      OperCIR:15000
      OperFIR:15000
      PacketByteOffset:0
      StatMode: offered-total-cir

      [Level 2 Weight 25]
      Assigned PIR:15000      Offered:45800
      Consumed:15000

      Assigned FIR:15000

--(P) : Policer 1->1/1/3:1->3
      MaxPIR:60000      MaxCIR:20000
      CBS:25600      MBS:77824
      HiPrio:0
      Depth:77834

      OperPIR:15000      OperCIR:15000
      OperFIR:15000
      PacketByteOffset:0
      StatMode: offered-total-cir

      [Level 2 Weight 25]
      Assigned PIR:15000      Offered:45800
      Consumed:15000

```

Show Output

```
| | |
| | | Assigned FIR:15000
| | |
| | | --(P) : Policer 1->1/1/3:1->2
| | | MaxPIR:60000 MaxCIR:20000
| | | CBS:25600 MBS:77824
| | | HiPrio:0
| | | Depth:77848
| | |
| | | OperPIR:30000 OperCIR:20000
| | | OperFIR:30000
| | | PacketByteOffset:0
| | | StatMode: offered-total-cir
| | |
| | | [Level 2 Weight 50]
| | | Assigned PIR:30000 Offered:45800
| | | Consumed:30000
| | |
| | | Assigned FIR:30000
| | |
| | | --(P) : Policer 1->1/1/3:1->5
| | | MaxPIR:10000 MaxCIR:10000
| | | CBS:12800 MBS:12800
| | | HiPrio:0
| | | Depth:12854
| | |
| | | OperPIR:10000 OperCIR:10000
| | | OperFIR:10000
| | | PacketByteOffset:0
| | | StatMode: offered-total-cir
| | |
| | | [Level 5 Weight 1]
| | | Assigned PIR:10000 Offered:10000
| | | Consumed:10000
| | |
| | | Assigned FIR:10000
| | |
| | | --(P) : Policer 1->1/1/3:1->1
| | | MaxPIR:100000 MaxCIR:0
| | | CBS:0 MBS:126976
| | | HiPrio:0
| | | Depth:135
| | |
| | | OperPIR:30000 OperCIR:0
| | | OperFIR:30000
| | | PacketByteOffset:0
| | | StatMode: offered-total-cir
| | |
| | | [Level 1 Weight 1]
| | | Assigned PIR:30000 Offered:45800
| | | Consumed:30000
| | |
| | | Assigned FIR:30000
```

root (Egr)

No Active Members Found on slot 1

```
=====
A:PE-1#
```

The output above gives the depth of the parent policer, which can be used with the output below to explain the operation of the policing in this example.

```
A:PE-1# show qos policer sap 1/1/3:1 detail | match expression "Slot | Bytes | Kbps"
Policer Info (1->1/1/3:1->1), Slot 1
Depth PIR      : 153 Bytes      Depth CIR      : 0 Bytes
Depth FIR      : 153 Bytes
Admin PIR      : 100000 Kbps    Admin CIR      : 0 Kbps
Oper PIR       : 30000 Kbps     Oper CIR       : 0 Kbps
Oper FIR       : 30000 Kbps
Offered Rate   : 45800 Kbps
Parent PIR     : 30000 Kbps     Parent FIR     : 30000 Kbps
Consumed       : 30000 Kbps
Policer Info (1->1/1/3:1->2), Slot 1
Depth PIR      : 77828 Bytes    Depth CIR      : 25624 Bytes
Depth FIR      : 77828 Bytes
Admin PIR      : 60000 Kbps     Admin CIR      : 20000 Kbps
Oper PIR       : 30000 Kbps     Oper CIR       : 20000 Kbps
Oper FIR       : 30000 Kbps
Offered Rate   : 45800 Kbps
Parent PIR     : 30000 Kbps     Parent FIR     : 30000 Kbps
Consumed       : 30000 Kbps
Policer Info (1->1/1/3:1->3), Slot 1
Depth PIR      : 77858 Bytes    Depth CIR      : 25634 Bytes
Depth FIR      : 77858 Bytes
Admin PIR      : 60000 Kbps     Admin CIR      : 20000 Kbps
Oper PIR       : 15000 Kbps     Oper CIR       : 15000 Kbps
Oper FIR       : 15000 Kbps
Offered Rate   : 45800 Kbps
Parent PIR     : 15000 Kbps     Parent FIR     : 15000 Kbps
Consumed       : 15000 Kbps
Policer Info (1->1/1/3:1->4), Slot 1
Depth PIR      : 77838 Bytes    Depth CIR      : 25614 Bytes
Depth FIR      : 77838 Bytes
Admin PIR      : 60000 Kbps     Admin CIR      : 20000 Kbps
Oper PIR       : 15000 Kbps     Oper CIR       : 15000 Kbps
Oper FIR       : 15000 Kbps
Offered Rate   : 45800 Kbps
Parent PIR     : 15000 Kbps     Parent FIR     : 15000 Kbps
Consumed       : 15000 Kbps
Policer Info (1->1/1/3:1->5), Slot 1
Depth PIR      : 12814 Bytes    Depth CIR      : 12814 Bytes
Depth FIR      : 12814 Bytes
Admin PIR      : 10000 Kbps     Admin CIR      : 10000 Kbps
Oper PIR       : 10000 Kbps     Oper CIR       : 10000 Kbps
Oper FIR       : 10000 Kbps
Offered Rate   : 10000 Kbps
Parent PIR     : 10000 Kbps     Parent FIR     : 10000 Kbps
Consumed       : 10000 Kbps
A:PE-1#
```

From the output above, it can be seen that the offered rate for policers 1-4 is 45800Kbps, in fact it is the same for policer 5 but this is capped at the admin PIR rate, 10000Kbps.

Show Output

The depth of the parent policer is only 8111 bytes, so this is not causing any discarding of priority 2-5 traffic at the parent policer as their discard thresholds are all above this value. Therefore the drops in policers 2-5 are all occurring in the child policers.

Policer 5 is consuming all of its operational capacity (PIR, CIR and FIR), and it can be seen that the level of the PIR bucket is 12814 bytes, which is slightly above its MBS of 12800 bytes. The level of the PIR bucket will oscillate around the MBS value as tokens are added to exceed the threshold (causing discards) then the draining reduces the level to just below the threshold (allowing forwarding).

Policers 2-4 are functioning in the same way as policer 5, as can be seen from their PIR bucket levels (levels are 77828 bytes with MBS of 77824), resulting in the PIR buckets constraining the rates of the traffic through these policers. This is happening because the arbiter “a3” is distributing its 60000Kbps in the configured ratio to these policers, which changes the operational PIR to 30000Kbps for policer 2 and 15000Kbps for policers 3 and 4, all being below the offered traffic rate. A similar effect can be seen with the CIR rates and bucket depths, as the operational CIR rate of policer 2 has reached its administrative value with those of policer 3 and 4 being constrained by the operational PIR. The CIR bucket depths are just above the CBS, again this will oscillate causing traffic to both in-profile and out-of-profile. As this is steady state traffic, the operational FIR rates for these policers have settled to match their operational PIR rates.

Policer 1 is also discarding traffic at the PIR bucket but it is also discarding traffic at the parent policer. This can be seen by the fact that policer 1 PIR depth is nowhere near its MBS whereas the parent policer level is just below the priority 1 discard-fair threshold. The level of the parent policer bucket will oscillate around this threshold causing policer 1 traffic to be discarded, which in turn is reflected back into the level of tokens in the policer 1 PIR bucket.

As this example is based on ingress unicast policing, the traffic exits the policers and then accesses the switch fabric using a set of shared-queue (policer-output-queues). The parameters for these queues can be seen using the following **show** command.

```
A:PE-1# show qos shared-queue "policer-output-queues" detail
=====
QoS Shared Queue Policy
=====
-----
Shared Queue Policy (policer-output-queues)
-----
Policy          : policer-output-queues
Description     : Default Policer Output Shared Queue Policy
-----
```

Queue	CIR	PIR	CBS	MBS	HiPrio	Multipoint	Pool-Name
1	0	100	1	50	10	FALSE	
2	25	100	3	50	10	FALSE	
3	25	100	10	50	10	FALSE	
4	25	100	3	25	10	FALSE	
5	100	100	10	50	10	FALSE	
6	100	100	10	50	10	FALSE	
7	10	100	3	25	10	FALSE	

Class Fair Hierarchical Policing for SAPs

```

8      10      100      3      25      10      FALSE
9      0       100      1      50      10      TRUE
10     25      100      3      50      10      TRUE
11     25      100      10     50      10      TRUE
12     25      100      3      25      10      TRUE
13     100     100      10     50      10      TRUE
14     100     100      10     50      10      TRUE
15     10      100      3      25      10      TRUE
16     10      100      3      25      10      TRUE

```

```

-----
FC      UCastQ      MCastQ      BCastQ      UnknownQ
-----
be      1           9           9           9
l2      2           10          10          10
af      3           11          11          11
l1      4           12          12          12
h2      5           13          13          13
ef      6           14          14          14
h1      7           15          15          15
nc      8           16          16          16

```

Associations

```

-----
Service : 1          SAP : 1/1/3:1
=====
A:PE-1#

```

For egress policing, policed traffic can access the exit port by a queue-group, the default being called policer-output-queues. The following shows the parameters for these queues.

```

A:PE-1# show qos queue-group "policer-output-queues" detail

```

QoS Queue-Group Ingress

QoS Queue-Group Egress

QoS Queue Group

```

-----
Group-Name      : policer-output-queues
Description     : Default egress policer output queues.

```

```

-----
Q  CIR Admin PIR Admin CBS      HiPrio PIR Lvl/Wt      Parent      BurstLimit(B)
   CIR Rule  PIR Rule  MBS      CIR Lvl/Wt      Wred-Queue   Slope
   Named-Buffer Pool
-----
1  0          max      def      def      1/1          None          default
   closest    closest  def      0/1          disabled      default
   (not-assigned)
2  0          max      def      def      1/1          None          default
   closest    closest  def      0/1          disabled      default

```

Show Output

```
(not-assigned)

=====
Queue Group Ports (access)
=====
Port                Sched Pol          Acctg Pol Stats    Description
-----
1/1/3                0                  No              No
1/1/4                0                  No              No
-----

=====
Queue Group Ports (network)
=====
Port                Sched Pol          Acctg Pol Stats    Description
-----
No Matching Entries

=====
Queue Group Sap FC Maps
=====
Sap Policy          FC Name            Queue Id
-----
No Matching Entries
=====
A:PE-1#
```

The following output shows the relative thresholds in the parent policer when a priority level has 0, 1 or 2 associated children, both with and without using the fixed parameter.

```
A:PE-1# show qos policer-hierarchy sap 1/1/3:2 ingress priority-info
=====
Policer Hierarchy - Sap 1/1/3:2
=====
Ingress Policer Control Policy : cfhp-2
-----
root (Ing)
|
| slot(1)
|   Priority 8
|     Oper Thresh Unfair:4352      Oper Thresh Fair:5120
|     Association count:0
|   Priority 7
|     Oper Thresh Unfair:4352      Oper Thresh Fair:5120
|     Association count:0
|   Priority 6
|     Oper Thresh Unfair:4352      Oper Thresh Fair:5120
|     Association count:2 fixed
|   Priority 5
|     Oper Thresh Unfair:3328      Oper Thresh Fair:4096
|     Association count:1 fixed
|   Priority 4
|     Oper Thresh Unfair:2304      Oper Thresh Fair:3072
|     Association count:0 fixed
|   Priority 3
```

```

|   Oper Thresh Unfair:1280      Oper Thresh Fair:2048
|   Association count:2
| Priority 2
|   Oper Thresh Unfair:0        Oper Thresh Fair:1024
|   Association count:1
| Priority 1
|   Oper Thresh Unfair:0        Oper Thresh Fair:0
|   Association count:0

```

```

=====
A:PE-1#

```

Where

- Priority Level 1 has no children so both its fair and unfair thresholds are 0.
- Priority Level 2 has one child so its unfair threshold is 0 and its fair threshold is at the configured mbs-contribution [1024 bytes] (given that this is larger than the min-thresh-separation).
- Priority Level 3 has two children so its unfair threshold is equal to the min-thresh-separation plus the fair threshold of priority 2 [256+1024=1280 bytes]. Its fair threshold is effectively the mbs-contribution plus the fair threshold of priority 2 [1024+1024=2048 bytes] (given that the mbs-contribution is larger than 2x min-thresh-separation).
- Priorities 4, 5 and 6 have the fixed parameter configured. Even though priority 4 has no children, priority 5 has only one child and priority 6 has two children, all three priorities have the same incremental values for their unfair and fair discard threshold. This result in
 - Priority 4's unfair threshold being equal to the min-thresh-separation plus the fair threshold of priority 3 [256+2048=2304 bytes]. Its fair threshold is effectively the mbs-contribution plus the fair threshold of priority 3 [1024+2048=3072 bytes] (given that the mbs-contribution is larger than 2x min-thresh-separation).
 - Priority 5's unfair threshold being equal to the min-thresh-separation plus the fair threshold of priority 4 [256+3072=3328 bytes]. Its fair threshold is effectively the mbs-contribution plus the fair threshold of priority 4 [1024+3072=4096 bytes] (given that the mbs-contribution is larger than 2x min-thresh-separation).
 - Priority 6's unfair threshold being equal to the min-thresh-separation plus the fair threshold of priority 5 [256+4096=4352 bytes]. Its fair threshold is effectively the mbs-contribution plus the fair threshold of priority 5 [1024+4096=5120 bytes] (given that the mbs-contribution is larger than 2x min-thresh-separation).

Note that the above parameter values were chosen to exactly match available hardware values to simplify the output.

Conclusion

This note has described the configuration of Class Fair Hierarchical Policing for SAPs. This hardware policing provides low latency ingress and egress prioritized traffic control with the ability to provide fairness between child policers at the same parent policer priority level.

Pseudowire QoS

In This Chapter

This section describes pseudowire QoS configurations.

Topics in this section include:

- [Applicability on page 1778](#)
- [Overview on page 1779](#)
- [Configuration on page 1784](#)
- [Conclusion on page 1801](#)

Applicability

This example is applicable to the 7950 XRS-16c/20, 7750 SR-7/12, 7750 SR-c4/12 and 7450 ESS-6/6v/7/12 platforms when all network IP interfaces are on IOM3-XP/IMM (FP2 and above) hardware.

The configuration was tested on release 11.0R4. There are no other specific pre-requisites for this configuration.

Overview

A pseudowire (PW) provides a virtual connection across an IP or MPLS network between services configured on provider edge (PE) devices. From SR OS R10.0R1, it is possible to provide specific QoS to either a single pseudowire or a multiple pseudowires. This is supported for the following:

- SDP
 - MPLS
 - GRE
- Epipe
 - Including vc-switching and dynamic MS-PW
 - PBB-epipe
 - BGP-VPWS (from 11.0R1)
- VPLS
 - Mesh and spoke SDP
 - LDP signaled pseudowires
 - BGP-AD signaled pseudowires
 - I-VPLS, B-VPLS
 - BGP-VPLS
- Spoke termination on IES/VPRN (both Epipe and Ipipe)
- Apipe (from R10.0R4)
- Cpipe (from R10.0R4)
- Fpipe (from R10.0R4)
- Ipipe (from R10.0R4)

It is supported at ingress on both Ethernet and POS/TDM ports on an IOM3-XP/IMM and only on Ethernet ports at egress.

Bandwidth control is achieved using queue-groups which are implemented per FP (flexpath) at the ingress and per port at the egress (these being relative to the data path through the system), as shown in [Figure 255](#) and [Figure 256](#), respectively.

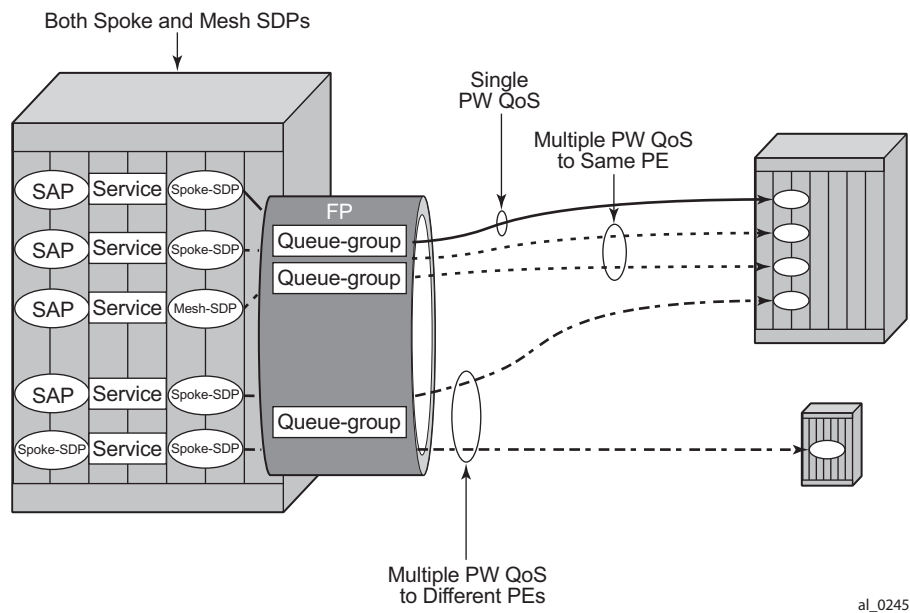


Figure 255: Ingress PW QoS

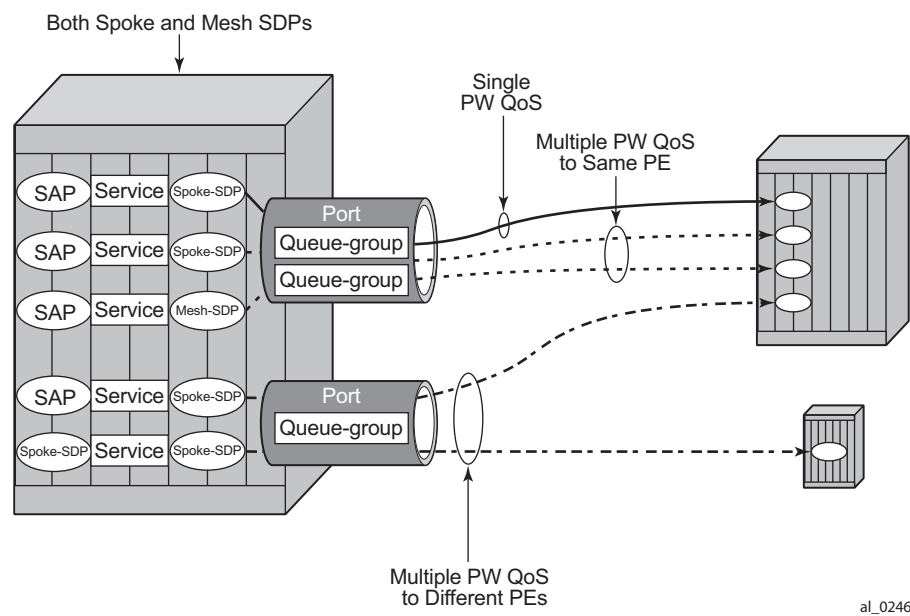


Figure 256: Egress PW QoS

Bandwidth control is applied independently for ingress and egress, and can be set up for a single pseudowire or for multiple pseudowires where the remote services are located on a single PE or on multiple PEs.

It is possible to benefit from Hierarchical QoS which can be configured under the queue-groups, but this is beyond the scope of this example.

The ingress and egress classification and egress marking is configured by applying a network QoS policy to each pseudowire.

Ingress QoS

Ingress QoS is achieved using a queue group which is applied to an ingress FP on a card. Queue groups applied to an FP can only contain policers, not queues. The network QoS policy applied to the pseudowire redirects forwarding classes (FCs) to the individual queue group (unicast or multipoint) policers. The actual queue group to be used is defined separately to the network QoS policy, thereby allowing the network QoS policies to be independent from the queue groups used and therefore both are reusable.

Ingress bandwidth control does not take into account the outer Ethernet header, the MPLS labels/control word or GRE headers, or the FCS of the incoming frame. The configuration allows an offset to be added or subtracted from the received frame size in order to change the actual length used for the bandwidth control.

For example: if the same ingress rate is configured on a pseudowire (without a control word) and a dot1q SAP, what packet-byte-offset needs to be used on the pseudowire in order to achieve the same throughput as on the SAP?

- SAP — The following shows the bytes in the frame that are used by default on a policer for the rate at a SAP ingress.

6B	6B	4B	2B	xxxxB	4B
Source MAC	Dest. MAC	802.1Q	Ether Type	Payload	CRC/FCS

al_0247

- VPLS Pseudowire — For a tagged (**vc-type vlan**) pseudowire, it would be necessary to add 4 bytes using the packet-byte-offset applied to the ingress policer in order to achieve the same throughput as on the SAP. This compensates for the omission of the FCS that is included on the SAP and so needs to be added.

Egress QoS

6B	6B	2B	4B	4B	6B	6B	4B	2B	xxxxB	4B
Source MAC	Dest. MAC	Ether Type	Tun MPLS Label	VC MPLS Label	Source MAC	Dest. MAC	802.1Q	Ether Type	Payload	CRC/FCS

al_0248

- **VPRN Pseudowire** — For an Ipipe (**vc-type** ipipe) pseudowire, it would be necessary to add 22 bytes using the packet-byte-offset to the ingress policer to achieve the same throughput as on the SAP. This compensates for the omission of the source and destination MAC addresses (12 bytes), Ether type (2 bytes), VLAN tag (4 bytes) and the FCS (4 bytes) that are included on the SAP and so needs to be added.

6B	6B	2B	4B	4B	xxxxB	4B
Source MAC	Dest. MAC	Ether Type	Tun MPLS Label	VC MPLS Label	Payload	CRC/FCS

al_0249

The ingress classification is configured in the ingress section of the network QoS policy and is based on the outer encapsulation header only, the outer Ethernet header (dot1p/DE), MPLS labels (EXP) or GRE headers (DSCP). At an egress LER, the ler-use-dscp is applicable only to IES and VPRN pseudowires.

Egress QoS

Egress QoS is achieved using a queue group which is applied to an egress port. Queue groups applied to a port can contain both policers and queues. The network QoS policy applied to the pseudowire redirects forwarding classes (FCs) to the individual queue group policers/queues. The actual queue group to be used is defined separately to the network QoS policy, thereby allowing the network QoS policies to be independent from the queue groups used and therefore both are reusable.

Egress bandwidth control does takes into account the outer Ethernet header, MPLS labels/control word or GRE headers, and the FCS of the outgoing frame. The configuration allows an offset to be added or subtracted from the sent frame size in order to affect the actual length used for the bandwidth control.

For example, if the same egress rate is configured on a pseudowire (without a control word) and a dot1q SAP, what packet-byte-offset needs to be used on the pseudowire in order to achieve the same throughput as on the SAP?

- **SAP** — The following shows the bytes in the frame that are used by default on a policer/queue at a SAP egress.

6B	6B	4B	2B	xxxxB	4B
Source MAC	Dest. MAC	802.1Q	Ether Type	Payload	CRC/FCS

al_0250

- VPLS Pseudowire — For a tagged (**vc-type vlan**) pseudowire, it would be necessary to subtract 22 bytes using the packet-byte-offset applied to the egress policer/queue applied to achieve the same throughput as on the SAP. This compensates for the MPLS header (source and destination MAC addresses (12 bytes), Ether type (2 bytes), two labels (8 bytes)) that is not included on the SAP and needs to be subtracted.

6B	6B	2B	4B	4B	6B	6B	4B	2B	xxxxB	4B
Source MAC	Dest. MAC	Ether Type	Tun MPLS Label	VC MPLS Label	Source MAC	Dest. MAC	802.1Q	Ether Type	Payload	CRC/FCS

al_0251

- VPRN Pseudowire — For an Ipipe (**vc-type ipipe**) pseudowire, it would be necessary to subtract 4 bytes using the packet-byte-offset applied to the egress policer/queue applied to achieve the same throughput as on the SAP. This compensates for the MPLS header (source and destination MAC addresses (12 bytes), Ether type (2 bytes), two labels (8 bytes)) that is not included on the SAP so is subtracted, and the source and destination MAC addresses (12 bytes), dot1q header (4 bytes) and Ether type (2 bytes) of the SAP frame which needs to be added. This results in subtracting 4 bytes.

6B	6B	2B	4B	4B	xxxxB	4B
Source MAC	Dest. MAC	Ether Type	Tun MPLS Label	VC MPLS Label	Payload	CRC/FCS

al_0252

The egress classification and marking is configured in the egress section of the network QoS policy. DSCP/prec egress reclassification is supported from release R10.0R4 for IES and VPRN spoke SDPs. The egress marking affects the outer encapsulation header, the outer Ethernet header (dot1p/DE), MPLS labels (EXP) or GRE headers (DSCP).

Configuration

The configuration of pseudowire QoS is described using an Epipe pseudowire. The topology is shown in [Figure 257](#).

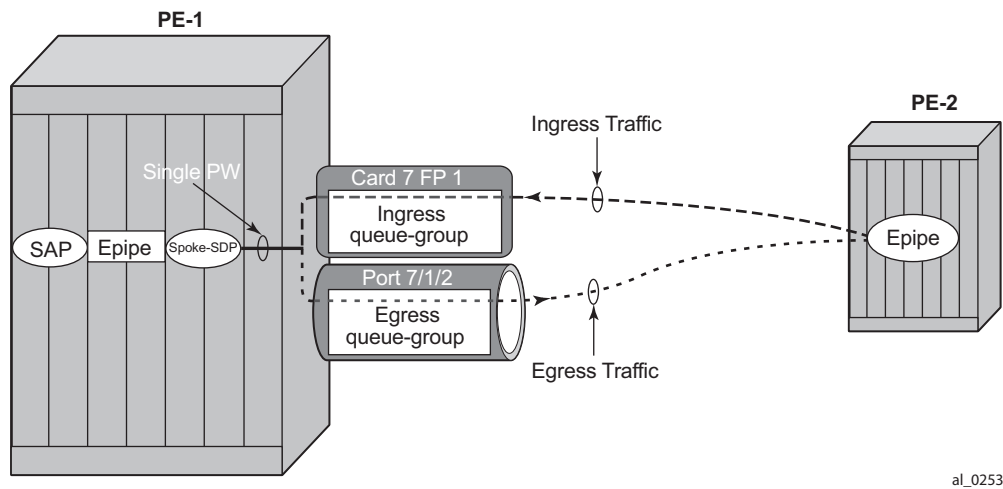


Figure 257: Example Epipe Pseudowire Topology

The following pre-requisite configuration is assumed to be in place:

- Hardware provisioning
- IP address and routing
- MPLS protocols
- SDP
- Epipe service, including the SAP
- SAP QoS policies

Traffic is sent across a virtual leased line between PE-1 and PE-2 using Epipes with a pseudowire configured as a spoke SDP on each PE. The QoS is applied to the pseudowire at the ingress and egress of PE-1.

The following configuration is required for applying pseudowire QoS:

- Create the ingress and egress queue groups.
These contain the ingress policer and egress policer/queue definitions.

- Create an instance of the ingress queue group on the ingress FP and instance of the egress queue group on the port that will be used for the pseudowire traffic.
- Create a network QoS policy to redirect the traffic to the ingress and egress queue groups, and to perform the ingress classification and egress marking.
- Apply the network QoS policy, together with the reference to the ingress and egress queue group instances, to the spoke SDP representing the pseudowire.

The traffic consists of two bidirectional flows, one in FC BE and one in FC EF. At the ingress of the pseudowire, each FC is assigned to its own policer, whereas at the egress of the pseudowire, FC BE is assigned to a queue and FC EF is assigned to a policer.

Although this example makes use of both ingress and egress queue groups, the focus is pseudowire QoS, so the full details of queue group configuration are not covered.

Create the Ingress and Egress Queue Groups

Queue groups are created using templates, which are separate for ingress and egress. The following shows the queue group templates configured.

```
configure qos
  queue-group-templates
    ingress
      queue-group "ingress-queue-group" create
      policer 1 create
        rate 6000
        packet-byte-offset add 4
      exit
      policer 2 create
        rate 4000
        packet-byte-offset add 4
      exit
    exit
  exit
  egress
    queue-group "egress-queue-group" create
    queue 1 best-effort create
      rate 6000
      xp-specific
        packet-byte-offset subtract 22
      exit
    exit
    policer 1 create
      rate 4000
      packet-byte-offset subtract 22
    exit
  exit
exit
exit
```

Create the Ingress FP and Egress Port Queue Group Instances

The ingress queue group has two policers associated with it; policer 1 will be used for the FC BE traffic and policer 2 will be used for the FC EF traffic. The configuration of policers in an ingress queue group is the same as that in a sap-ingress QoS policy, with the exception that the percent-rate is not supported within the queue group.

In order to achieve the same ingress throughput as that when applying the same rates to policers on a dot1q tagged SAP, the packet-byte-offset adds 4 bytes to the packet length for both policers.

The egress queue group has one queue (queue 1) that will be used for the FC BE traffic and one policer (policer 1) that will be used for the FC EF traffic. The configuration of policers in an egress queue group is the same as that in a sap-egress QoS policy, with the exception that the percent-rate is not supported within the queue group. The configuration of queues in an egress queue group is the same as in a sap-egress QoS policy, with the exception that the avg-frame-overhead is not supported within the queue group.

In order to achieve the same egress throughput as that when applying the same rates to policers/queues on a dot1q tagged SAP, the packet-byte-offset subtracts 22 bytes from the packet length for both the policer and queue.

Rates have been configured such that the ingress and egress capacity of the BE traffic is 6Mb/s and 4Mb/s for the EF traffic.

Create the Ingress FP and Egress Port Queue Group Instances

The queue group templates are then applied as individual instances to the ingress FP and egress port; using instances allows the reuse of the same template.

Below is the ingress FP configuration. From a QoS perspective, it is also possible to configure a policer-control-policy under the ingress queue group in order to perform hierarchical policing. From R11.0R4, the configuration supports overrides for both the policer-control-policy parameters and some of the queue group policer parameters.

```
configure
  card 7
    card-type imm5-10gb-xfp
    mda 1
      no shutdown
    exit
    fp 1
      ingress
        network
          queue-group "ingress-queue-group" instance 1 create
        exit
      exit
    exit
  exit
  no shutdown
exit
```

Below is the egress port configuration. From a QoS perspective, it is also possible to configure under the egress queue group a policer-control-policy in order to perform hierarchical policing, a scheduler-policy in order to perform hierarchical shaping and overrides for some of the queue group queue parameters.

```
configure
  port 7/1/2
    ethernet
      network
        egress
          queue-group "egress-queue-group" instance 1 create
          exit
        exit
      exit
    exit
  no shutdown
exit
```

If there are redundant network interfaces over which the pseudowire traffic can enter or exit the system, it is necessary to configure any ingress FP and egress port queue groups consistently across all possible interfaces to be used by the pseudowire to ensure the QoS is always applied. If a queue group configuration was omitted, the pseudowire would not be subject to the QoS defined in that queue group.

If a LAG is used, the system only allows the egress port queue group to be added or removed from the LAG primary port, thereby keeping the LAG configuration consistent. However, this is not possible at the ingress as the queue-group is applied at the FP, so it is necessary to ensure that the ingress queue group is applied consistently on all FP's corresponding to the configured LAG.

Create the Network QoS Policy

A network QoS policy is created to redirect ingress and egress traffic to the respective queue groups, and perform ingress classification (in this example).

The redirection to the queue group policer/queue is performed per FC.

At ingress, traffic can be redirected to policers (being the same or different policers) based on the traffic type. Unicast traffic is redirected to a policer specified by the policer command and will use the ingress shared policer-output-queues to access the switch fabric. All multipoint traffic is redirected to the policer specified by the multicast-policer command (for example with a pseudowire configured in a VPLS service, all broadcast, unknown and multicast traffic will use this policer). The multipoint traffic accesses the switch fabric using the Ingress Multicast Path Management queues. It is possible to individually redirect one traffic type (unicast or multipoint) within an FC to a queue group policer while allowing the other traffic type to use default network queues.

At egress, traffic can be redirected to a queue or to a policer. The policed traffic will exit the egress port using one of the default network queues (with the queue chosen by FC assignment) or optionally can use a queue in the egress queue group if configured in the port-redirect-group command following the policer parameter.

Any FC not redirected to a queue-group, will continue to use the regular default network ingress and egress queues.

The syntax for the FC redirection is as follows.

```
config# qos
  network <network-policy-id> [create]
    ingress
      fc <fc-name>
        fp-redirect-group multicast-policer <policer-id>
        fp-redirect-group policer <policer-id>
    egress
      fc <fc-name>
        port-redirect-group {queue <queue-id>|
                             policer <policer-id> [queue <queue-id>]}
```

The required commands are shown below.

```
configure qos
  network 10 create
    ingress
      lsp-exp 5 fc ef profile in
      fc be
        fp-redirect-group policer 1
      exit
      fc ef
        fp-redirect-group policer 2
```



```

        exit
    exit
    egress
        fc be
            port-redirect-group queue 1
        exit
        fc ef
            port-redirect-group policer 1
        exit
    exit
exit

```

At ingress, the FC BE and FC EF traffic are redirected to the two policers in the queue-group applied to the FP. At egress, the two FCs are redirected to the queue and policer in the queue group applied to the egress port.

The ingress classification required here is for the traffic which is received with exp=5 to be in FC EF.

Apply Network QoS Policy with Queue Group Instances to the Spoke SDP

To apply the QoS to the pseudowire, the following commands can be used, dependent on the service type.

```

config# service {apipe|cpipe|epipe|fpipe|ipipe} <service-id>
    spoke-sdp <sdp-id:vc-id>
        ingress
            qos <network-policy-id> fp-redirect-group <queue-group-name>
                                                    instance <instance-id>
        egress
            qos <network-policy-id> port-redirect-group <queue-group-name> instance <instance-id>

config# service {ies|vprn} <service-id>
    interface <ip-int-name>
        spoke-sdp <sdp-id:vc-id>
            ingress
                qos <network-policy-id> fp-redirect-group <queue-group-name> instance <instance-id>
            egress
                qos <network-policy-id> port-redirect-group <queue-group-name>
                                                    instance <instance-id>

config# service vpls <service-id>
    {spoke-sdp|mesh-sdp} <sdp-id:vc-id>
        ingress
            qos <network-policy-id> fp-redirect-group <queue-group-name> instance <instance-id>
        egress
            qos <network-policy-id> port-redirect-group <queue-group-name>
                                                    instance <instance-id>

```

Apply Network QoS Policy with Queue Group Instances to the Spoke SDP

For services using BGP auto-discovery to signal the pseudowire, the QoS configuration is included in the pseudowire template.

```
config# service pw-template <policy-id>
    ingress
        qos <network-policy-id> fp-redirect-group <queue-group-name> instance <instance-id>
    egress
        qos <network-policy-id> port-redirect-group <queue-group-name>
                                           instance <instance-id>
```

To propagate changes in a pw-template to existing BGP-AD pseudowires, it is necessary to use the following command:

```
tools perform service eval-pw-template policy-id
```

Note that the allow-service-impact parameter is not required for changing the ingress or egress QoS definition as these do not affect the operational state of the pseudowire.

QoS applied directly to a pseudowire, using the above commands, takes precedence over any QoS applied to the network interface (using a network QoS policy with or without queue group redirection).

Note that each time a pseudowire uses a network egress port queue group an FP resource is allocated. This only requires that the pseudowire egress QoS is configured with a port-redirect-group, and will occur even if there are no FCs redirected using a port-redirect-group within the configured network QoS policy. The resources used can be seen using the **tools dump system-resources** command and is listed under Egr Network Queue Group Mappings which is part of the total for the “Dynamic Service Entries”.

As an Epipe is used in this example, QoS is configured directly under a spoke SDP.

```
configure service
    epipe 1 customer 1 create
        spoke-sdp 1:1 vc-type vlan create
            ingress
                qos 10 fp-redirect-group "ingress-queue-group" instance 1
            exit
            egress
                qos 10 port-redirect-group "egress-queue-group" instance 1
            exit
            no shutdown
        exit
        no shutdown
    exit
```

The created network QoS policy is applied at both ingress and egress, with the ingress referencing the ingress queue group instance applied to the FP and the egress referencing the egress queue group instance applied to the port.

Show Output

The configured ingress queue group can be shown, including the details of the configured policers and where it is applied.

```
*A:PE-1# show qos queue-group "ingress-queue-group" ingress detail
=====
QoS Queue-Group Ingress
=====
-----
QoS Queue Group
-----
Group-Name      : ingress-queue-group
Description     : (Not Specified)
-----
...
=====
Queue Group FP Maps
=====
Card Num      Fp Num      Instance      Type
-----
7             1             1             Network
-----
Entries found: 1
-----
=====
Queue Group Policer
=====
Policer Id     : 1
Description    : (Not Specified)
PIR Adptn      : closest          CIR Adptn      : closest
Parent         : none             Level         : 1
Weight         : 1               Adv. Cfg Plcy: none
Admin PIR      : 6000            Admin CIR     : 0
CBS            : def             MBS           : def
Hi Prio Only   : def             Pkt Offset    : 4
Profile Capped : Disabled
StatMode       : minimal
=====
Policer Id     : 2
Description    : (Not Specified)
PIR Adptn      : closest          CIR Adptn      : closest
Parent         : none             Level         : 1
Weight         : 1               Adv. Cfg Plcy: none
Admin PIR      : 4000            Admin CIR     : 0
CBS            : def             MBS           : def
Hi Prio Only   : def             Pkt Offset    : 4
Profile Capped : Disabled
StatMode       : minimal
```

Similar information can be shown for the egress queue group, including the details of the configured queue and policer and again where it is applied.

```
*A:PE-1# show qos queue-group "egress-queue-group" egress detail
=====
QoS Queue-Group Egress
=====
-----
QoS Queue Group
-----
Group-Name      : egress-queue-group
Description     : (Not Specified)

-----
Q  CIR Admin PIR Admin CBS          HiPrio PIR Lvl/Wt      Parent      BurstLimit(B)
   CIR Rule  PIR Rule  MBS          CIR Lvl/Wt      Wred-Queue   Slope
   Named-Buffer Pool              Adv Config Policy Name
-----
1  0          6000      def          def        1/1          None        default
   closest    closest   def          0/1          disabled    default
   (not-assigned)          (not-assigned)

...
=====
Queue Group Ports (network)
=====
Port  Sched Pol  Policer-Ctrl-Pol  Acctg Pol  Stats Description  QGrp-Instance
-----
7/1/2                                No          1

...
=====
Queue Group Policer
=====
Policer Id      : 1
Description     : (Not Specified)
PIR Adptn       : closest          CIR Adptn      : closest
Parent          : none             Level          : 1
Weight          : 1                Adv. Cfg Plcy : none
Admin PIR       : 4000             Admin CIR      : 0
CBS             : def              MBS            : def
Hi Prio Only    : def              Pkt Offset     : -22
Profile Capped  : Disabled
StatMode       : minimal

...
```

The following command shows where the ingress queue group has been applied.

```
*A:PE-1# show qos queue-group ingress association
=====
QoS Queue-Group Ingress
=====
...
-----
QoS Queue Group
-----
Group-Name      : ingress-queue-group
Description     : (Not Specified)
...
=====
Queue Group FP Maps
=====
Card Num      Fp Num      Instance      Type
-----
7              1              1              Network
-----
Entries found: 1
-----
...
=====
```

A similar command shows where the egress queue group has been applied.

```
*A:PE-1# show qos queue-group egress association
=====
QoS Queue-Group Egress
=====
-----
QoS Queue Group
-----
Group-Name      : egress-queue-group
Description     : (Not Specified)
...
=====
Queue Group Ports (network)
=====
Port   Sched Pol   Policer-Ctrl-Pol  Acctg Pol  Stats  Description  QGrp-Instance
-----
7/1/2                No              1
-----
...
=====
```

Apply Network QoS Policy with Queue Group Instances to the Spoke SDP

The ingress queue group applied to the FP on card 7 can be shown.

```
*A:PE-1# show card 7 fp 1 ingress queue-group "ingress-queue-group" instance 1
mode network
=====
Card:7  Net.QGrp: ingress-queue-group  Instance: 1
=====
Group Name      : ingress-queue-group
Description     : (Not Specified)
Pol Ctl Pol     : None                      Acct Pol      : None
Collect Stats   : disabled
```

In order to show the details of the policers in the ingress FP queue group, the following command can be used.

```
*A:PE-1# show qos policer card 7 fp 1 queue-group "ingress-queue-group" instance 1 network
detail
=====
Policer Info (Net-FPQG-1-ingress-queue-group:1->1), Slot 7
=====
Policer Name      : Net-FPQG-1-ingress-queue-group:1->1
Direction         : Ingress                      Fwding Plane      : 1
Depth PIR         : 0 Bytes                      Depth CIR         : 0 Bytes
Depth FIR         : 0 Bytes
MBS               : 7680 B                      CBS               : 0 KB
Hi Prio Only      : 768 B                      Pkt Byte Offset   : 4
Admin PIR         : 6000 Kbps                   Admin CIR         : 0 Kbps
Oper PIR          : 6000 Kbps                   Oper CIR         : 0 Kbps
Oper FIR          : 6000 Kbps
Stat Mode         : minimal
PIR Adaption      : closest                      CIR Adaption      : closest
Adv.Cfg Plcy     : None                        Profile Capped    : disabled
Parent Arbiter Name: (Not Specified)
-----
Arbiter Member Information
-----
Offered Rate      : 0 Kbps
Level            : 0                          Weight            : 0
Parent PIR       : 0 Kbps                    Parent FIR       : 0 Kbps
Consumed         : 0 Kbps
-----
=====
Policer Info (Net-FPQG-1-ingress-queue-group:1->2), Slot 7
=====
Policer Name      : Net-FPQG-1-ingress-queue-group:1->2
Direction         : Ingress                      Fwding Plane      : 1
Depth PIR         : 0 Bytes                      Depth CIR         : 0 Bytes
Depth FIR         : 0 Bytes
MBS               : 5 KB                      CBS               : 0 KB
Hi Prio Only      : 512 B                      Pkt Byte Offset   : 4
Admin PIR         : 4000 Kbps                   Admin CIR         : 0 Kbps
Oper PIR          : 4000 Kbps                   Oper CIR         : 0 Kbps
Oper FIR          : 4000 Kbps
Stat Mode         : minimal
PIR Adaption      : closest                      CIR Adaption      : closest
Adv.Cfg Plcy     : None                        Profile Capped    : disabled
```

```

Parent Arbiter Name: (Not Specified)
-----
Arbiter Member Information
-----
Offered Rate      : 0 Kbps
Level             : 0
Parent PIR        : 0 Kbps
Consumed          : 0 Kbps
Weight            : 0
Parent FIR        : 0 Kbps
-----
Network Interface Association
-----
No Association Found.
-----
SDP Association
-----
Policer Info (1->1:1->10), Slot 7
-----
SDP Association Count : 1
-----

```

The details of the queue and policer in the egress queue group applied to port 7/1/2 can also be shown.

```

*A:PE-1# show port 7/1/2 queue-group egress "egress-queue-group" network instance 1
=====
Ethernet port 7/1/2 Network Egress queue-group
=====
Group Name      : egress-queue-group Instance-Id   : 1
Description     : (Not Specified)
Sched Policy    : None                        Acct Pol   : None
Collect Stats   : disabled                    Agg. Limit  : -1

Queues
-----
Queue-Group     : egress-queue-group Instance-Id   : 1      Queue-Id    : 1
Description     : (Not Specified)
Admin PIR       : 6000*                      Admin CIR   : 0*
PIR Rule        : closest*                   CIR Rule    : closest*
CBS             : def*                       MBS         : def*
Hi Prio         : def*

Policers
-----
Queue-Group     : egress-queue-group Instance-Id   : 1      Policer-Id   : 1
Description     : (Not Specified)
Admin PIR       : 4000*                      Admin CIR   : 0*
PIR Rule        : closest*                   CIR Rule    : closest*
CBS             : def*                       MBS         : def*
Hi Prio         : def*

* means the value is inherited

```

Apply Network QoS Policy with Queue Group Instances to the Spoke SDP

The network QoS policy can be shown with the details of the configured FC redirection and ingress classification used on the pseudowire.

```
*A:PE-1# show qos network 10 detail
=====
QoS Network Policy
=====
-----
Network Policy (10)
-----
Policy-id      : 10                      Remark      : False
Forward Class  : be                      Profile     : Out
LER Use DSCP   : False
Description    : (Not Specified)
...
-----
LSP EXP Bit Map          Forwarding Class          Profile
-----
5                          ef                      In
...
-----
Egress Forwarding Class Mapping
-----
FC Value      : 0                      FC Name      : be
- DSCP Mapping
Out-of-Profile : be                      In-Profile   : be
...
DE Mark       : None
Redirect Grp Q : 1                      Redirect Grp Plcr: None
...
FC Value      : 5                      FC Name      : ef
...
DE Mark       : None
Redirect Grp Q : None                   Redirect Grp Plcr: 1
...
-----
Ingress Forwarding Class Mapping
-----
FC Value      : 0                      FC Name      : be
Redirect UniCast Plcr : 1              Redirect MultiCast Plcr : None
...
FC Value      : 5                      FC Name      : ef
Redirect UniCast Plcr : 2              Redirect MultiCast Plcr : None
...
```


The details of the configuration of the pseudowire QoS can be seen when showing the details of the SDP within the Epipe service.

```
*A:PE-1# show service id 1 sdp 1:1 detail
=====
Service Destination Point (Sdp Id : 1:1) Details
=====
-----
Sdp Id 1:1   - (192.0.2.2)
-----
Description      : (Not Specified)
SDP Id           : 1:1                               Type           : Spoke
Spoke Descr      : (Not Specified)
VC Type          : VLAN                               VC Tag          : 0
Admin Path MTU   : 0                                 Oper Path MTU   : 9190
Delivery         : MPLS
Far End          : 192.0.2.2
Tunnel Far End   : 192.0.2.2                         LSP Types       : LDP
Hash Label       : Disabled                           Hash Lbl Sig Cap : Disabled
Oper Hash Label  : Disabled

Admin State      : Up                               Oper State      : Up
...
Ingress Qos Policy : 10                             Egress Qos Policy : 10
Ingress FP QGrp   : ingress-queue-group              Egress Port QGrp  : egress-queue*
Ing FP QGrp Inst  : 1                               Egr Port QGrp Inst: 1
```

The usage of the “Egr Network Queue Group Mappings” out of the total number of “Dynamic Service Entries” can be seen with the following command. Only one egress QoS pseudowire redirection has been configured.

```
*A:PE-1# tools dump system-resources
Resource Manager info at 005 m 07/31/13 13:11:03.355:

Hardware Resource Usage for Slot #7, CardType imm5-10gb-xfp, Cmplx #0:
-----+-----+-----+-----
...
Dynamic Service Entries |      65535 |      1 |      65534
Subscriber Hosts       |            |      0 |
Encap Group Members    |            |      0 |
Egr Network Queue Group Mappings |      1 |
```

It is possible to show the statistics on the ingress FP queue group used by the pseudowire.

```
*A:PE-1# show card 7 fp 1 ingress queue-group "ingress-queue-group" instance 1 mode net-
work statistics

=====
Card:7 Net.QGrp: ingress-queue-group Instance: 1
=====
Group Name      : ingress-queue-group
Description     : (Not Specified)
Pol Ctl Pol     : None                               Acct Pol       : None
```

Apply Network QoS Policy with Queue Group Instances to the Spoke SDP

```
Collect Stats : disabled
```

Statistics

```
-----
Packets                               Octets
Ing. Policer:  1  Grp: ingress-queue-group (Stats mode: minimal)
Off. All       :           184275           23587200
Dro. All       :           36801            4710528
For. All       :           147474           18876672
```

```
Ing. Policer:  2  Grp: ingress-queue-group (Stats mode: minimal)
Off. All       :           184274           23587072
Dro. All       :           85955            11002240
For. All       :           98319            12584832
```

Similar statistics can be shown for the egress port queue group used by the pseudowire.

```
*A:PE-1# show port 7/1/2 queue-group egress "egress-queue-group" network statistics
instance 1
```

Ethernet port 7/1/2 Network Egress queue-group

```
-----
Packets                               Octets
Egress Queue:  1  Group: egress-queue-group  Instance-Id:  1
In Profile forwarded : 0                      0
In Profile dropped   : 0                      0
Out Profile forwarded : 150989                 19326592
Out Profile dropped   : 37123                  4751744
```

```
Egress Policer:  1  Group: egress-queue-group  Instance-Id:  1
Stats mode: minimal
Off. All       : 188421           24117888
Dro. All       : 87894            11250432
For. All       : 100527           12867456
```

Monitor commands are available to see the statistics (and rates on egress port queue group). As an example, the following shows the utilization on the queue and policer in the egress queue-group.

```
*A:PE-1# monitor port 7/1/2 queue-group "egress-queue-group" instance 1 egress network
egress-queue 1 repeat 1 rate
```

Monitor Port Queue-Group Egress Network Queue Statistics

```
-----
At time t = 0 sec (Base Statistics)
```

```
-----
Packets                               Octets
In Profile forwarded : 0                      0
In Profile dropped   : 0                      0
Out Profile forwarded : 299113                 38286464
Out Profile dropped   : 74155                  9491840
```

```
-----
At time t = 11 sec (Mode: Rate)
```

```

                                Packets/sec          Octets/sec          % Port
                                                Util.

In Profile forwarded   : 0                      0                      0.00
In Profile dropped    : 0                      0                      0.00
Out Profile forwarded : 5863                    750436                 0.06
Out Profile dropped   : 1466                    187609                 0.01
=====

*A:PE-1# monitor port 7/1/2 queue-group "egress-queue-group" instance 1 egress network
policer 1 repeat 1 rate
=====
Monitor Port Queue-Group Egress Network Policer Statistics
=====
-----
At time t = 0 sec (Base Statistics)
-----
                                Packets          Octets

Off. All      : 454750                    58208000
Dro. All      : 212181                    27159168
For. All      : 242569                    31048832
-----

At time t = 11 sec (Mode: Rate)
-----
                                Packets/sec          Octets/sec          % Port
                                                Util.

Off. All      : 7326                      937716                0.07
Dro. All      : 3419                      437609                0.03
For. All      : 3907                      500108                0.04
=====

*A:PE-1#

```

As mentioned, the egress policer (FC EF) traffic exits the egress port by default using the related network queue on the port. This can be seen below.

```

*A:PE-1# show port 7/1/2 detail
=====
Ethernet Interface
=====
Description      : 10-Gig Ethernet
Interface        : 7/1/2
Link-level       : Ethernet
Admin State      : up
Oper State       : up
Oper Speed       : 10 Gbps
Config Speed     : N/A
Oper Duplex      : full
Config Duplex    : N/A
...
=====
Queue Statistics
=====
-----
...
Egress Queue 6
In Profile forwarded : 0          Octets
In Profile dropped   : 0          0

```

Apply Network QoS Policy with Queue Group Instances to the Spoke SDP

```

Out Profile forwarded :    102381                15357150
Out Profile dropped   :      0                    0

```

The throughput achieved using the above configuration can be verified in the traffic generator output. Port 202/1 is connected to PE-1 and port 204/1 is connected to PE-2.

Port ▲	Tx Test Packets	Rx Test Packets	Tx Test Octets	Rx Test Octets	Tx Test Throughput (Mb/s)	Rx Test Throughput (Mb/s)	Rx Packet Loss	Average Latency (us)
All Ports	29296	19531	3749888	2499968	29.999	20.000	n/a	15512.18
202/1	14648	9765	1874944	1249920	15.000	9.999	n/a	39.28
202/1->204/1, BE traffic	7324	5860	937472	750080	7.500	6.001	1464	51609.56
202/1->204/1, EF traffic	7324	3906	937472	499968	7.500	4.000	3418	39.13
204/1	14648	9766	1874944	1250048	15.000	10.000	n/a	30983.50
204/1->202/1, BE traffic	7324	5859	937472	749952	7.500	6.000	1465	39.28
204/1->202/1, EF traffic'	7324	3906	937472	499968	7.500	4.000	3418	39.27

Conclusion

This example has shown the configuration and monitoring of pseudowire QoS, providing a powerful QoS solution for pseudowire applications. QoS can be applied independently to the ingress and/or egress of a single pseudowire or multiple pseudowires.

QoS Architecture and Basic Operation

In This Chapter

This section provides information about QoS architecture and basic operation.

Topics in this section include:

- [Applicability on page 1804](#)
- [Overview on page 1805](#)
- [Configuration on page 1806](#)
- [Conclusion on page 1845](#)

Applicability

The information in this section is applicable to all of the Alcatel-Lucent 7x50 platforms and is focused on the FP2 chipset, which is used in the IOM3-XP/IMM and in the 7750 SR-c12/4. The configuration was tested on release 9.0R3.

Overview

The 7x50 platforms provide an extensive Quality of Service (QoS) capability for service provider solutions. QoS is a system behavior to treat different traffic with different amounts of resources, including buffer memory and queue serving time.

By allocating system resources with certain degrees of guarantee, the bandwidth can be used more efficiently and more controllably. Lack of buffer memory leads to packet drop, while a smaller amount of queue serving time normally means longer delay for the packet and may cause buffer memory to be completely consumed and eventually also lead to packet drop.

In a single box system, such as the 7x50 platforms, different types of traffic contend for the same resources at several major points, such as the ingress to the switch fabric and the egress out of a physical port. In a multi-node network, QoS is achieved on hop by hop basis. Thus, QoS needs to be configured individually but with the consistency across the whole network.

This note is focused on the configuration of the basic QoS, namely the use of queues to shape traffic at the ingress and egress of the system. More sophisticated aspects will be referenced where appropriate but their details are beyond the scope of this note. Other topics not included are Hierarchical QoS scheduling, egress port-scheduler, queue-groups, named buffer pools, WRED-per-queue, LAGs, high scale MDA, QoS for ATM/FR and Enhanced Subscriber Management.

QoS Components

QoS consists of four main components:

- Classification
- Buffering (enqueueing)
- Scheduling (dequeueing)
- Remarking

These are also the fundamental building blocks of the QoS implementation in the 7x50. Ingress packets, classified by various rules, belong to one of eight Forwarding Classes (FC). A FC can be viewed a set of packets which are treated in a similar manner within the system (have the same priority level and scheduling behavior). Each FC has a queue associated with it and each queue has a set of parameters controlling how buffer memory is allocated for packets; if a packet is enqueued (placed on the queue) a scheduler will control the way the packet gets dequeued (removed from the queue) relative to other queues. When a packet exits an egress port, a remarking action can be taken to guarantee the next downstream device will properly handle the different types of traffic.

Configuration

Policies

QoS policies are used to control how traffic is handled at distinct points in the service delivery model within the device. There are different types of QoS policies catering to the different QoS needs at each point. QoS policies only take effect when applied to a relevant entity (Service Access Point (SAP) or network port/interface) so by default can be seen as templates with each application instantiating a new set of related resources.

The following QoS policies are discussed:

- Ingress/egress QoS Policies — For classification, queue attributes and remarking.
 - Slope policies — Define the RED slope definitions.
 - Scheduler policies — Determine how queues are scheduled (only the default scheduling is included here).
-

Access Network and Hybrid Ports

The system has two different types of interfaces: access and network.

- A network interface will classify packets received from the network core at ingress and remark packets sent to the core at egress. Aggregated differentiated service QoS is performed on network ports, aggregating traffic from multiple services into a set of queues per FC.
- An access interface connects to one or more customer devices; it receives customer packets, classifies them into different FCs at ingress and remarks packets according to FCs at egress. Since an access interface needs application awareness, it has many more rules to classify the ingress packets. Access and network also differ in how buffer memory is handled, as will be made clear when discussing the buffer management. Here the QoS is performed per SAP.

Access interfaces (SAPs) are configured on access ports and network interfaces are configured on network ports. A third type of port is available, the hybrid port, which supports both access and network interfaces on the same port.

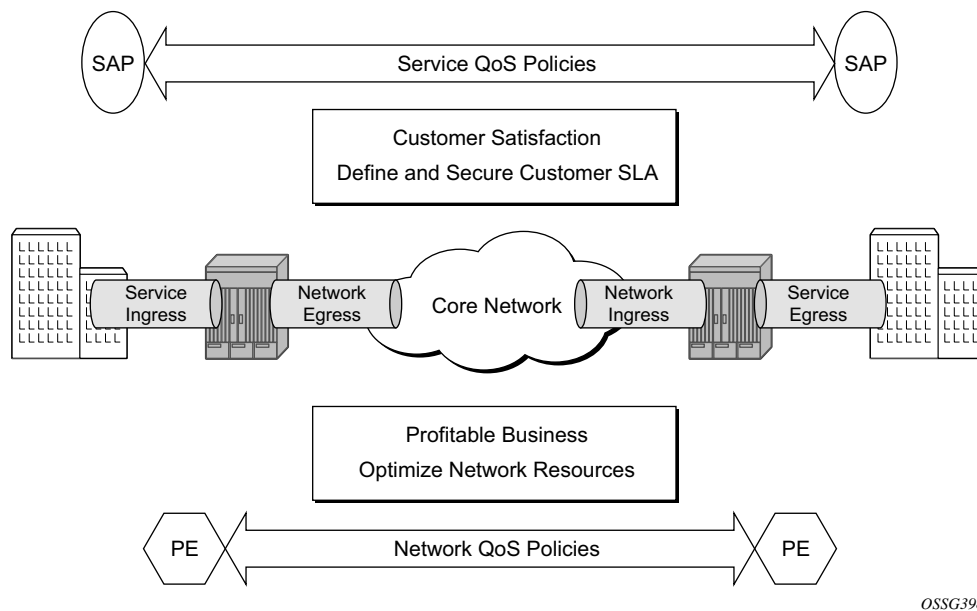
Hybrid ports are only supported on Ethernet ports and optionally with a single-chassis LAG. They must be configured to use VLANs (either single (dot1q encapsulation) or double (QinQ encapsulation) tagging) with each VLAN mapping to either the access or network part of the port. This allows the classification to associated incoming traffic with the correct port type and service.

Note that port based traffic, such as LACP, CCM and EFM, uses a system queue on an access port but the default network queues on a hybrid port.

Customer traffic follows the path shown below:

[service ingress → network egress] ingress PE → [network ingress → network egress] transit P → [network ingress → service egress] egress PE

The network administrator needs to make sure that QoS is configured correctly at each point using the appropriate QoS policies ([Figure 258](#)).



OSSG398

Figure 258: Service and Network QoS Policies

Service Ingress QoS Policy

The SAP ingress policies are created under the *qos* node of the CLI and require a unique identifier (from 1 to 65535). The default *sap-ingress* policy has identifier 1.

Classification

Services can be delineated at the SAP ingress by

- A physical port (null encapsulated) or
- An encapsulation on the physical port, for example a VLAN ID on an Ethernet port or a DLCI on a Frame Relay port.

The following configuration is an example of an IES service created with an IP interface on VLAN 2 of port 3/2/10 (IOM 3, MDA 2, port 10) and has SAP ingress QoS policy 10 applied.

```
configure service
  ies 1 customer 1 create
    interface "int-access" create
      address 192.168.1.1/30
      sap 3/2/10:2 create
        ingress
          qos 10
        exit
      exit
    exit
  exit
  no shutdown
exit
```

As traffic enters the port, the service can be identified by the VLAN tag (and unwanted packets dropped). The ingress service QoS policy applied to the SAP maps traffic to FCs, and thus to queues, and sets the enqueueing priority. Mapping flows to FCs is controlled by comparing each packet to the match criteria in the QoS policy. The match criteria that can be used in ingress QoS policies can be combinations of those listed in [Table 14](#). Note that when a packet matches two criteria (802.1p priority and DSCP) it is the lowest precedence value that is used to map the packet to the FC.

Table 14: SAP Ingress Classification Match Criteria

Match Precedence	Match Criteria		
1	IPv4 fields match criteria: <ul style="list-style-type: none"> • Destination IP address/prefix • Destination port/range • DSCP value • IP fragment • Protocol type (TCP, UDP, etc.) • Source port/range • Source IP address/prefix 	IPv6 fields match criteria: <ul style="list-style-type: none"> • Destination IP address/prefix • Destination port/range • DSCP value • Next header • Source port/range • Source IP address/prefix 	MAC fields match criteria: <ul style="list-style-type: none"> • Frame type [802dot3 802dot2-llc 802dot2-snap ethernetII atm] • ATM VCI value • IEEE 802.1p value/mask • Source MAC address/mask • Destination MAC address/mask • EtherType value • IEEE 802.2 LLC SSAP value/mask • IEEE 802.2 LLC DSAP value/mask • IEEE 802.3 LLC SNAP OUI zero or non-zero value • IEEE 802.3 LLC SNAP PID value
	Note: For an ingress QoS policy, either IP match criteria or MAC match criteria can be defined, not both.		
2	DSCP		
3	IP precedence		
4	LSP EXP		
5	IEEE 802.1p priority and/or Drop Eligibility Indicator (DEI)		
6	Default forwarding class for non-matching traffic		

It is possible to match MAC criteria on VPLS/Epipe SAPs and IP criteria on IP interface SAPs. However, it is also possible to classify on MAC criteria on an IP interface SAP and conversely to classify on IP criteria on VPLS/Epipe SAPs. When MPLS labeled traffic is received on a VPLS/Epipe SAP, it is possible to match on either of the LSP EXP bits (outer label) or the MAC criteria.

A SAP can be configured to have no VLAN tag (null encapsulated), one VLAN tag (dot1q encapsulated) or two VLAN tags (QinQ encapsulated). The configuration allows the selection of which VLAN tag to match against for the 802.1p bits, using the keyword **match-qinq-dot1p** with the keyword **top** or **bottom**.

The following example configuration shows match QinQ traffic with dot1p value 1 in the top VLAN tag entering the QinQ SAP in Epipe service 1 and assign it to FC **af** using queue 2.

Service Ingress QoS Policy

```
configure qos
  sap-ingress 10 create
    queue 2 create
    exit
    fc "af" create
      queue 2
    exit
    dot1p 1 fc "af"
  exit
exit

configure service
  epipe 1 customer 1 create
    sap 1/1/1:100.1 create
      ingress
        qos 10
        match-qinq-dot1p top
      exit
    exit
  no shutdown
exit
```

The classification of traffic using the default, **top** and **bottom** keyword parameters is summarized in [Table 15](#). Note that a TopQ SAP is a QinQ SAP where only the outer (top) VLAN tag is explicitly specified (sap 1/1/1:10.* or sap 1/1/1:10.0).

Table 15: QinQ Dot1p Bit Classification

Port/SAP Type	Existing Packet Tags	Pbits Used for Match		
		Default	Match Top	Match Bottom
Null	None	None	None	None
Null	Dot1P (VLAN-ID 0)	Dot1P PBits	Dot1P PBits	Dot1P PBits
Null	Dot1Q	Dot1Q PBits	Dot1Q PBits	Dot1Q PBits
Null	TopQ BottomQ	TopQ PBits	TopQ PBits	BottomQ PBits
Null	TopQ (No BottomQ)	TopQ PBits	TopQ PBits	TopQ PBits
Dot1Q	None (Default SAP)	None	None	None
Dot1Q	Dot1P (Default SAP VLAN-ID 0)	Dot1P PBits	Dot1P PBits	Dot1P PBits
Dot1Q	Dot1Q	Dot1Q PBits	Dot1Q PBits	Dot1Q PBits
QinQ/TopQ	TopQ	TopQ PBits	TopQ PBits	TopQ PBits
QinQ/TopQ	TopQ BottomQ	TopQ PBits	TopQ PBits	BottomQ PBits
QinQ/QinQ	TopQ BottomQ	BottomQ PBits	TopQ PBits	BottomQ Pbits

The Drop Eligibility Indicator (DEI)¹ bit can be used to indicate the in/out profile state of the packet, this will be covered later in the discussion on profile mode.

Note that ingress traffic with a local destination (for example, OSPF hellos) is classified by the system automatically and uses a set of dedicated system queues.

1. IEEE 802.1ad-2005 and IEEE 802.1ah (PBB)

After the traffic has been classified, the next step is to assign it to a given FC. There are 8 pre-defined FCs within the system which are shown in [Table 16](#) (note that the FC identifiers are keywords and do not have a fixed relationship with the associated Differentiated Services Code Points (DSCP)).

Table 16: Forwarding Classes

FC Identifier	FC Name	Default Scheduling Priority
NC	Network Control	Expedited
H1	High-1	Expedited
EF	Expedited	Expedited
H2	High-2	Expedited
L1	Low-1	Best Effort
AF	Assured	Best Effort
L2	Low-2	Best Effort
BE	Best Effort	Best Effort

When a FC is configured for a classification, it must first be created in the configuration. One of the FCs can be also configured to be the default in case there is no explicit classification match and by default this FC is **be**.

Normally, once traffic is assigned to a FC at the ingress it remains in that FC throughout its time within the system. Re-classification of IP traffic at a SAP egress is possible, but is beyond the scope of this note.

Packets also have a state of being in-profile or out-of-profile which represents their drop precedence within the system, therefore there can be up to 8 distinct per hop behavior (PHB) classes with two drop precedences.

Buffering (Enqueuing)

Once a packet is assigned to a certain forwarding class, it will try to get a buffer in order to be enqueued. Whether the packet can get a buffer is determined by the instantaneous buffer utilization and several attributes of the queue (such as Maximum Burst Size (MBS), Committed Burst Size (CBS) and high-prio-only) that will be discussed in more detail later in this chapter. If a packet cannot get a buffer for whatever reason, the packet will get dropped immediately.

As traffic is classified at the SAP ingress it is also assigned an enqueueing priority, which can be high or low. This governs the likelihood of a packet being accepted into a buffer and so onto a queue, and is managed using the queue's high-priority-only parameter and the buffer pools weighted random early detection (WRED) slope policies. Traffic having a high enqueueing priority has more chance of getting a buffer than traffic with low enqueueing priority. The enqueueing priority is specified with the classification using the parameter *priority*, and a default enqueueing priority can be configured, its default being low.

Enqueueing priority is a property of a packet and should not to be confused with scheduling priority, expedited or best-effort, which is a property of a queue.

The following configuration shows an example with all packets with dot1p value 3 are classified as ef and have their enqueueing priority set to high, all other packets are classified as **af** with a low enqueueing priority.

```
configure qos
  sap-ingress 10 create
    fc "af" create
    exit
    fc "ef" create
    exit
    dot1p 3 fc "ef" priority high
    default-fc "af"
    default-priority low # this is the default
  exit
```

Each forwarding class is associated with at most one unicast queue. In the case of a VPLS service, each FC can also be assigned a single multipoint queue at ingress, or for more granular control, separate queues for broadcast, multicast and unknown traffic. Since each queue maintains forward/drop statistics, it allows the network operator to easily track unicast, broadcast, multicast and unknown traffic load per forwarding class. Separate multicast queues can also be assigned for IES/VP RN services which have IP multicast enabled.

This results in an Epipe SAP having up to 8 ingress queues, an IES/VP RN SAP having up to 16 ingress queues and a VPLS SAP having up to 32 ingress queues. Each queue has a locally significant (to the policy) identifier, which can be from 1 to 32.

The default SAP ingress QoS policy (id=1) has two queues; queue 1 for unicast traffic and queue 11 for multipoint traffic, and is assigned to every ingress SAP at service creation time. Equally, when a new (non-default) SAP ingress policy is created, queue 1 and queue 11 are automatically created with the default FC (BE) assigned to both. Additional queues must be created before being

assigned to a FC, with multipoint queues requiring the **multipoint** keyword. When a SAP ingress policy is applied to a SAP, physical hardware queues on the IOM are allocated for each queue with a FC assigned (if no QoS policy is explicitly configured, the default policy is applied). Multipoint queues within the SAP ingress policy are ignored when applied to an Epipe SAP or an IES/VRPN SAP which is not configured for IP multicast.

The mechanism described here uses a separate set of queues per SAP. For cases where per-SAP queuing is not required it is possible to use port based queues, known as *queue-groups*, which reduces the number of queues required.

Scheduling (Dequeuing)

A queue has a priority which effects the relative scheduling of packets from it compared to other queues. There are two queue priorities: expedite and best-effort, with expedited being the higher. When creating a queue, one of these priorities can be configured thereby explicitly setting the queue's priority. Alternatively the default is auto-expedite in which case the queue's priority is governed by the FCs assigned to it, as shown in [Table 16](#). If there is a mix of expedited and best-effort FCs assigned, the queue is deemed to be best-effort.

The following configuration displays an example that ensures that EF traffic is treated as expedited by assigning it to new unicast and multicast queues.

```
configure qos
  sap-ingress 10 create
    queue 3 expedite create
  exit
  queue 13 multipoint expedite create
  exit
  fc ef create
    queue 3
    multicast-queue 13
  exit
  default-fc "ef"
exit
```

Once a packet gets a buffer and is queued, it will wait to be served and sent through the switch fabric to its destination port by the hardware scheduler. There are two scheduler priorities: expedited or best-effort, corresponding to the queue's priority. The expedited hardware schedulers are used to enforce priority access to internal switch fabric destinations with expedited queues normally having a higher preference than best-effort queues. Queues of the same priority get equally serviced in round robin fashion by the associated scheduler.

When a queue gets its turn to be serviced, the scheduler will use the operational Peak Information Rate (PIR) and Committed Information Rate (CIR) attributes of the queue to determine what to do with the packet.

- The scheduler does not allow queues to exceed their configured PIR. If the packet arrival rate for a given queue is higher than the rate at which it is drained, the queue will fill. If

the queue size (in Kbytes) reaches its defined MBS all subsequent packets will be discarded, this is known as tail drop.

- If the dequeue rate is below the operational CIR, the packet will be forwarded and marked as **in-profile**.
- If the dequeue rate is below the operational PIR but higher than the CIR, the packet will be forwarded but marked as **out-of-profile**.

Out-of-profile packets have a higher probability of being dropped when there is congestion somewhere in the downstream path. Packets that are marked with out-of-profile will also be treated differently at the network egress and service egress.

These marking actions are known as color marking (green for in-profile and yellow for out-of-profile). Using the default queue setting of **priority-mode**, as described above, the in/out-of-profile state of a packet is determined from the queue scheduling state (within CIR or above CIR, as described later) at the time that the packet is dequeued. An alternative queue mode is **profile-mode**.

Profile Mode

A queue is created with profile mode when the aim is that the in/out-of-profile state of packets is determined by the QoS bits of the incoming packets, this is known as color-aware (as opposed to color-unaware for priority mode).

As part of the classification, the profile state of the packets is explicitly configured. To provide granular control, it is possible to configure FC sub-classes with each having a different profile state, while inheriting the other parameters from their parent FC (for example the queue, in order to avoid out of order packets). The FC subclasses are named *fc.sub-class*, where *sub-class* is a text string up to 29 characters (though normally the words **in** and **out** are used for clarity). Any traffic classified without an explicit profile state is treated as if the queue were in priority mode.

When using the profile mode, the DEI in the Ethernet header can be used to classify a packet as in-profile (DEI=0) or out-of-profile (DEI=1).

The following configuration shows traffic with dot1p 3 is set to in-profile, dot1p 2 to out-of-profile and the profile state of dot1p 0 depends on the scheduling state of the queue.

```
configure qos
  sap-ingress 20 create
    queue 2 profile-mode create
  exit
  fc "af" create
    queue 2
  exit
  fc "af.in" create
    profile in
  exit
  fc "af.out" create
```

```
        profile out
    exit
    dot1p 0 fc "af"
    dot1p 2 fc "af.out"
    dot1p 3 fc "af.in"
exit
```

The difference between a queue configured in priority (default) and profile mode is summarized in [Table 17](#) (within/above CIR is described later).

Table 17: Queue Priority vs. Profile Mode

	Priority Mode	Profile Mode
Packet In-Profile/ Out-of-Profile state	Determined by state of the queue at scheduling time. Within CIR – In Profile Above CIR – Out Profile	Explicitly stated in FC or subclass classification. If not, then defaults to state of the queue at scheduling time
Packet High/Low Enqueuing Priority	Explicitly stated in FC classification. If not then defaults to Low priority	Always follows state of in-profile/out-of-profile determined above In-profile = High Priority Out-Profile = Low Priority If not set = High Priority

Remarking

Remarking at the service ingress is possible when using an IES or VPRN service. The DSCP/precedence field can be remarked for in-profile (**in-remark**) and out-of-profile (**out-remark**) traffic as defined above for queues in either priority mode or profile mode. If configured for other services, the remarking is ignored. If remarking is performed at the service ingress, then the traffic is not subject to any egress remarking on the same system.

The following configuration displays an example classifying traffic to 10.0.0.0/8 as FC **ef** in-profile and remark its DSCP to **ef**.

```
configure qos
  sap-ingress 300 create
    queue 2 profile-mode create
  exit
  fc "ef" create
    queue 2
    profile in
    in-remark dscp ef
  exit
  ip-criteria
    entry 10 create
      match
        dst-ip 10.0.0.0/8
      exit
      action fc "ef"
    exit
  exit
exit
```

Service Egress QoS Policy

The service egress uses a SAP egress QoS policy to define how FCs map to queues and how a packet of a given FC is remarked. SAP egress policies are created in the CLI qos context and require a unique identifier (from 1 to 65535). The default SAP egress policy has identifier 1.

Once a service packet is delivered to the egress SAP, it has following attributes:

- Forwarding class, determined from classification at the ingress of the node.
- High/low enqueueing priority, which corresponds directly to the in/out-of-profile state from the service ingress or network ingress.

Similar to the service ingress enqueueing process, it is possible that a packet can not get a buffer and thus gets dropped. Once on an egress queue, a packet is scheduled from the queue based on priority of the queue (expedited or best-effort) and the scheduling state with respect to the CIR/PIR rates (note that the profile state of the packet [in/out] is not modified here). Egress queues do not have a priority/profile mode and have no concept of multipoint.

Only one queue exists in the default SAP egress QoS policy (id=1) and also when a new *sap-egress* policy is created, this being queue 1 which is used for both unicast traffic and multipoint traffic. All FCs are assigned to this queue unless otherwise explicitly configured to a different configured queue. When a SAP egress policy is applied to a SAP, physical hardware queues on the IOM are allocated for each queue with FC assigned (if no QoS policy is explicitly configured, the default policy is applied).

As mentioned earlier, re-classification of IP traffic at a SAP egress is possible.

Traffic originated by the system (known as self generated traffic) has its FC and marking configured under *router/sgt-qos* (for the base routing) or under *service/vprn/sgt-qos* (for a VPRN service). This is beyond the scope of this note.

Remarking

At the service egress, the dot1p/DEI can be remarked for any service per FC with separate marking for in/out-of-profile if required. The DEI bit can also be forced to a specific value (using the **de-mark force** command). When no dot1p/de-mark is configured, the ingress dot1p/DEI is preserved; if the ingress was un-tagged the dot1p/DEI bit is set to 0.

The following configuration shows a remark example with different FCs with different dot1p values. FC **af** also differentiates between in/out-of-profile and then remarks the DEI bit accordingly based on the packet's profile.

```
configure qos
  sap-egress 10 create
    queue 1 create
    rate 20000
    exit
    queue 2 create
    rate 10000 cir 5000
    exit
    queue 3 create
    rate 2000 cir 2000
    exit
    fc af create
    queue 2
    dot1p in-profile 3 out-profile 2
    de-mark
    exit
    fc be create
    queue 1
    dot1p 0
    exit
    fc ef create
    queue 3
    dot1p 5
    exit
  exit
```

If QinQ encapsulation is used, the default is to remark both tags in the same way. However it is also possible to remark only the top tag using the **qinq-mark-top-only** parameter configured under the SAP egress.

The following configuration shows a remark example with only the dot1p/DEI bits in top tag of a QinQ SAP.

```
configure service
  vpls 2 customer 1 create
    sap 1/1/11:2.2 create
    egress
    qos 20
    qinq-mark-top-only
    exit
  exit
exit
```

Service Egress QoS Policy

For IES and VPRN services, the DSCP/precedence field can be remarked in the same way as at the service ingress, namely based on the in/out-of-profile state of the packets (and only if no ingress remarking was performed).

The following configuration shows DSCP values for FC **af** based on in/out-of-profile traffic.

```
configured qos
  sap-egress 20 create
    queue 2 create
    fc af create
      queue 2
      dscp in-profile af41 out-profile 43
    exit
  exit
```


Network Ports

The QoS policies relating to the network ports are divided into a network and a network-queue policy. The network policy covers the ingress classification into FCs and the egress remarking based on FCs, while the network-queue policy covers the queues/parameters and the FC to queue mapping. The logic behind this is that there is only one set of queues provisioned on a network port, whereas the use of these queues is configured per network IP interface. This in turn determines where the two policies can be applied. Note that network ports are used for IP routing and switching, and for GRE/MPLS tunneling.

Network QoS Policy

The network QoS policy has an ingress section and an egress section. It is created under the *qos* node of the CLI and requires a unique identifier (from 1 to 65535). The default network policy has identifier 1. Network QoS policies are applied to IP interfaces configured on a network port.

The following configuration show an example to apply different network QoS policies to two network interfaces.

```
configure router
  interface "int-network-1"
    address 192.168.0.1/30
    port 1/1/11:1
    qos 28
  exit
  interface "int-network-2"
    address 192.168.0.5/30
    port 1/1/12
    qos 18
  exit
exit
```

Classification

The ingress section defines the classification rules for IP/MPLS packets received on a network IP interface. The rules for classifying traffic are based on the incoming QoS bits (Dot1p, DSCP, EXP [MPLS experimental bits]). The order in which classification occurs relative to these fields is:

1. EXP (for MPLS packets) or DSCP (for IP packets)
Dot1p/DEI bit ²
2. default action (default= fc be profile out)

2. Note that network ports do not support QinQ encapsulation.

The configuration specifies the QoS bits to match against the incoming traffic together with the FC and profile (in/out) to be used (it is analogous to the SAP profile-mode in that the profile of the traffic is determined from the incoming traffic, rather than the CIR configured on the queue). A **default-action** keyword configures a default FC and profile state.

For tunneled traffic (GRE or MPLS), the match is based on the outer encapsulation header unless the keyword **ler-use-dscp** is configured. In this case, traffic received on the router terminating the tunnel that is to be routed to the base router or a VPRN destination is classified based on the encapsulated packet DSCP value (assuming it is an IP packet) rather than its EXP bits.

Note that Release 8.0 added the ability for an egress LER to signal an implicit-null label (numeric value 3). This informs the previous hop to send MPLS packets without an outer label and so is known as penultimate hop popping (PHP). This can result in MPLS traffic being received at the termination of an LSP without any MPLS labels. In general, this would only be the case for IP encapsulated traffic, in which case the egress LER would need to classify the incoming traffic using IP criteria.

Remarking

The egress section of the network policy defines the remarking of the egress packets, there is no remarking possible at the network ingress. The egress remarking is configured per FC and can set the related dot1p/DEI (explicitly or dependant on in/out-of-profile), DSCP (dependent on in/out-of-profile) and EXP (dependent on in/out-of-profile).

The traffic exiting a network port is either tunneled (in GRE or MPLS) or IP routed.

For tunneled traffic exiting a network port, the remarking³ applies to the DSCP/EXP bits in any tunnel encapsulation headers (GRE/MPLS) pushed⁴ onto the packet by this system, together with the associated dot1p/DEI bits if the traffic has an outer VLAN tag. Note that for MPLS tunnels, the EXP bits in the entire label stack are remarked.

For VPLS/Epipe services there is no additional remarking possible. However, for IES/VPRN/ base-routing traffic the remarking capabilities at the network egress are different at the first network egress (egress on the system on which the traffic entered by a SAP ingress) and subsequent network egress in the network (egress on the systems on which the traffic entered through another network interface).

At the first network egress, the DSCP of the routed/tunneled IP packet can be remarked but this is dependent on two configuration settings:

-
3. Strictly speaking this is marking (as opposed to remarking) as the action is adding QoS information rather than changing it.
 4. A new outer encapsulation header is pushed onto traffic at each MPLS transit label switched router as part of the label swap operation.

- The trusted state of the ingress (service/network) interface and
- The **marking** keyword in the network QoS policy at the network egress. The configuration combinations are summarized in [Table 18](#).

This is in addition to the remarking of any encapsulation headers and, as stated earlier, is not performed if the traffic was remarked at the service ingress.

For traffic exiting a subsequent network egress in the network, only the IP routed traffic can be remarked, again this is dependent on the ingress trusted state and egress remarking parameter.

There is one addition to the above to handle the marking for IP-VPN Option-B in order to remark the EXP, DSCP and dot1p/DEI bits at a network egress, this being **marking force**. Without this, only the EXP and dot1p/DEI bits are remarked. Note that this does not apply to label switched path traffic switched at a label switched router.

Table 18: Network QoS Policy DSCP Marking

Ingress	Trusted State	Marking Configuration	Marking Performed
IES	Untrusted (default)	marking	Yes
		no marking (default)	Yes
	Trusted	marking	Yes
		no marking (default)	No
Network	Untrusted	marking	Yes
		no marking (default)	Yes
	Trusted (default)	marking	Yes
		no marking (default)	No
VPRN	Untrusted	marking	Yes
		no marking (default)	Yes
	Trusted (default)	marking	No
		no marking (default)	No

The following configuration shows a ingress network classification for DSCP EF explicitly, with a default action for the remainder of the traffic and use the DSCP from the encapsulated IP packet if terminating a tunnel. Remark the DSCP values for FC **af** and **ef** and remark all traffic (except incoming VPRN traffic) at the egress. Apply this policy to a network interface.

Network QoS Policy

```
configure qos
  network 20 create
    ingress
      default-action fc af profile out
      ler-use-dscp
      dscp ef fc ef profile in
    exit
    egress
      remarking
      fc af
        no dscp-in-profile
        dscp-out-profile af13
        lsp-exp-in-profile 6
        lsp-exp-out-profile 5
      exit
      fc ef
        dscp-in-profile af41
      exit
    exit
  exit
exit
configure router
  interface "int-network-3"
    address 192.168.0.9/30
    port 1/1/3
    qos 20
  exit
```

The following configuration shows the trusted IES interface.

```
configure service
  ies 1 customer 1 create
    interface "int-access" create
      address 192.168.1.1/30
      tos-marking-state trusted
      sap 1/1/10:1 create
    exit
  exit
  no shutdown
exit
```

The network QoS egress section also contains the configuration for the use of port-based queues by queue-groups which are out of scope of this note.

Network Queue Policy

The network queue QoS policy defines the queues and their parameters together with the FC to queue mapping. The policies are named, with the default policy having the name **default** and are applied under **config>card>mda>network>ingress** for the network ingress queues and under Ethernet: **config>port>ethernet>network**, POS: **config>port>sonet-sdh>path>network**, TDM: **config>port>tdm>e3 | ds3>network** for the egress.

The following configuration shows an ingress and egress network-queue policy.

```
configure card 1
  card-type iom3-xp
  mda 1
    mda-type m20-1gb-xp-sfp
    network
      ingress
        queue-policy "network-queue-1"
      exit
    exit
  exit
exit

configure port 1/1/11
  ethernet
    encap-type dot1q
    network
      queue-policy "network-queue-1"
    exit
  exit
  no shutdown
exit
```

There can be up to 16 queues configured in a network-queue policy, each with a queue-type of best-effort, expedite or auto-expedite. A new network-queue policy contains two queues, queue 1 for unicast traffic and queue 9 for multipoint traffic and by default all FCs are mapped to these queues. Note that there is no differentiation for broadcast, multicast and unknown traffic. If the policy is applied to the egress then any multipoint queues are ignored. As there are 8 FCs, there would be up to 8 unicast queues and 8 multipoint queues, resulting in 16 ingress queues and 8 egress queues. Normally the network queue configuration is symmetric (the same queues/FC-mapping at the ingress and egress).

The following configuration defines a network-queue policy with FC **af** and **ef** assigned to queues 2 and 3 for unicast traffic, and queue 9 for multipoint traffic.

```
configure qos
  network-queue "network-queue-1" create
    queue 1 create
      mbs 50
      high-prio-only 10
    exit
    queue 2 create
  exit
```

Network Queue Policy

```
queue 3 create
exit
queue 9 multipoint create
    mbs 50
    high-prio-only 10
exit
fc af create
    multicast-queue 9
    queue 2
exit
fc ef create
    multicast-queue 9
    queue 3
exit
exit
```

Summary of Network Policies

Figure 259 displays the default network policies with respect to classification, FC to queue mapping and remarking.

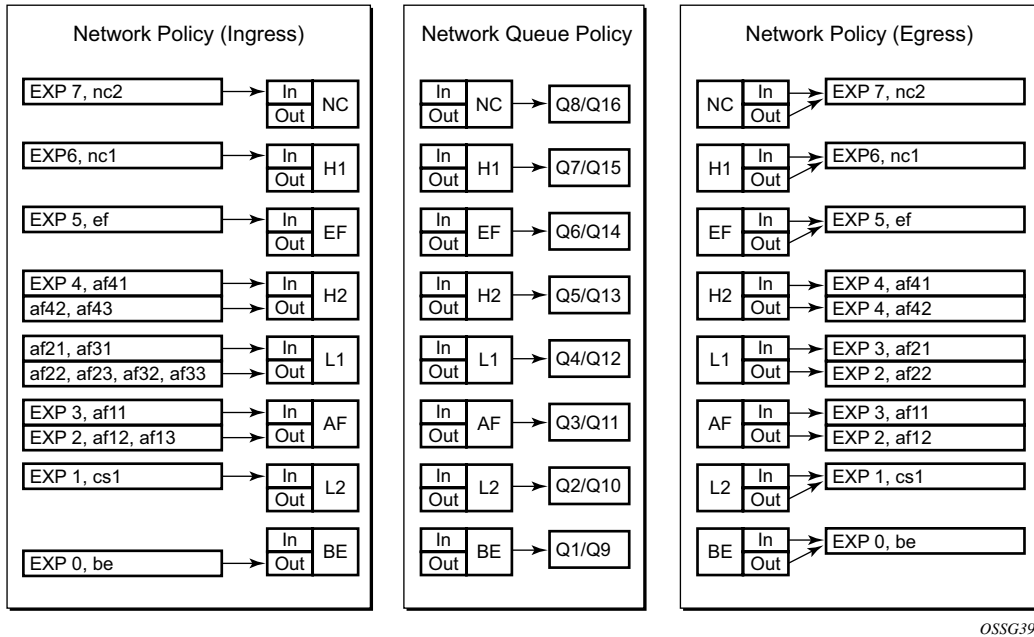


Figure 259: Visualization of Default Network Policies

Queue Management

The policies described so far define queues but not the characteristics of those queues which determine how they behave. This section describes the detailed configuration associated with these queues. There are two aspects:

- Enqueuing packets onto a queue
 - buffer pools
 - queue sizing
 - Weight Random Early Detection (WRED)
 - Dequeuing packets from a queue
 - queue rates
 - scheduling
-

Enqueuing Packets: Buffer Pools

The packet buffer space is divided equally between ingress and egress. Beyond that, by default there is one pool for network ingress per FP2⁵/IOM, with one pool per access ingress port and one pool per access/network egress port. This is shown in [Figure 260](#). This segregation provides isolation against buffer starvation between the separate pools. An additional ingress pool exists for managed multicast traffic (the multicast path management pool) but this is beyond the scope of this note.

The buffer management can be modified using named buffer pools and/or WRED-per-queue pools which are out of scope of this note.

5. The FP2 chipset is used in the IOM3-XP/IMM and in the 7750 SR-c 12/4.

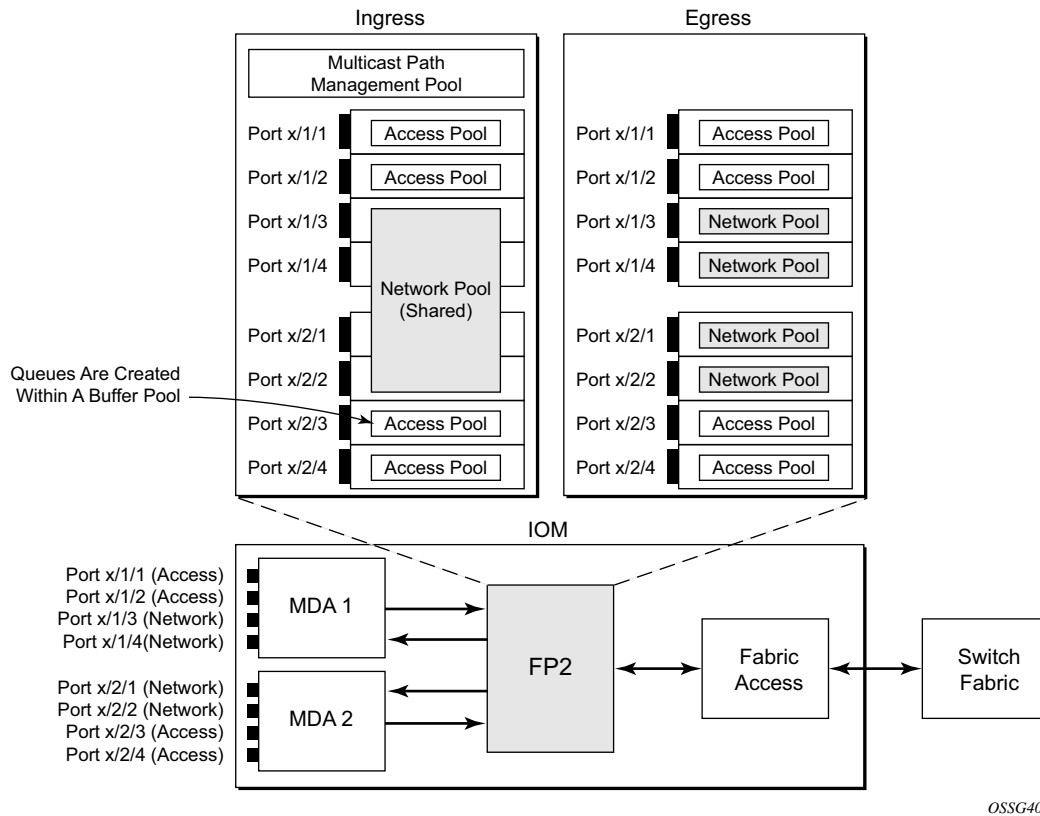


Figure 260: Default Buffer Pools

The size of the pools is based on the MDA type and the speed/type (access or network) of each port. Buffer space is allocated in proportion to the active bandwidth of each port, which is dependant on:

- The actual speed of the port
- Bandwidth for configured channels only (on channelized cards)
- Zero for ports without queues configured

This calculation can be tuned separately for ingress and egress, without modifying the actual port speed, using the port/modify-buffer-allocation-rate. Note that changing the port's egress-rate will also modify its buffer sizes.

The following configuration changes the relative size for the ingress/egress buffer space on port 1/1/10 to 50% of the default.

```
configure port 1/1/10
  modify-buffer-allocation-rate
    ing-percentage-of-rate 50
    egr-percentage-of-rate 50
  exit
```

Each of the buffer pools created is further divided into a section of reserved buffers and another of shared buffers, see [Figure 262](#). The amount of reserved buffers is calculated differently for network and access pools. For network pools, the default is approximately the sum of the CBS (committed burst size) values defined for all of the queues within the pool. The reserved buffer size can also be statically configured to a percentage of the full pool size (ingress: **config>card>mda>network>ingress>pool**; egress: **config>port>network>egress>pool**). For access pools, the default reserved buffer size is 30% of the full pool size and can be set statically to an explicit value (ingress: **config>port>access>ingress>pool**; egress: **config>port>access>egress>pool**).

The following configuration sets the reserved buffer size to 50% of the egress pool space.

```
configure port 1/1/10
  network
    egress
      pool
        resv-cbs 50
      exit
    exit
  exit
exit

configure port 1/1/11
  access
    egress
      pool
        resv-cbs 50
      exit
    exit
  exit
exit
```

Both the total buffer and the reserved buffer sizes are allocated in blocks (discrete values of Kbytes). The pool sizes can be seen using the **show pools** command.

It is possible to configure alarms to be triggered when the usage of the reserved buffers in the buffer pools reaches a certain percentage. Two alarm percentages are configurable, amber and red, **amber-alarm-threshold** <percentage> and **red-alarm-threshold** <percentage>. The percentage range is 1 — 1000.

- The percentage for the red must be at least as large as that for the amber.
- The alarms are cleared when the reserved CBS drops below the related threshold.

- When the amber alarm is enabled, dynamic reserved buffer sizing can be used; after the amber alarm is triggered the reserved buffer size is increased or decreased depending on the CBS usage. This requires a non-default resv-cbs to be configured together with a step and max value for the amber-alarm-action parameters. As the reserved CBS usage increases above the amber alarm percentage, the reserved buffer size is increased in increments defined by the step, up to a maximum of the max. If the CBS usage decreases, the reserved buffer size is reduced in steps down to its configured size.
- As the reserved buffer size changes, alarms will continue to be triggered at the same color (amber or red) indicating the new reserved buffer size. Note that the pool sizing is checked at intervals, so it can take up to one minute for the alarms and pool re-sizing to occur.

The following displays a configuration for access ingress and egress pools.

```
configure port 1/1/1
  access
    ingress
      pool
        amber-alarm-threshold 25
        red-alarm-threshold 50
        resv-cbs 20 amber-alarm-action step 5 max 50
      exit
    exit
  egress
    pool
      amber-alarm-threshold 25
      red-alarm-threshold 25
      resv-cbs 20 amber-alarm-action step 5 max 50
    exit
  exit
exit
```

The following is an example alarm that is triggered when the amber percentage has been exceeded and the reserved buffer size has increased from 20% to 25%:

```
19 2011/12/20 16:38:14.94 UTC MINOR: PORT #2050 Base Resv CBS Alarm
"Amber Alarm: CBS over Amber threshold: ObjType=port Owner=1/1/1 Type=accessEgre
ss Pool=default NamedPoolPolicy= Old ResvSize=13824 ResvSize=16128 SumOfQ ResvSi
ze=3744 Old ResvCBS=20 New ResvCBS=25"
```

When a port is configured to be a hybrid port, its buffer space is divided into an access portion and a network portion. The split by default is 50:50 but it can be configured on a per port basis.

```
configure port 1/1/1
  ethernet
    mode hybrid
    encap-type dot1q
  exit
  hybrid-buffer-allocation
    ing-weight access 70 network 30
    egr-weight access 70 network 30
  exit
```

Enqueuing Packets: Queue Sizing

Queue sizes change dynamically when packets are added to a queue faster than they are removed, without any traffic the queue depth is zero. When packets arrive for a queue there will be request for buffer memory which will result in buffers being allocated dynamically from the buffer pool that the queue belongs to.

A queue has three buffer size related attributes: MBS, CBS and high-prio-only, which affect packets only during the enqueuing process.

- **Maximum Burst Size (MBS)** defines the maximum buffer size that a queue can use. If the actual queue depth is equal to the MBS, any incoming packet will not be able to get a buffer and the packet will be dropped. This is defined in bytes or Kbytes for access queues with a configurable non-zero minimum of 1byte or a default (without configuring the MBS) of the maximum between 10ms of the PIR or 64Kbytes. A value of zero will cause all packets to be dropped. It is a fractional percentage (xx.xx%) of pool size for network queues with defaults varying dependant on the queue (see default network-queue policy for default values). The MBS setting is the main factor determining the packet latency through a system when packets experience congestion.
- **Committed Burst Size (CBS)** defines the maximum guaranteed buffer size for an incoming packet. This buffer space is effectively reserved for this queue as long as the CBS is not oversubscribed (such the sum of the CBS for all queues using this pool does not exceed its reserved buffer pool size). The CBS is defined in Kbytes with a configurable non-zero minimum of 6Kbytes or a default (without configuring the CBS) of the maximum between 10ms of the CIR or 6Kbytes. It is a fractional percentage (xx.xx%) of pool size for network queues with defaults varying dependant on the queue (see default network-queue policy for default values). Regardless of what is configured, the CBS attained will never be larger than the MBS. The only case where CBS could be configured larger than MBS is for queues on LAGs, as in some cases the CBS is shared among the LAG ports (LAG QoS is not covered in this document). If the MBS and CBS values are configured to be equal (or nearly equal) this will result in the CBS being slightly higher than the value configured.
- **High-prio-only.** As a queue can accept both high and low enqueueing priority packets, a high enqueueing priority packet should have a higher probability to get a buffer. High-prio-only is a way to achieve this. Within the MBS, high-prio-only defines that a certain amount of buffer space will be exclusively available for high enqueueing priority packets. At network ingress and all egress buffering, high corresponds to in-profile and low to out-of-profile. At service ingress, enqueueing priority is part of the classification. The high-prio-only is defined as a percentage of the MBS, with the default being 10%. Note that a queue being used only for low priority/out-of-profile packets would normally have this set to zero. The high-prio-only could be considered to be an MBS for low enqueueing/out-of-profile packets.

As with the buffer pools, the MBS, CBS and high-prio-only values attained are based on a number of discrete values (not always an increment of 3Kbytes). The values for these parameters can be seen using the **show pools** command.

As packets are added to a queue they will use the available CBS space, in which case they are placed in the reserved portion in the buffer pool. Once the CBS is exhausted, packets use the shared buffer pool space up to high-prio-only threshold (for out-of-profile packets) or the maximum MBS size (for in-profile packets).

The following configuration shows a queue with a specific MBS, CBS and disable high-prio-only.

```
configure qos
  sap-ingress 10 create
    queue 1 create
      mbs 10000
      cbs 100
      high-prio-only 0
    exit
  exit
```

Enqueuing Packets: Weight Random Earlier Detection (WRED)

In order to gracefully manage the use of the shared portion of the buffer pool, WRED can be configured on that part of the pool, and therefore applies to all queues in the shared pool as it fills. WRED is a congestion avoidance mechanism designed for TCP traffic. This note will only focus on the configuration of WRED. WRED-per-queue is an option to have WRED apply on a per egress queue basis, but is not covered here.

WRED is configured by a slope-policy which contains two WRED slope definitions, a high-slope which applies WRED to high enqueueing priority/in-profile packets and a low-slope which applies WRED to low enqueueing priority/out-of-profile packets. Both have the standard WRED parameters: start average (start-avg), maximum average (max-avg) and maximum probability (max-prob), and can be enabled or disabled individually. The WRED slope characteristics are shown in [Figure 261](#).

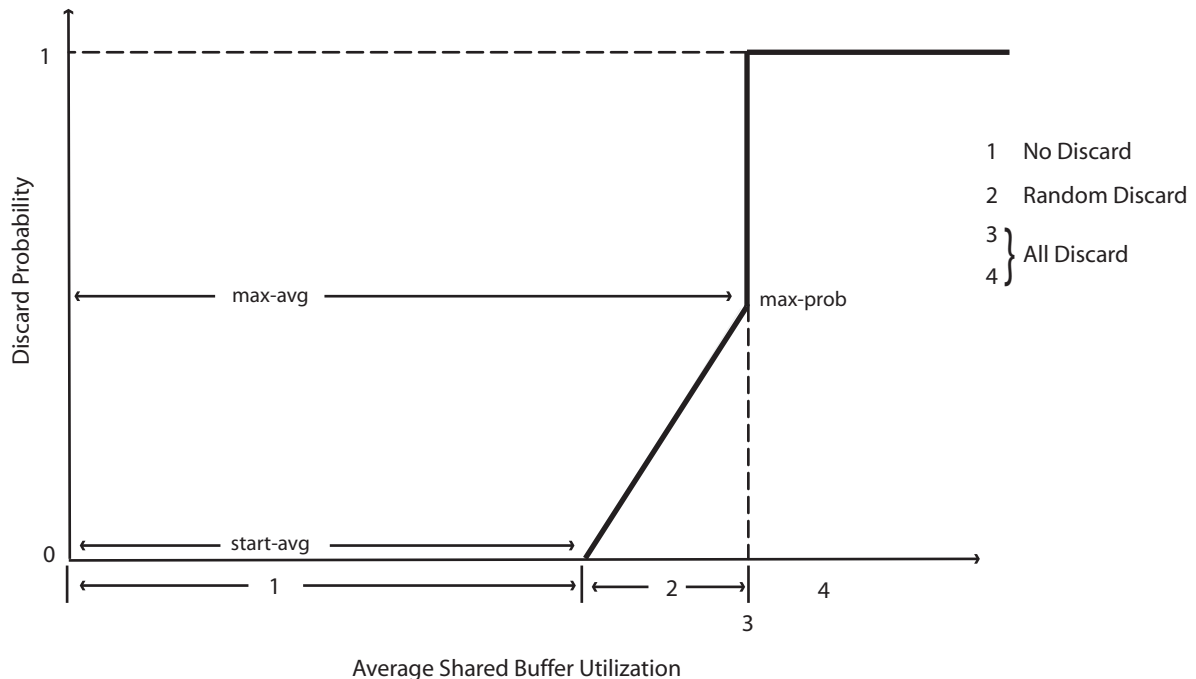


Figure 261: WRED Slope Characteristics

A time-average-factor parameter can be configured per slope-policy which determines the sensitivity of the WRED algorithm to shared buffer utilization fluctuations (the smaller the value makes the average buffer utilization more reactive to changes in the instantaneous buffer utilization). The slope-policy is applied on a network port under **config>card>mda>network>ingress>pool** and **port>network>egress>pool** and on an access port under **config>port>access>ingress>pool** and **config>port>access>egress>pool**.

WRED is usually configured for assured and best-effort service traffic with premium traffic not typically being subject to WRED as it is always given preferential treatment and should never be dropped.

The following configuration defines a WRED slope policy and apply it to an ingress access port.

```
configure qos
  slope-policy "slope1" create
    high-slope
      start-avg 80
      max-avg 100
      max-prob 100
      no shutdown
    exit
    low-slope
      max-avg 100
      start-avg 80
      max-prob 100
      no shutdown
    exit
    time-average-factor 12
  exit
exit

configure port 1/1/10
  access
    ingress
      pool
        slope-policy "slope1"
      exit
    exit
  exit
exit
```

The queue sizing parameters and buffer pools layout is shown in [Figure 262](#).

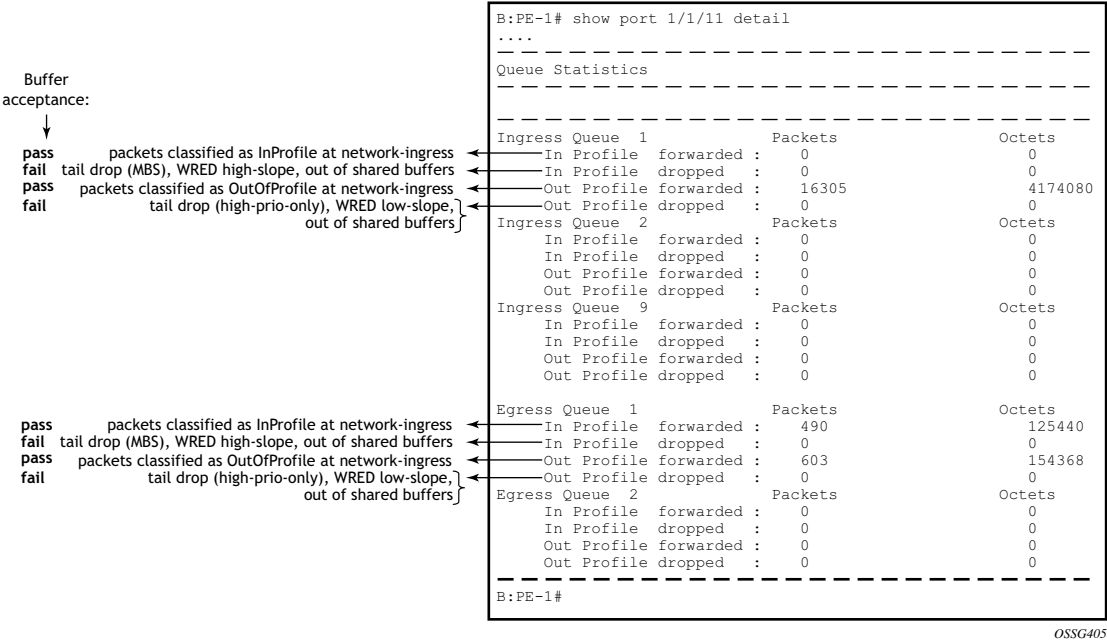


Figure 262: Buffer Pools and Queue Sizing

Dequeuing Packets: Queue Rates

A queue has two rate attributes: PIR and CIR. These affect packets only during the dequeue process.

- **PIR** — If the instantaneous dequeue rate of a queue reaches this rate, the queue is no longer served. Excess packets will be discarded eventually when the queue reaches its MBS/high-prio-only sizes. The PIR for access ports can be set in Kb/s with a default of **max** or as a percentage (see below). For network ports it is set as a percentage of 390000Kb/s for ingress queues and of the port speed for egress queues, both with a default of 100%.
- **CIR** — This is used to determine whether an ingress packet is in-profile or out-of-profile at the SAP ingress. It is also used by the scheduler in that queues operating within their CIRs will be served ahead of queues operating above their CIRs. The CIR for access ports can be set in Kb/s with a default of zero or as a percentage (see below). For network ports it is set as a percentage of 390000Kb/s for ingress queues and of the port speed for egress queues, with defaults varying dependant on the queue.

A percentage rate can be used in the sap-ingress and sap-egress policies, and can be defined relative to the local-limit (the parent scheduler rate) or the port-limit (the rate of the port on which the SAP is configured, including any egress-rate configured). The parameters rate and percent-rate are mutually exclusive and will overwrite each other when configured in the same policy. The example below shows a percent-rate configured as a port-limit.

```
config>qos#
  qos
    sap-egress 10 create
      queue 1 create
        percent-rate 50.00 cir 10.00 port-limit
      exit
    exit
```

The PIR and CIR rates are shown in [Figure 263](#).

The queues operate at discrete rates supported by the hardware. If a configured rate does not match exactly one of the hardware rates an adaptation rule can be configured to control whether the rate is rounded up or down or set to the closest attainable value. The actual rate used can be seen under the operational PIR/CIR (O.PIR/O.CIR) in the **show pools** command output.

The following configuration shows a queue with a PIR, CIR and adaptation rule.

```
configure qos
  sap-ingress 20 create
    queue 2 create
      adaptation-rule pir max cir min
      rate 10000 cir 5000
    exit
  exit
```

Queue Management

By default, the rates apply to packet bytes based on packet accounting, which for Ethernet includes the Layer 2 frame plus the FCS. An alternative is frame accounting which adds the Ethernet inter-frame gap, preamble and start frame delimiter.

Dequeuing Packets: Scheduling

Once a packet is placed on a queue, it is always dequeued from the queue by a scheduler. The scheduling order of the queues dynamically changes depending on whether a queue is currently operating below or above its CIR, with expedited queues being serviced before best-effort queues. This results in a default scheduling order of (in strict priority).

1. Expedited queues operating below CIR
2. Best-effort queues operating below CIR
3. Expedited queues operating above CIR
4. Best-effort queues operating above CIR

This is displayed in [Figure 263](#).

The scheduling order can be explicitly configured using hierarchical QoS (with a scheduler-policy or port-scheduler-policy) which is out of scope of this section.

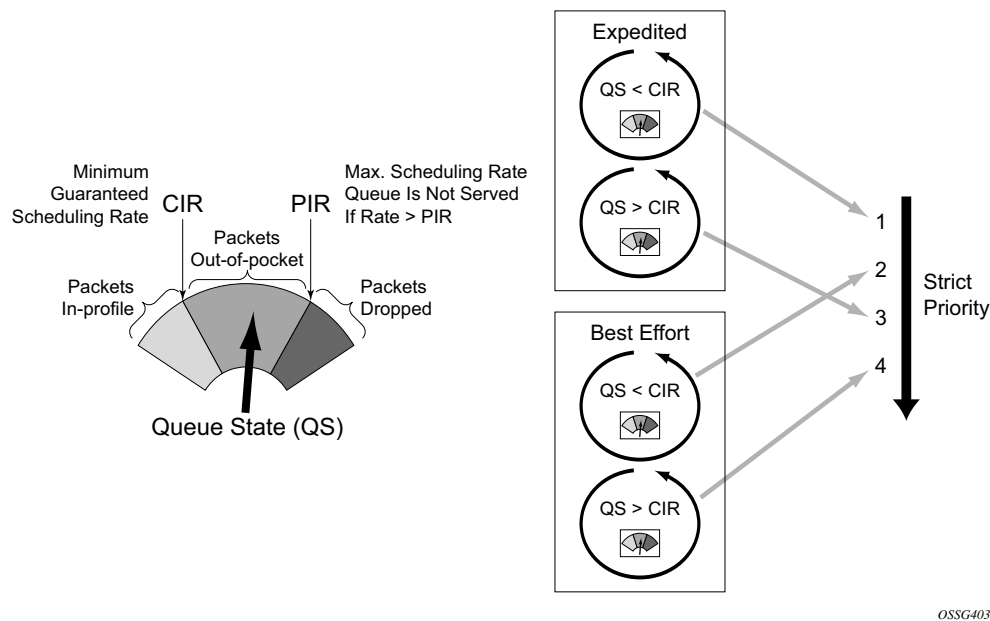
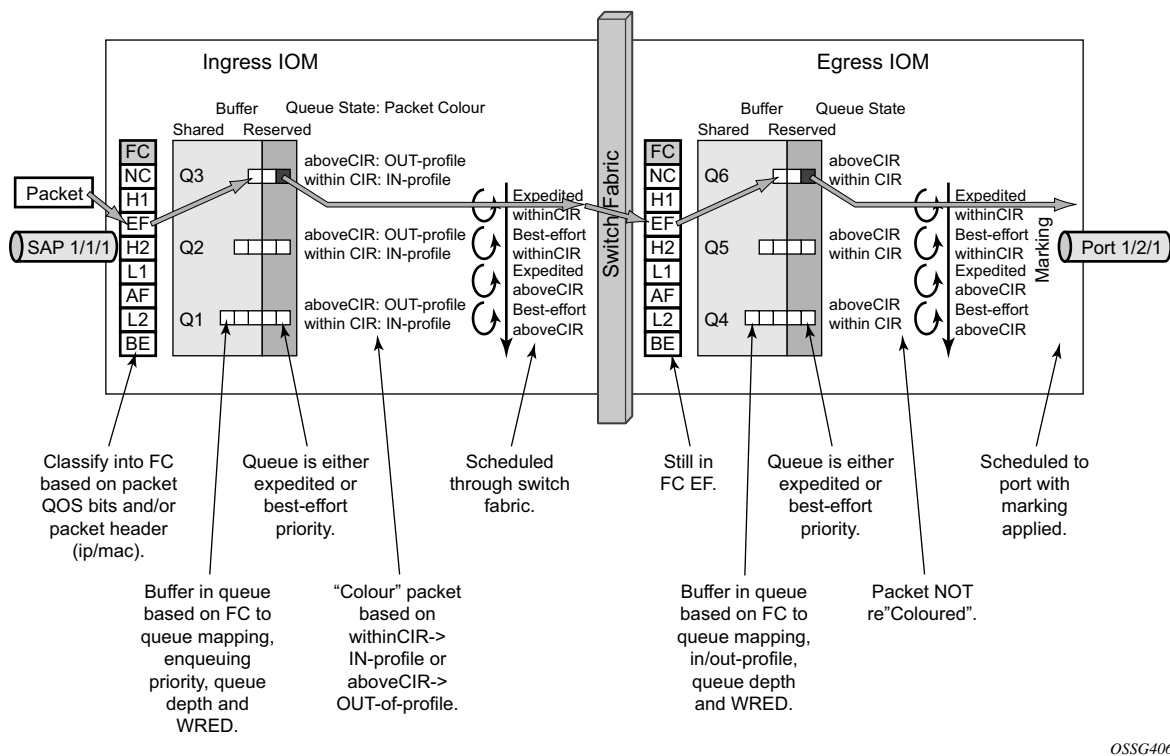


Figure 263: Scheduling (Dequeuing Packets from the Queue)

The overall QoS actions at both the ingress and egress IOMs are shown in [Figure 264](#).



OSSG406

Figure 264: IOM QoS Overview

Show Output

The following displays **show** command output for:

- SAP queue statistics
- port queue statistics
- access-ingress pools

The **show pools** command output for network-ingress and network/access-egress is similar to that of access-ingress and is not included here.

SAP Queue Statistics

The output below shows an example of the ingress and egress statistics on a SAP for an IES service (without multicast enabled, hence no ingress multicast queue). There are two ingress queues, one being in priority mode and the other in profile mode. An explanation of the statistics is given for each entry.

Buffer acceptance:



fail {
pass {

Based on sap-ingress classification {
tail drop (MBS), WRED high-slope, out of shared buffers
tail drop (high-prio-only), WRED low-slope, out of shared buffers
packets forwarded while queue was operating withinCIR
packets forwarded while queue was operating aboveCIR

fail {
pass {

Based on sap-ingress classification {
tail drop (high-prio-only), WRED low-slope, out of shared buffers
tail drop (MBS), WRED high-slope, out of shared buffers
ColorIn packets or Uncolor while queue was operating withinCIR
ColorOut packets or Uncolor while queue was operating aboveCIR

pass {
fail {

packets with profile state = InProfile (determined at ingress)
Packets with profile state = OutOfProfile (determined at ingress)
tail drop (MBS), WRED high-slope, out of shared buffers
tail drop (high-prio-only), WRED low-slope, out of shared buffers

```
B:PE-1# show service id 1 sap 1/1/10:1 stats
....
-----
Sap per Queue stats
-----
Packets      Octets
Ingress Queue 1 (Unicast) (Priority)
Off. HiPrio   : 0                0
Off. LoPrio   : 19022           4869632
Dro. HiPrio   : 0                0
Dro. LoPrio   : 17783           4552448
For. InProf   : 548             140288
For. OutProf  : 691             176896
Ingress Queue 2 (Unicast) (Profile)
Off. ColorIn  : 29439           7536384
Off. ColorOut : 0                0
Off. Uncolor  : 0                0
Dro. ColorOut : 0                0
Dro. ColorIn & Uncolor: 16193      4145408
For. InProf   : 17098           4377088
For. OutProf  : 0                0
Egress Queue 1
For. InProf   : 0                0
For. OutProf  : 48461           12406016
Dro. InProf   : 0                0
Dro. OutProf  : 0                0
=====
B:PE-1#
```

Port Queue Statistics

This output shows an example of the ingress and egress network port statistics. There are two unicast ingress queues (1 and 2) and one multicast ingress queue (9) with two egress queues. An explanation of the statistics is given for each entry.

Buffer acceptance:
↓
pass packets classified as InProfile at network-ingress
fail tail drop (MBS), WRED high-slope, out of shared buffers
pass packets classified as OutOfProfile at network-ingress
fail tail drop (high-prio-only), WRED low-slope, out of shared buffers

```
B:PE-1# show port 1/1/11 detail
....
=====
Queue Statistics
=====

-----
Ingress Queue 1
Packets      Octets
In Profile forwarded : 0          0
In Profile dropped  : 0          0
Out Profile forwarded : 16305     4174080
Out Profile dropped  : 0          0
Ingress Queue 2
Packets      Octets
In Profile forwarded : 0          0
In Profile dropped  : 0          0
Out Profile forwarded : 0          0
Out Profile dropped  : 0          0
Ingress Queue 9
Packets      Octets
In Profile forwarded : 0          0
In Profile dropped  : 0          0
Out Profile forwarded : 0          0
Out Profile dropped  : 0          0

Egress Queue 1
Packets      Octets
In Profile forwarded : 490        125440
In Profile dropped  : 0          0
Out Profile forwarded : 603        154368
Out Profile dropped  : 0          0
Egress Queue 2
Packets      Octets
In Profile forwarded : 0          0
In Profile dropped  : 0          0
Out Profile forwarded : 0          0
Out Profile dropped  : 0          0
=====
B:PE-1#
```

Access-Ingress Pools

This output shows an example of the default pools output for access-ingress. It includes the pools sizes, WRED information and queue parameters for each queue in the pool.

For this particular output, queue 3 on SAP 1/1/10:1 is being over-loaded which is causing its queue depth to be 6858Kbytes, made up of 5853Kbytes from the shared pool (in use) and 1008Kbytes from the reserved pool (in use). The output shows the pool total in usage as 6861Kbytes and the queue depth 3Kbytes less at 6858Kbytes, this is simply due to the dynamics of the buffer allocation which uses a 'sliding-window' mechanism and may therefore not always be perfectly aligned.

It can be seen that the high and low WRED slopes are both enabled and their instantaneous drop probability is shown 100% and their start/max averages are 5088Kbytes and 5856Kbytes, respectively – this shows that the reserved portion of the buffer pool on this port is exhausted causing WRED to drop the packets for this queue.

The admin and operational PIR on the overloaded queues is 10Mb/s with CIR values of zero.

```
B:PE-1# show pools 1/1/10 access-ingress
=====
Pool Information
=====
Port                : 1/1/10
Application          : Acc-Ing          Pool Name          : default
Resv CBS             : Sum
-----
Queue-Groups
-----
-----
Utilization          State      Start-Avg    Max-Avg      Max-Prob
-----
High-Slope           Up           80%          100%         100%
Low-Slope            Up           80%          100%         100%
-----
Time Avg Factor      : 12
Pool Total           : 8448 KB
Pool Shared          : 5856 KB          Pool Resv           : 2592 KB
-----
High Slope Start Avg : 5088 KB          High slope Max Avg : 5856 KB
Low Slope Start Avg  : 5088 KB          Low slope Max Avg  : 5856 KB
-----
Pool Total In Use    : 6861 KB
Pool Shared In Use   : 5853 KB          Pool Resv In Use    : 1008 KB
WA Shared In Use     : 5853 KB
-----
Hi-Slope Drop Prob   : 100          Lo-Slope Drop Prob : 100
-----
Name                 Tap        FC-Maps      MBS          HP-Only A.PIR   A.CIR
                   CBS          Depth        O.PIR        O.CIR
-----
28->1/1/10:28->3
                   1/*         af           10176        0           10000        0
                   1008        0           10000        0
```

Show Output

28->1/1/10:28->1	1/*	be l2 l1 h2	1224	144	1000000	0
		h1 nc	0	0	Max	0
28->1/1/10:28->11	MCast	be l2 af l1	1224	144	1000000	0
		h2 ef h1 nc	0	0	Max	0
1->1/1/10:1->1	1/*	be l2 l1 h2	1224	144	1000000	0
		h1 nc	0	0	Max	0
1->1/1/10:1->3	1/*	af	10176	0	10000	0
			1008	6858	10000	0
1->1/1/10:1->2	1/*	ef	1224	144	1000000	0
			0	0	Max	0
28->1/1/10:28->2	1/*	ef	1224	144	1000000	0
			0	0	Max	0
=====						
B:PE-1#						

Conclusion

This note has described the basic QoS functionality available on the Alcatel-Lucent 7x50 platforms, specifically focused on the FP2 chipset. This comprises of the use of queues to shape traffic at the ingress and egress of the system and the classification, buffering, scheduling and remarking of traffic on both access, network and hybrid ports.

Glossary

6PE — IPv6 Provider Edge router. An MPLS IPv4 core network that supports IPv6 domains which communicate over an IES service.

6VPE — IPv6 Provider Edge router with IP-VPN Services. An MPLS IPv4 core network that supports the communication using IPv6 VPRN services.

AA — Application Assurance

AARP — Application Assurance Redundancy Protocol

ARP — Address Resolution Protocol

Bridged CO — Bridged Central Office

B-DA — Backbone destination MAC address

BFD — Bi-Directional Forwarding Detection

BGP — Border Gateway Protocol

BITS — Building Integrated Timing Supply

B-MAC — The backbone source and destination MAC address fields defined in the 802.1ah provider MAC encapsulation header

BMCA — Best Master Clock Algorithm

BNG — Broadband Network Gateway

BSA — Broadband Service Aggregator

BSAN — Broadband Service Access Node

BSM — Basic Subscriber Management

BSR — Broadband Service Router

BTV — Broadcast TV

B-VPLS — Backbone VPLS

CBS — Committed Burst Size

CCM — Continuity Check Messages

CE — Customer premises equipment dedicated to one particular business/enterprise.

CHAP — Challenge-Handshake Authentication Protocol

CIR — Committed Information Rate

C-MAC — Customer MAC

CO — Central Office

CSC-CE — Peering router managed and operated by the Customer Carrier that is connected to CSC-PEs for purposes of using the associated CSC IP VPN services for backbone transport. The CSC-CE may attach directly to CEs if it is also configured to be a PE for business VPN services.

CSC-PE — A PE router managed and operated by the Super Carrier that supports one or more CSC IP VPN services possibly in addition to other traditional PE services.

CSPF — Constraint-Based Shortest Path First

DE — Discard-Eligible

DEI — Drop Eligibility Indicator

DHCP — Dynamic Host Configuration Protocol

DHCPv6 — Dynamic Host Configuration Protocol for IPv6

DSCP — Differentiated Services Code Point

eBGP — External Border Gateway Protocol

Epipe — Ethernet P2P VLL Service

eLER — Egress Label Edge Router

ERO — Explicit Router Object

ESM — Enhanced Subscriber Management

ESMC — Ethernet Synchronization Messaging Channel

FEC — Forwarding Equivalence Class

FPP — Floor Packet Percentage

FRR — Fast Reroute

GMPLS — Generalized Multi-Protocol Label Switching

GNSS — Global Navigation Satellite System

GPS — Global Positioning System (the GNSS operated by the US Government)

iBGP — Interior Border Gateway Protocol

IA-NA — Identity Association for Non-Temporary Addresses Option

IA-PD — Identity Association for Prefix Delegation Option

ICB — Inter-Chassis Backup

ICMP — Internet Control Message Protocol

IES — Internet Enhanced Service

IGP — Interior Gateway Protocol

iLER — Ingress Label Edge Router

IMPM — Ingress Multicast Path Management

IPoE — IP over Ethernet

ITU-T — International Telecommunications Union - Telecommunications

I-VPLS — A field of the backbone service instance tag that identifies the backbone service instance of a frame

LDP FRR — Label Distribution Protocol Fast Re-Route

LDPoRSVP — LDP over RSVP

LSA — Link State Advertisement

LSP — Label Switched Path

LUDB — Local User Data Base

MBB — Make-Before-Break

MBS — Maximum Burst Size

MC-APS — Multi-chassis Automatic Protection Switching

MC-EP — Multi-Chassis Endpoint

MC-LAG — Multi-Chassis Link Aggregation

MDT-SAFI — Multicast Distribution Tree Sub-Address Family Indicator

MEP — Maintenance End Point

MMRP — Multiple MAC Registration Protocol

MP-BGP — Multi-Protocol BGP

MSAP — Managed Service Access Point

MTSO — Mobile Telephony Switching Office

MTU — Multi-Tenant Unit

MVPN — Multicast Virtual Private Network

NAT — Network Address Translation

NH — Next-Hop

NLHI — Network Layer Reachability Information

NTP — Network Time Protocol

OAM — Operation, Administration and Management

OIL — Outgoing Interface List

OPS — On-Path Support

P2MP — Point-to-Multipoint

PAP — Password authentication protocol

PBB — Provider Backbone Bridging

PBB-Epipe — A combination of the best of the PBB and Epipe

PBB-VPLS — A combination of the best of the PBB and VPLS

PBT — Port Based Timestamping

PDV — Packet Delay Variation

PE — Edge router managed and operated by the “Customer Carrier” that connects to CEs to provide business VPN or Internet services.

PIR — Peak Information Rate

PLR — Point of Local Repair

POP — Points of Presence

PPPoE — Point-to-Point Protocol over Ethernet

PPS — Packets per second

PTP — Precision Time Protocol

QoS — Quality of Service

RADIUS — Remote Authentication Dial In User Service

Routed CO — Routed Central Office

RPF — Reverse Path Forwarding

RR — Route Reflector

RTM — Routing Table Manager

SAP — Service Access Point

SASE — Stand Alone Synchronization Equipment

SDH — Synchronous Digital Hierarchy

SDP — Service Distribution Point

SHCV — Subscriber Host Connectivity Verification

SLA — Service Level Agreement

SONET — Synchronous Optical Network

Spoke-SDP — Spoke Service Distribution Point

SRLG — Shared Risk Link Groups

SSM — Synchronization Status Messages

TCN — Topology Change Notification

TE — Traffic Engineering

TED — Traffic Engineering Database

T-LDP — Targeted LDP

TPSDA — Triple Play Service Delivery Architecture

TTM — Tunnel Table Manager

VLL — Virtual Leased Line

VPLS — Virtual Private LAN Service

VPN-IPv6 — Virtual Private Network address family for IPv6 addressing.

VPRN — Virtual Private Routed Network

VSA — Vendor Specific Attribute

WRED — Weighted Random Early Detection

Customer documentation and product support



Customer documentation

<http://documentation.alcatel-lucent.com>



Technical support

<http://support.alcatel-lucent.com>



Documentation feedback

documentation.feedback@alcatel-lucent.com

