

HP MSR2000/3000/4000 Router Series

MPLS

Configuration Guide (V7)

Part number: 5998-3994

Software version: CMW710-R0007P02

Document version: 6PW100-20130927



Legal and notice information

© Copyright 2013 Hewlett-Packard Development Company, L.P.

No part of this documentation may be reproduced or transmitted in any form or by any means without prior written consent of Hewlett-Packard Development Company, L.P.

The information contained herein is subject to change without notice.

HEWLETT-PACKARD COMPANY MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Contents

Configuring basic MPLS	1
Overview	1
Basic concepts	1
MPLS network architecture	2
LSP establishment	3
MPLS forwarding	4
PHP	4
Protocols and standards	5
MPLS configuration task list	5
Enabling MPLS	5
Configuring MPLS MTU	6
Specifying the label type advertised by the egress	6
Configuring TTL propagation	7
Enabling sending of MPLS TTL-expired messages	9
Enabling SNMP notifications for MPLS	9
Displaying and maintaining MPLS	9
Configuring a static LSP	11
Overview	11
Configuration prerequisites	11
Configuration procedure	11
Displaying static LSPs	12
Static LSP configuration example	12
Configuring LDP	15
Overview	15
Terminology	15
LDP messages	15
LDP operation	16
Label distribution and control	17
LDP GR	19
Protocols	20
LDP configuration task list	20
Enabling LDP	21
Enabling LDP globally	21
Enabling LDP on an interface	21
Configuring Link Hello parameters	21
Configuring LDP session parameters	22
Configuring LDP backoff	22
Configuring LDP MD5 authentication	23
Configuring an LSP generation policy	23
Configuring the LDP label distribution control mode	24
Configuring a label advertisement policy	24
Configuring a label acceptance policy	25
Configuring LDP loop detection	26
Configuring LDP GR	27
Resetting LDP sessions	27
Enabling SNMP notifications for LDP	28
Displaying and maintaining LDP	28
LDP configuration examples	28

LDP LSP configuration example.....	28
Label acceptance control configuration example.....	33
Label advertisement control configuration example.....	37
Configuring MPLS TE.....	43
Overview.....	43
TE and MPLS TE.....	43
MPLS TE basic concepts.....	43
Static CRLSP establishment.....	43
Dynamic CRLSP establishment.....	43
Traffic forwarding.....	45
Make-before-break.....	46
Route pinning.....	46
Tunnel reoptimization.....	47
CRLSP backup.....	47
FRR.....	47
DiffServ-aware TE.....	48
Bidirectional MPLS TE tunnel.....	50
Protocols and standards.....	51
MPLS TE configuration task list.....	51
Enabling MPLS TE.....	52
Configuring a tunnel interface.....	53
Configuring DS-TE.....	53
Configuring an MPLS TE tunnel to use a static CRLSP.....	54
Configuring an MPLS TE tunnel to use a dynamic CRLSP.....	54
Configuration task list.....	55
Configuring MPLS TE attributes for a link.....	55
Configuring MPLS TE tunnel constraints.....	56
Establishing an MPLS TE tunnel by using RSVP-TE.....	58
Controlling CRLSP path selection.....	58
Controlling MPLS TE tunnel setup.....	60
Configuring traffic forwarding.....	62
Configuring static routing to direct traffic to an MPLS TE tunnel.....	62
Configuring PBR to direct traffic to an MPLS TE tunnel.....	62
Configuring a bidirectional MPLS TE tunnel.....	63
Configuring CRLSP backup.....	64
Configuring MPLS TE FRR.....	64
Enabling FRR.....	65
Configuring a bypass tunnel on the PLR.....	65
Configuring node fault detection.....	66
Configuring the optimal bypass tunnel selection interval.....	66
Displaying and maintaining MPLS TE.....	67
MPLS TE configuration examples.....	67
Establishing an MPLS TE tunnel over a static CRLSP.....	67
Establishing an inter-AS MPLS TE tunnel with RSVP-TE.....	72
FRR configuration example.....	78
Configuring a static CRLSP.....	84
Overview.....	84
Configuration procedure.....	84
Displaying static CRLSPs.....	85
Static CRLSP configuration example.....	85
Configuring RSVP.....	90
Overview.....	90
RSVP messages.....	90

CRLSP setup procedure	91
RSVP refresh mechanism	91
RSVP authentication	92
RSVP GR	92
Protocols and standards	93
RSVP configuration task list	93
Enabling RSVP	93
Configuring RSVP refresh	94
Configuring RSVP Srefresh and reliable RSVP message delivery	94
Configuring RSVP hello extension	95
Configuring RSVP authentication	95
Configuring RSVP GR	97
Enabling BFD for RSVP	97
Displaying and maintaining RSVP	98
Configuring tunnel policies	99
Overview	99
Configuring a tunnel policy	99
Configuration guidelines	99
Configuration procedure	100
Displaying tunnel information	101
Preferred tunnel configuration example	101
Exclusive tunnel configuration example	101
Tunnel selection order configuration example	102
Preferred tunnel and tunnel selection order configuration example	103
Configuring MPLS L3VPN	105
Overview	105
Basic MPLS L3VPN architecture	105
MPLS L3VPN concepts	106
MPLS L3VPN route advertisement	107
MPLS L3VPN packet forwarding	108
MPLS L3VPN networking schemes	109
Inter-AS VPN	111
Carrier's carrier	114
Nested VPN	116
HoVPN	118
OSPF VPN extension	120
BGP AS number substitution	122
Multi-VPN-instance CE	123
MPLS L3VPN configuration task list	124
Configuring basic MPLS L3VPN	125
Configuration prerequisites	125
Configuring VPN instances	125
Configuring routing between a PE and a CE	127
Configuring routing between PEs	133
Configuring BGP VPNv4 route control	133
Configuring inter-AS VPN	135
Configuring inter-AS option A	135
Configuring inter-AS option B	136
Configuring inter-AS option C	136
Configuring nested VPN	138
Configuring HoVPN	139
Configuring an OSPF sham link	140
Configuring a loopback interface	140

Redistributing the loopback interface route	141
Creating a sham link	141
Configuring routing on an MCE	142
Configuring routing between an MCE and a VPN site	142
Configuring routing between an MCE and a PE	147
Specifying the VPN label processing mode on the egress PE	151
Configuring BGP AS number substitution	151
Enabling SNMP notifications for MPLS L3VPN	152
Displaying and maintaining MPLS L3VPN	152
MPLS L3VPN configuration examples	154
Configuring basic MPLS L3VPN	154
Configuring MPLS L3VPN over a GRE tunnel	161
Configuring MPLS L3VPN inter-AS option A	165
Configuring MPLS L3VPN inter-AS option B	170
Configuring MPLS L3VPN inter-AS option C	175
Configuring MPLS L3VPN carrier's carrier	181
Configuring nested VPN	188
Configuring HoVPN	198
Configuring OSPF sham links	205
Configuring MCE	209
Configuring BGP AS number substitution	214
Configuring IPv6 MPLS L3VPN	218
Overview	218
IPv6 MPLS L3VPN packet forwarding	218
IPv6 MPLS L3VPN routing information advertisement	219
IPv6 MPLS L3VPN network schemes and functions	219
IPv6 MPLS L3VPN configuration task list	220
Configuring basic IPv6 MPLS L3VPN	220
Configuring VPN instances	220
Configuring routing between a PE and a CE	223
Configuring routing between PEs	227
Configuring BGP VPNv6 route control	228
Configuring inter-AS IPv6 VPN	229
Configuring inter-AS IPv6 VPN option A	229
Configuring inter-AS IPv6 VPN option C	230
Configuring routing on an MCE	231
Configuring routing between an MCE and a VPN site	231
Configuring routing between an MCE and a PE	236
Displaying and maintaining IPv6 MPLS L3VPN	239
IPv6 MPLS L3VPN configuration examples	241
Configuring IPv6 MPLS L3VPNs	241
Configuring an IPv6 MPLS L3VPN over a GRE tunnel	248
Configuring IPv6 MPLS L3VPN inter-AS option A	252
Configuring IPv6 MPLS L3VPN inter-AS option C	257
Configuring IPv6 MPLS L3VPN carrier's carrier	263
Configuring IPv6 MCE	270
Configuring MPLS OAM	277
Overview	277
MPLS ping	277
MPLS traceroute	277
Periodic MPLS traceroute	277
MPLS BFD	277
Protocols and standards	278

Configuring MPLS OAM for LSP tunnels.....	278
Configuring MPLS ping for LSPs	278
Configuring MPLS traceroute for LSPs	278
Configuring periodic MPLS traceroute for LSPs.....	279
Configuring MPLS BFD for LSPs	279
Configuring MPLS OAM for MPLS TE tunnels.....	280
Displaying MPLS OAM.....	281
BFD for LSP configuration example.....	281
Support and other resources	284
Contacting HP	284
Subscription service	284
Related information.....	284
Documents.....	284
Websites.....	284
Conventions	285
Index	287

Configuring basic MPLS

Multiprotocol Label Switching (MPLS) provides connection-oriented label switching over connectionless IP backbone networks. It integrates both the flexibility of IP routing and the simplicity of Layer 2 switching.

Overview

MPLS has the following advantages:

- **High speed and efficiency**—MPLS uses short- and fixed-length labels to forward packets, avoiding complicated routing table lookups.
- **Multiprotocol support**—MPLS resides between the link layer and the network layer. It can work over various link layer protocols (for example, PPP, ATM, frame relay, and Ethernet) to provide connection-oriented services for various network layer protocols (for example, IPv4, IPv6, and IPX).
- **Good scalability**—The connection-oriented switching and multi-layer label stack features enable MPLS to deliver various extended services, such as VPN, traffic engineering, and QoS.

Basic concepts

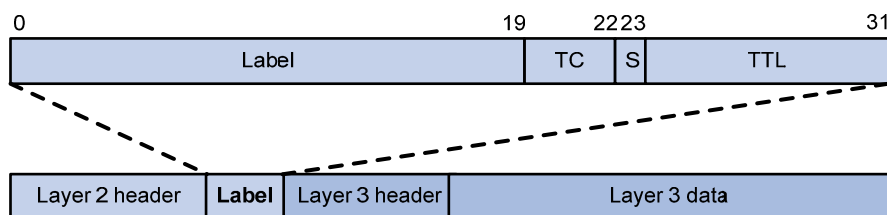
FEC

MPLS groups packets with the same characteristics (such as packets with the same destination or service class) into a class, called a "forwarding equivalence class (FEC)." Packets of the same FEC are handled in the same way on an MPLS network.

Label

A label uniquely identifies a FEC and has local significance.

Figure 1 Format of a label



A label is encapsulated between the Layer 2 header and Layer 3 header of a packet. It is four bytes long and consists of the following fields:

- **Label**—20-bit label value.
- **TC**—3-bit traffic class, used for QoS. It is also called "Exp."
- **S**—1-bit bottom of stack flag. A label stack can comprise multiple labels. The label nearest to the Layer 2 header is called the "top label," and the label nearest to the Layer 3 header is called the "bottom label." The **S** field is set to 1 if the label is the bottom label and set to 0 if not.
- **TTL**—8-bit time to live field used for routing loop prevention.

LSR

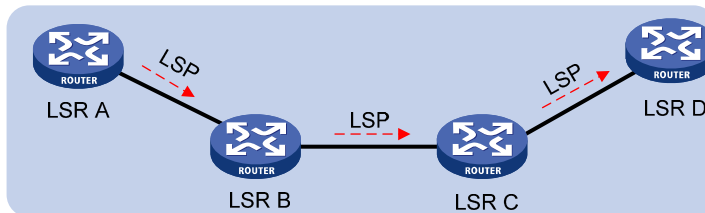
A router that performs MPLS forwarding is a label switching router (LSR).

LSP

A label switched path (LSP) is the path along which packets of a FEC travel through an MPLS network.

An LSP is a unidirectional packet forwarding path. Two neighboring LSRs are called the "upstream LSR" and "downstream LSR" along the direction of an LSP. In [Figure 2](#), LSR B is the downstream LSR of LSR A, and LSR A is the upstream LSR of LSR B.

Figure 2 Label switched path



LFIB

The Label Forwarding Information Base (LFIB) on an MPLS network functions like the Forwarding Information Base (FIB) on an IP network. When an LSR receives a labeled packet, it searches the LFIB to obtain information for forwarding the packet, such as the label operation type, the outgoing label value, and the next hop.

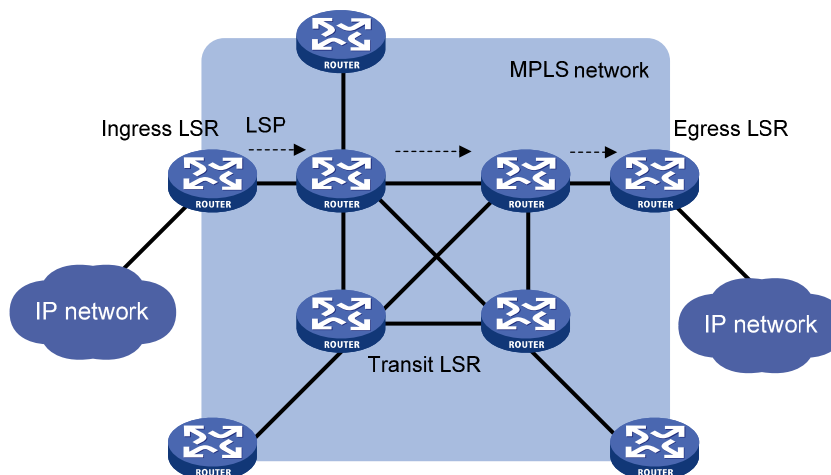
Control plane and forwarding plane

An MPLS node consists of a control plane and a forwarding plane.

- **Control plane**—Assigns labels, distributes FEC-label mappings to neighbor LSRs, creates the LFIB, and establishes and removes LSPs.
- **Forwarding plane**—Forwards packets according to the LFIB.

MPLS network architecture

Figure 3 MPLS network architecture



An MPLS network comprises the following types of LSRs:

- **Ingress LSR**—Ingress LSR of packets. It labels packets entering into the MPLS network.
- **Transit LSR**—Intermediate LSRs in the MPLS network. The transit LSRs on an LSP forward packets to the egress LSR according to labels.
- **Egress LSR**—Egress LSR of packets. It removes labels from packets and forwards the packets to their destination networks.

LSP establishment

LSPs include static and dynamic LSPs.

- **Static LSP**—To establish a static LSP, you must configure an LFIB entry on each LSR along the LSP. Establishing static LSPs consumes fewer resources than establishing dynamic LSPs, but static LSPs cannot automatically adapt to network topology changes. Therefore, static LSPs are suitable for small-scale networks with simple, stable topologies.
- **Dynamic LSP**—Established by a label distribution protocol (also called an MPLS signaling protocol). A label distribution protocol classifies FECs, distributes FEC-label mappings, and establishes and maintains LSPs. Label distribution protocols include protocols designed specifically for label distribution, such as the Label Distribution Protocol (LDP), and protocols extended to support label distribution, such as MP-BGP and RSVP-TE.

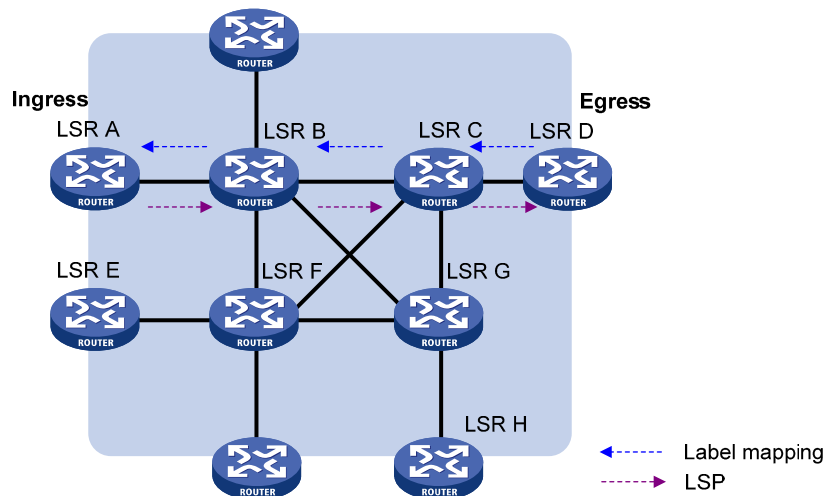
In this document, the term "label distribution protocols" refers to all protocols for label distribution. The term "LDP" refers to the RFC 5036 LDP.

A dynamic LSP is established in the following steps:

1. A downstream LSR classifies FECs according to destination addresses.
2. The downstream LSR assigns a label for each FEC, and distributes the FEC-label binding to its upstream LSR.
3. The upstream LSR establishes an LFIB entry for the FEC according to the binding information.

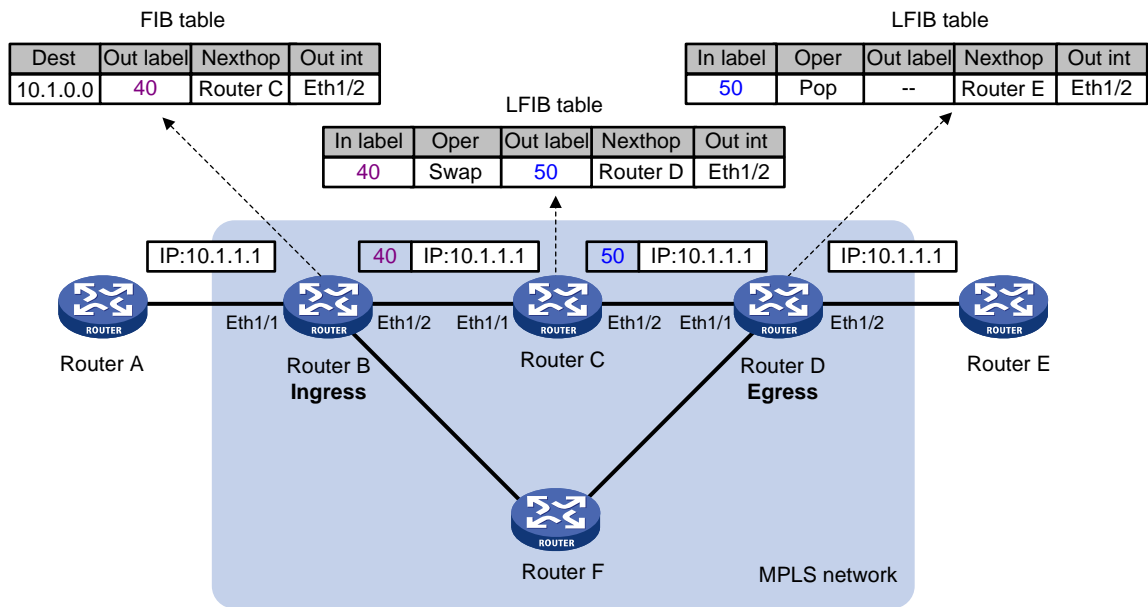
After all LSRs along the LSP establish an LFIB entry for the FEC, a dynamic LSP is established for the packets of this FEC.

Figure 4 Dynamic LSP establishment



MPLS forwarding

Figure 5 MPLS forwarding



As shown in Figure 5, a packet is forwarded over the MPLS network in the following steps:

1. Router B (the ingress LSR) receives a packet with no label. It identifies the FIB entry that matches the destination address of the packet, pushes the outgoing label (40 in this example) to the packet, and forwards the labeled packet out of the interface Ethernet 1/2 to the next hop LSR Router C.
2. When receiving the labeled packet, Router C identifies the LFIB entry that has an incoming label of 40, uses the outgoing label 50 of the entry to replace label 40 in the packet, and forwards the labeled packet out of the outgoing interface Ethernet 1/2 to the next hop LSR Router D.
3. When receiving the labeled packet, Router D (the egress) identifies the LFIB entry that has an incoming label of 50, removes the label from the packet, and forwards the packet out of the outgoing interface Ethernet 1/2 to the next hop LSR Router E. If the LFIB entry records no outgoing interface or next hop information, Router D identifies the FIB entry by the IP header and then forwards the packet according to the FIB entry.

PHP

An egress node must perform two forwarding table lookups to forward a packet: two LFIB lookups (if the packet has more than one label), or one LFIB lookup and one FIB lookup (if the packet has only one label).

The penultimate hop popping (PHP) feature can pop the label at the penultimate node, so the egress node only performs one table lookup.

A PHP-capable egress node sends the penultimate node an implicit null label of 3. This label never appears in the label stack of packets. If an incoming packet matches an LFIB entry comprising the implicit null label, the penultimate node pops the top label of the packet and forwards the packet to the egress LSR. The egress LSR directly forwards the packet.

Sometimes, the egress node must use the TC field in the label to perform QoS. To keep the TC information, you can configure the egress node to send the penultimate node an explicit null label of 0. If an incoming

packet matches an LFIB entry comprising the explicit null label, the penultimate hop replaces the value of the top label with value 0, and forwards the packet to the egress node. The egress node gets the TC information, pops the label of the packet, and forwards the packet.

Protocols and standards

- RFC 3031, *Multiprotocol Label Switching Architecture*
- RFC 3032, *MPLS Label Stack Encoding*
- RFC 5462, *Multiprotocol Label Switching (MPLS) Label Stack Entry: "EXP" Field Renamed to "Traffic Class" Field*

MPLS configuration task list

Tasks at a glance
(Required.) Enabling MPLS
(Optional.) Configuring MPLS MTU
(Optional.) Specifying the label type advertised by the egress
(Optional.) Configuring TTL propagation
(Optional.) Enabling sending of MPLS TTL-expired messages
(Optional.) Enabling SNMP notifications for MPLS

Enabling MPLS

You must enable MPLS on all interfaces related to MPLS forwarding.

Before you enable MPLS, complete the following tasks:

- Configure link layer protocols to ensure connectivity at the link layer.
- Configure IP addresses for interfaces to ensure IP connectivity between neighboring nodes. Configure static routes or an IGP protocol to ensure IP connectivity among LSRs. To enable MPLS:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure an LSR ID for the local node.	mpls lsr-id <i>lsr-id</i>	By default, no LSR ID is configured. An LSR ID must be unique in an MPLS network and in IP address format. HP recommends using the IP address of a loopback interface as an LSR ID.
3. Enter the view of the interface that needs to perform MPLS forwarding.	interface <i>interface-type</i> <i>interface-number</i>	N/A
4. Enable MPLS for the interface.	mpls enable	By default, MPLS is disabled on an interface.

Configuring MPLS MTU

MPLS inserts the label stack between the link layer header and network layer header of each packet. To make sure the size of MPLS labeled packets is smaller than the MTU of an interface, configure an MPLS MTU on the interface.

MPLS compares each MPLS packet against the interface MPLS MTU. When the packet exceeds the MPLS MTU:

- If fragmentation is allowed, MPLS removes the label stack from the packet, fragments the IP packet (the length of a fragment is the MPLS MTU minus the length of the label stack), adds the label stack to each fragment, and forwards the fragments.
- If fragmentation is not allowed, the LSR drops the packet.

To configure an MPLS MTU for an interface:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Configure an MPLS MTU for the interface.	mpls mtu <i>value</i>	By default, no MPLS MTU is configured on an interface.

The following applies when an interface handles MPLS packets:

- MPLS packets carrying IPv6 packets are always forwarded by an interface, even if the length of the MPLS packets exceeds the MPLS MTU of the interface. Whether the forwarding can succeed depends on the actual forwarding capacity of the interface.
- If the MPLS MTU of an interface is greater than the MTU of the interface, data forwarding might fail on the interface.
- If you do not configure the MPLS MTU of an interface, fragmentation of MPLS packets is based on the MTU of the interface without considering MPLS labels. An MPLS fragment might be larger than the interface MTU and be dropped.

Specifying the label type advertised by the egress

In an MPLS network, an egress can advertise the following types of labels:

- Implicit null label with a value of 3.
- Explicit null label with a value of 0.
- Non-null label.

For LSPs established by a label distribution protocol, the label advertised by the egress determines how the penultimate hop processes a labeled packet.

- If the egress advertises an implicit null label, the penultimate hop directly pops the top label of a matching packet.
- If the egress advertises an explicit null label, the penultimate hop swaps the top label value of a matching packet with the explicit null label.

- If the egress advertises a non-null label (normal label), the penultimate hop swaps the top label of a matching packet with the specific label assigned by the egress.

Configuration guidelines

If the penultimate hop supports PHP, HP recommends that you configure the egress to advertise an implicit null label to the penultimate hop. If you want to simplify packet forwarding on the egress but keep labels in packets for the egress to determine QoS policies, you can configure the egress to advertise an explicit null label to the penultimate hop. HP recommends that you do not use non-null labels except in some special scenarios. For example, when OAM is configured on the egress, the egress can get the OAM function entity status only through non-null labels.

As a penultimate hop, the device accepts the implicit null label, explicit null label, or normal label advertised by the egress device.

For LDP LSPs, the **mpls label advertise** command triggers LDP to delete the LSPs established before the command is executed and re-establishes new LSPs.

For BGP LSPs, the **mpls label advertise** command takes effect only for the BGP LSPs established after the command is executed. To apply the new setting to BGP LSPs established before the command is executed, delete the routes corresponding to the BGP LSPs, and then redistribute the routes.

Configuration procedure

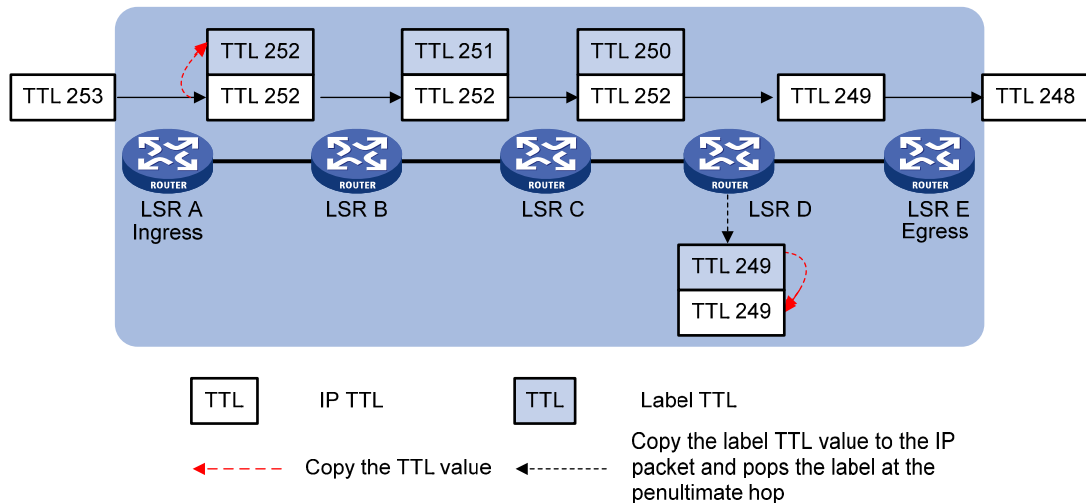
To specify the type of label that the egress node will advertise to the penultimate hop:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Specify the label type advertised by the egress to the penultimate hop.	mpls label advertise { explicit-null implicit-null non-null }	By default, an egress advertises an implicit null label to the penultimate hop.

Configuring TTL propagation

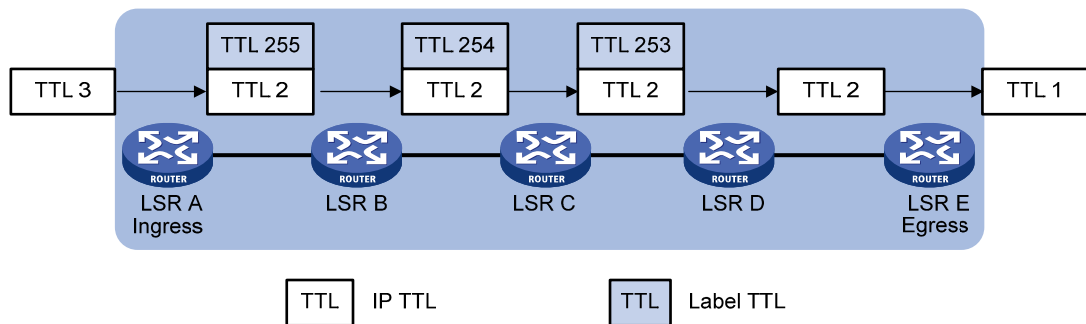
When TTL propagation is enabled, the ingress node copies the TTL value of an IP packet to the TTL field of the label. Each LSR on the LSP decreases the label TTL value by 1. The LSR that pops the label copies the remaining label TTL value back to the IP TTL of the packet, so the IP TTL value can reflect how many hops the packet has traversed in the MPLS network. The IP tracert facility can show the real path along which the packet has traveled.

Figure 6 TTL propagation



When TTL propagation is disabled, the ingress node sets the label TTL to 255. Each LSR on the LSP decreases the label TTL value by 1. The LSR that pops the label does not change the IP TTL value when popping the label. Therefore, the MPLS backbone nodes are invisible to user networks, and the IP traceroute facility cannot show the real path in the MPLS network.

Figure 7 Without TTL propagation



Follow these guidelines when you configure TTL propagation:

- HP recommends setting the same TTL processing mode on all LSRs of an LSP.
- To enable TTL propagation for a VPN, you must enable it on all PE devices in the VPN, so that you can get the same traceroute result (hop count) from those PEs.

To enable TTL propagation:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable TTL propagation.	mpls ttl propagate { public vpn }	By default, TTL propagation is enabled only for public-network packets. This command affects only the propagation between IP TTL and label TTL. Within an MPLS network, TTL is always copied between the labels of an MPLS packet.

Enabling sending of MPLS TTL-expired messages

This feature enables an LSR to generate an ICMP TTL-expired message upon receiving an MPLS packet with a TTL of 1. If the MPLS packet has only one label, the LSR sends the ICMP TTL-expired message back to the source through IP routing. If the MPLS packet has multiple labels, the LSR sends it along the LSP to the egress, which then sends the message back to the source.

To enable sending of MPLS TTL-expired messages:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable sending of MPLS TTL-expired messages.	mpls ttl expiration enable	By default, this function is enabled.

Enabling SNMP notifications for MPLS

This feature enables MPLS to generate SNMP notifications. The generated SNMP notifications are sent to the SNMP module.

For more information about SNMP notifications, see *Network Management and Monitoring Configuration Guide*.

To enable SNMP notifications for MPLS:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable SNMP notifications for MPLS.	snmp-agent trap enable mpls	By default, SNMP notifications for MPLS are enabled.

Displaying and maintaining MPLS

Execute **display** commands in any view.

Task	Command
Display MPLS interface information.	display mpls interface [<i>interface-type interface-number</i>]
Display usage information about MPLS labels.	display mpls label { <i>label-value1</i> [to <i>label-value2</i>] all }
Display LSP information.	display mpls lsp [egress in-label <i>label-value</i> ingress outgoing-interface <i>interface-type interface-number</i> protocol { bgp ldp local rsvp-te static static-cr } transit] [vpn-instance <i>vpn-instance-name</i>] [<i>ipv4-dest mask-length</i> ipv6 [<i>ipv6-dest prefix-length</i>]] [verbose]
Display MPLS Nexthop Information Base (NIB) information.	display mpls nib [<i>nib-id</i>]
Display usage information about NIDs.	display mpls nid [<i>nid-value1</i> [to <i>nid-value2</i>]]
Display MPLS summary information.	display mpls summary

Task	Command
Display ILM entries (MSR2000/MSR3000).	display mpls forwarding ilm [<i>label</i>]
Display ILM entries (MSR4000).	display mpls forwarding ilm [<i>label</i>] slot <i>slot-number</i>
Display NHLFE entries (MSR2000/MSR3000).	display mpls forwarding nhlfe [<i>nid</i>]
Display NHLFE entries (MSR4000).	display mpls forwarding nhlfe [<i>nid</i>] slot <i>slot-number</i>

Configuring a static LSP

Overview

A static label switched path (LSP) is established by manually specifying the incoming label and outgoing label on each node (ingress, transit, or egress node) of the forwarding path.

Static LSPs consume fewer resources, but they cannot automatically adapt to network topology changes. Therefore, static LSPs are suitable for small and stable networks with simple topologies.

Follow these guidelines to establish a static LSP:

- The ingress node determines an FEC for a packet according to the destination address, inserts the label for that FEC into the packet, and forwards the packet to the next hop or out of the outgoing interface. Therefore, on the ingress node, you must specify the outgoing label for the destination address (the FEC) and the next hop or the outgoing interface.
- A transit node swaps the label carried in a received packet with a specific label, and forwards the packet to the next hop or out of the outgoing interface. Therefore, on each transit node, you must specify the incoming label, the outgoing label, and the next hop or the outgoing interface.
- If the penultimate hop popping function is not configured, an egress node pops the incoming label of a packet, and performs label forwarding according to the inner label or IP forwarding. Therefore, on the egress node, you only need to specify the incoming label.
- The outgoing label specified on an LSR must be the same as the incoming label specified on the directly-connected downstream LSR.

Configuration prerequisites

Before you configure a static LSP, complete the following tasks:

- Identify the ingress node, transit nodes, and egress node of the LSP.
- Enable MPLS on all interfaces that participate in MPLS forwarding. For more information, see "Configuring basic MPLS."
- Make sure the ingress node has a route to the destination address of the LSP. This is not required on transit and egress nodes.

Configuration procedure

To configure a static LSP:

Step	Command	Remarks
1. Enter system view.	system-view	N/A

Step	Command	Remarks
2. Configure the ingress node of the static LSP.	static-lsp ingress <i>lsp-name</i> destination <i>dest-addr { mask mask-length }</i> { nexthop <i>next-hop-addr</i> outgoing-interface <i>interface-type interface-number</i> } out-label <i>out-label</i>	If you specify a next hop for the static LSP, make sure the ingress node has an active route to the specified next hop address.
3. Configure the transit node of the static LSP.	static-lsp transit <i>lsp-name</i> in-label <i>in-label</i> { nexthop <i>next-hop-addr</i> outgoing-interface <i>interface-type interface-number</i> } out-label <i>out-label</i>	If you specify a next hop for the static LSP, make sure the transit node has an active route to the specified next hop address.
4. Configure the egress node of the static LSP.	static-lsp egress <i>lsp-name</i> in-label <i>in-label</i>	You do not need to configure this command if the outgoing label configured on the penultimate hop of the static LSP is 0 or 3.

Displaying static LSPs

Execute **display** commands in any view.

Task	Command
Display static LSP information.	display mpls static-lsp [<i>lsp-name lsp-name</i>]

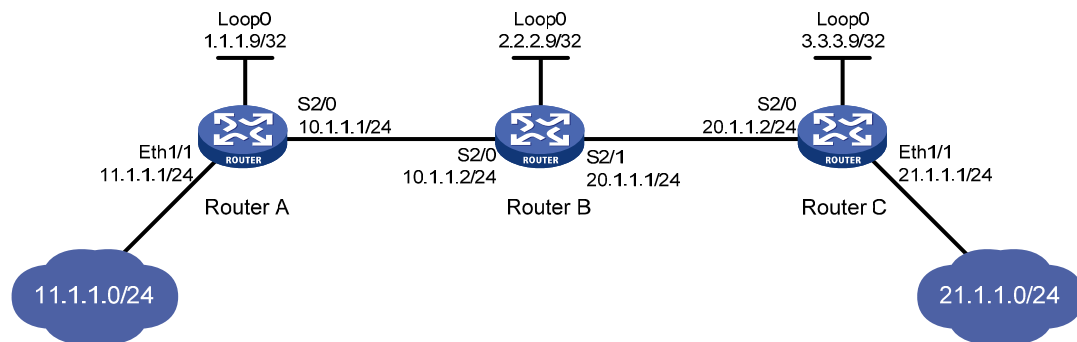
Static LSP configuration example

Network requirements

Router A, Router B, and Router C all support MPLS.

Establish static LSPs between Router A and Router C, so that subnets 11.1.1.0/24 and 21.1.1.0/24 can access each other over MPLS.

Figure 8 Network diagram



Configuration considerations

- For an LSP, the outgoing label specified on an LSR must be identical with the incoming label specified on the downstream LSR.
- LSPs are unidirectional. You must configure an LSP for each direction of the data forwarding path.

- A route to the destination address of the LSP must be available on the ingress node, but it is not needed on transit and egress nodes. Therefore, you do not need to configure a routing protocol to ensure IP connectivity among all routers.

Configuration procedure

1. Configure IP addresses for all interfaces, including the loopback interfaces, as shown in [Figure 8](#). (Details not shown.)

2. Configure a static route to the destination address of each LSP:

On Router A, configure a static route to network 21.1.1.0/24.

```
<RouterA> system-view
[RouterA] ip route-static 21.1.1.0 24 10.1.1.2
```

On Router C, configure a static route to network 11.1.1.0/24.

```
<RouterC> system-view
[RouterC] ip route-static 11.1.1.0 255.255.255.0 20.1.1.1
```

3. Configure basic MPLS on the routers:

Configure Router A.

```
[RouterA] mpls lsr-id 1.1.1.9
[RouterA] interface serial 2/0
[RouterA-Serial2/0] mpls enable
[RouterA-Serial2/0] quit
```

Configure Router B.

```
[RouterB] mpls lsr-id 2.2.2.9
[RouterB] interface serial 2/0
[RouterB-Serial2/0] mpls enable
[RouterB-Serial2/0] quit
[RouterB] interface serial 2/1
[RouterB-Serial2/1] mpls enable
[RouterB-Serial2/1] quit
```

Configure Router C.

```
[RouterC] mpls lsr-id 3.3.3.9
[RouterC] interface serial 2/0
[RouterC-Serial2/0] mpls enable
[RouterC-Serial2/0] quit
```

4. Configure a static LSP from Router A to Router C:

Configure the LSP ingress node, Router A.

```
[RouterA] static-lsp ingress AtoC destination 21.1.1.0 24 nexthop 10.1.1.2 out-label 30
```

Configure the LSP transit node, Router B.

```
[RouterB] static-lsp transit AtoC in-label 30 nexthop 20.1.1.2 out-label 50
```

Configure the LSP egress node, Router C.

```
[RouterC] static-lsp egress AtoC in-label 50
```

5. Create a static LSP from Router C to Router A:

Configure the LSP ingress node, Router C.

```
[RouterC] static-lsp ingress CtoA destination 11.1.1.0 24 nexthop 20.1.1.1 out-label 40
```

Configure the LSP transit node, Router B.

```
[RouterB] static-lsp transit CtoA in-label 40 nexthop 10.1.1.1 out-label 70
# Configure the LSP egress node, Router A.
[RouterA] static-lsp egress CtoA in-label 70
```

Verifying the configuration

Use the **display mpls static-lsp** command on each router to view information about static LSPs. Take Router A as an example:

```
[RouterA] display mpls static-lsp
Total: 2
Name          FEC                In/Out Label Nexthop/Out Interface  State
AtoC          21.1.1.0/24        NULL/30      10.1.1.2
CtoA          -/-                70/NULL     -
```

On Router A, test the connectivity of the LSP from Router A to Router C.

```
[RouterA] ping mpls -a 11.1.1.1 ipv4 21.1.1.0 24
MPLS Ping FEC: 21.1.1.0/24 : 100 data bytes
100 bytes from 20.1.1.2: Sequence=1 time=4 ms
100 bytes from 20.1.1.2: Sequence=2 time=1 ms
100 bytes from 20.1.1.2: Sequence=3 time=1 ms
100 bytes from 20.1.1.2: Sequence=4 time=1 ms
100 bytes from 20.1.1.2: Sequence=5 time=1 ms

--- FEC: 21.1.1.0/24 ping statistics ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max = 1/1/4 ms
```

On Router C, test the connectivity of the LSP from Router C to Router A.

```
[RouterC] ping mpls -a 21.1.1.1 ipv4 11.1.1.0 24
MPLS Ping FEC: 11.1.1.0/24 : 100 data bytes
100 bytes from 10.1.1.1: Sequence=1 time=5 ms
100 bytes from 10.1.1.1: Sequence=2 time=1 ms
100 bytes from 10.1.1.1: Sequence=3 time=1 ms
100 bytes from 10.1.1.1: Sequence=4 time=1 ms
100 bytes from 10.1.1.1: Sequence=5 time=1 ms

--- FEC: 11.1.1.0/24 ping statistics ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max = 1/1/5 ms
```

Configuring LDP

Overview

The Label Distribution Protocol (LDP) dynamically distributes FEC-label mapping information between LSRs to establish LSPs.

Terminology

LDP session

Two LSRs establish a TCP-based LDP session to exchange FEC-label mappings.

LDP peer

Two LSRs that use LDP to exchange FEC-label mappings are LSR peers.

Label spaces and LDP identifiers

Label spaces include the following types:

- **Per-interface label space**—Each interface uses a single, independent label space. Different interfaces can use the same label values.
- **Per-platform label space**—Each LSR uses a single label space. The device only supports the per-platform label space.

A six-byte LDP Identifier (LDP ID) identifies a label space on an LSR. It is in the format of <LSR ID>:<label space number>, where the LSR ID takes four bytes to identify the LSR, and the label space number takes two bytes to identify a label space within the LSR. A label space number of 0 indicates that the label space is a per-platform label space. A label space number other than 0 indicates a per-interface label space.

FECs and FEC-label mappings

MPLS groups packets with the same characteristics (such as the same destination or service class) into a class, called an "FEC." The packets of the same FEC are handled in the same way on an MPLS network.

LDP can classify FECs by destination IP address.

An LSR assigns a label for a FEC and advertises the FEC-label mapping, or FEC-label binding, to its peers in a Label Mapping message.

LDP messages

LDP mainly uses the following types of messages:

- **Discovery messages**—Declare and maintain the presence of LSRs, such as Hello messages.
- **Session messages**—Establish, maintain, and terminate sessions between LDP peers, such as Initialization messages used for parameter negotiation and Keepalive messages used to maintain sessions.

- **Advertisement messages**—Create, alter, and remove FEC-label mappings, such as Label Mapping messages used to advertise FEC-label mappings.
- **Notification messages**—Provide advisory information and notify errors, such as Notification messages.

LDP uses UDP to transport discovery messages for efficiency, and uses TCP to transport session, advertisement, and notification messages for reliability.

LDP operation

LDP operates in the following phases:

Discovering and maintaining LDP peers

LDP discovers peers by sending Link Hello messages to multicast address 224.0.0.2 that identifies all routers on the subnet. In this way, all directly-connected LSRs can discover the LSR and establish a hello adjacency.

LDP sends Hello messages at the hello interval to maintain a hello adjacency. If LDP receives no Hello message from a hello adjacency before the hello hold timer expires, it removes the hello adjacency.

Establishing and maintaining LDP sessions

LDP establishes a session with a peer in the following steps:

1. Establishes a TCP connection with the neighbor.
2. Negotiates session parameters such as LDP version, label distribution method, and Keepalive timer, and establishes an LDP session with the neighbor if the negotiation succeeds.

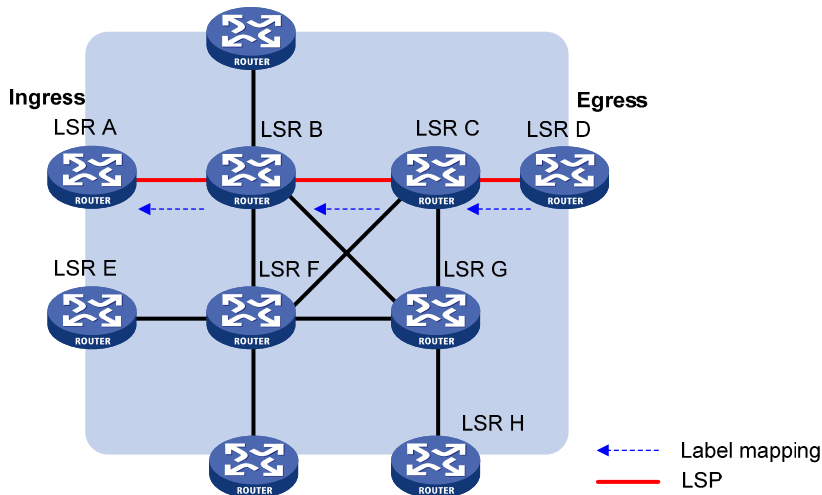
After a session is established, LDP sends LDP PDUs (an LDP PDU carries one or more LDP messages) to maintain the session. If no information is exchanged between the LDP peers within the Keepalive interval, LDP sends Keepalive messages at the Keepalive interval to maintain the session. If LDP receives no LDP PDU from a neighbor before the keepalive hold timer expires, or the last hello adjacency with the neighbor is removed, LDP terminates the session.

LDP can also send a Shutdown message to a neighbor to terminate the LDP session.

Establishing LSPs

LDP classifies FECs according to destination IP addresses in IP routing entries, creates FEC-label mappings, and advertises the mappings to LDP peers through LDP sessions. After an LDP peer receives a FEC-label mapping, it uses the received label and the label locally assigned to that FEC to create an LFIB entry for that FEC. When all LSRs (from the Ingress to the Egress) establish an LFIB entry for the FEC, an LSP is established exclusively for the FEC.

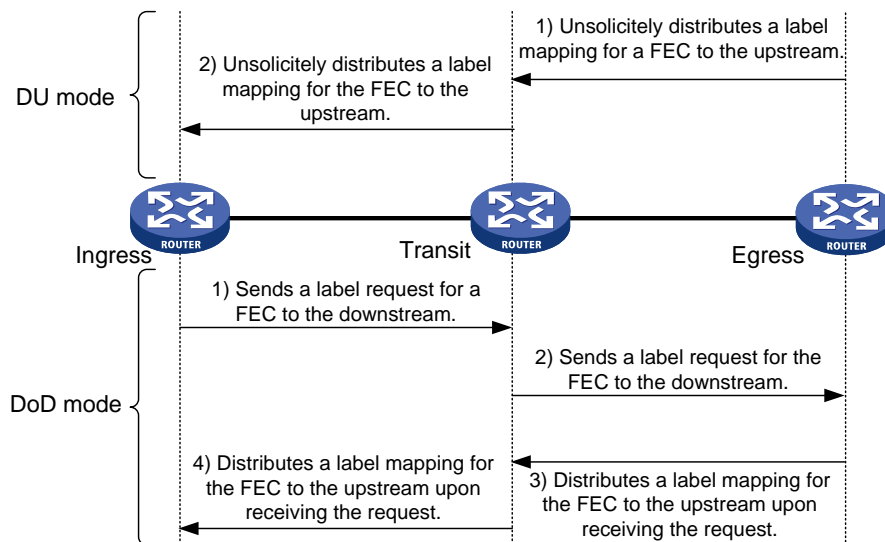
Figure 9 Dynamically establishing an LSP



Label distribution and control

Label advertisement modes

Figure 10 Label advertisement modes



LDP advertises label-FEC mappings in one of the following ways:

- **Downstream Unsolicited (DU) mode**—Unsolicitedly distributes FEC-label mappings to the upstream LSR, without waiting for label requests. The device supports only the DU mode.
- **Downstream on Demand (DoD) mode**—Sends a label request for a FEC to the downstream LSR. After receiving the label request, the downstream LSR distributes the FEC-label mapping for that FEC to the upstream LSR.

NOTE:

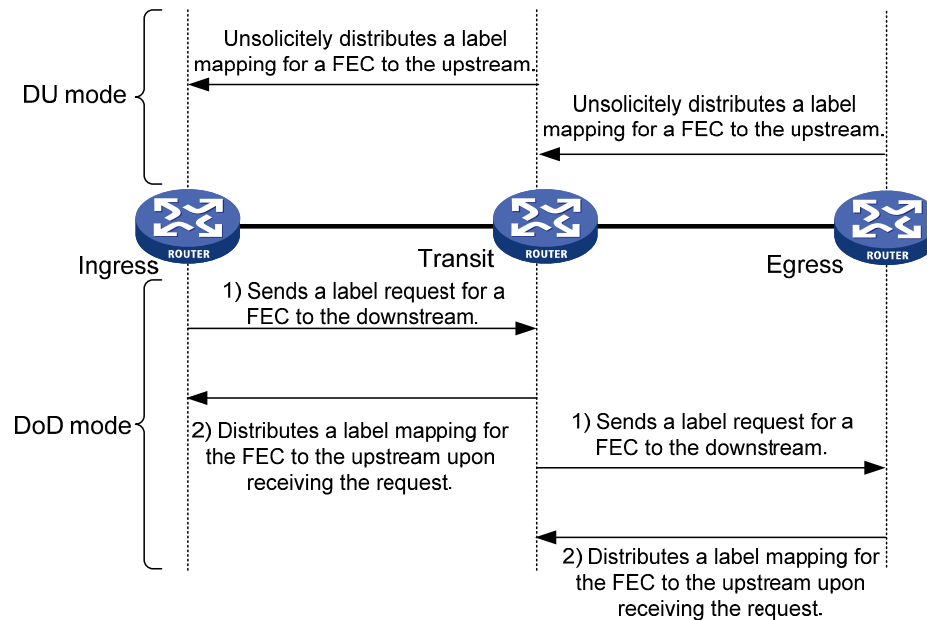
A pair of upstream and downstream LSRs must use the same label advertisement mode. Otherwise, the LSP cannot be established.

Label distribution control

LDP controls label distribution in one of the following ways:

- **Independent label distribution**—Distributes a FEC-label mapping to an upstream LSR at any time. An LSR might distribute a mapping for a FEC to its upstream LSR before it receives a label mapping for that FEC from its downstream LSR. As shown in Figure 11, in DU mode, each LSR distributes a label mapping for a FEC to its upstream LSR whenever it is ready to label-switch the FEC, without waiting for a label mapping for the FEC from its downstream LSR. In DoD mode, an LSR distributes a label mapping for a FEC to its upstream LSR after it receives a label request for the FEC, without waiting for a label mapping for the FEC from its downstream LSR.

Figure 11 Independent label distribution control mode



- **Ordered label distribution**—Distributes a label mapping for a FEC to its upstream LSR only after it receives a label mapping for that FEC from its downstream LSR unless the local node is the egress node of the FEC. As shown in Figure 10, in DU mode, an LSR distributes a label mapping for a FEC to its upstream LSR only if it receives a label mapping for the FEC from its downstream LSR. In DoD mode, when an LSR (Transit) receives a label request for a FEC from its upstream LSR (Ingress), it continues to send a label request for the FEC to its downstream LSR (Egress). After the transit LSR receives a label mapping for the FEC from the egress LSR, it distributes a label mapping for the FEC to the ingress.

Label retention mode

The label retention mode specifies whether an LSR maintains a label mapping for a FEC learned from a neighbor that is not its next hop.

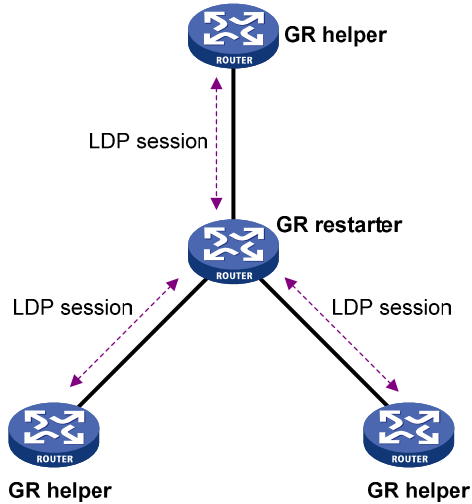
- **Liberal label retention**—Retains a received label mapping for a FEC regardless of whether the advertising LSR is the next hop of the FEC. This mechanism allows for quicker adaptation to topology changes, but it wastes system resources because LDP has to keep useless labels. The device only supports liberal label retention.
- **Conservative label retention**—Retains a received label mapping for a FEC only when the advertising LSR is the next hop of the FEC. This mechanism saves label resources, but it cannot quickly adapt to topology changes.

LDP GR

LDP GR overview

LDP Graceful Restart enables an LSR to retain MPLS forwarding entries during an LDP restart, ensuring continuous MPLS forwarding.

Figure 12 LDP GR

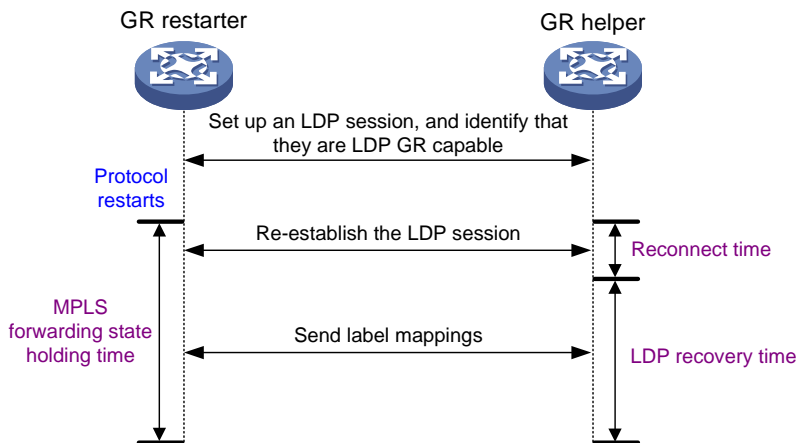


As shown in Figure 12, GR defines the following roles:

- **GR restarter**—An LSR that performs GR. It must be GR-capable.
- **GR helper**—A neighbor LSR that helps the GR restarter to complete GR.

The device can act as a GR restarter or a GR helper.

Figure 13 LDP GR operation



As shown in Figure 13, LDP GR works in the following steps:

1. LSRs establish an LDP session. The L flag of the Fault Tolerance TLV in their Initialization messages is set to 1 to indicate that they support LDP GR.
2. When LDP restarts, the GR restarter starts the MPLS Forwarding State Holding timer, and marks the MPLS forwarding entries as stale. When the GR helper detects that the LDP session with the GR

restarter goes down, it marks the FEC-label mappings learned from the session as stale and starts the Reconnect timer received from the GR restarter.

3. After LDP completes restart, the GR restarter re-establishes an LDP session with the GR helper. If the LDP session is not set up before the Reconnect timer expires, the GR helper deletes the stale FEC-label mappings and the corresponding MPLS forwarding entries. If the LDP session is successfully set up before the Reconnect timer expires, the GR restarter sends the remaining time of the MPLS Forwarding State Holding timer as the LDP Recovery time to the GR helper.
4. After the LDP session is re-established, the GR helper starts the LDP Recovery timer.
5. The GR restarter and the GR helper exchange label mappings and update their MPLS forwarding tables.

The GR restarter compares each received label mapping against stale MPLS forwarding entries. If a match is found, the restarter deletes the stale mark for the matching entry. Otherwise, it adds a new entry for the label mapping.

The GR helper compares each received label mapping against stale FEC-label mappings. If a match is found, the helper deletes the stale mark for the matching mapping. Otherwise, it adds the received FEC-label mapping and a new MPLS forwarding entry for the mapping.

6. When the MPLS Forwarding State Holding timer expires, the GR restarter deletes all stale MPLS forwarding entries.
7. When the LDP Recovery timer expires, the GR helper deletes all stale FEC-label mappings.

Protocols

RFC 5036, *LDP Specification*

LDP configuration task list

Tasks at a glance

Enable LDP:

1. (Required.) [Enabling LDP globally](#)
2. (Required.) [Enabling LDP on an interface](#)

(Optional.) [Configuring Link Hello parameters](#)

(Optional.) [Configuring LDP session parameters](#)

(Optional.) [Configuring LDP backoff](#)

(Optional.) [Configuring LDP MD5 authentication](#)

(Optional.) [Configuring an LSP generation policy](#)

(Optional.) [Configuring the LDP label distribution control mode](#)

(Optional.) [Configuring a label advertisement policy](#)

(Optional.) [Configuring a label acceptance policy](#)

(Optional.) [Configuring LDP loop detection](#)

(Optional.) [Configuring LDP GR](#)

(Optional.) [Resetting LDP sessions](#)

(Optional.) [Enabling SNMP notifications for LDP](#)

Enabling LDP

To enable LDP, you must enable LDP globally, and then enable LDP on relevant interfaces or configure IGP to automatically enable LDP on those interfaces.

Enabling LDP globally

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable LDP for the local node or for a VPN.	<ul style="list-style-type: none">• Enable LDP for the local node and enter LDP view: mpls ldp• Enable LDP for a VPN and enter LDP:VPN instance view:<ul style="list-style-type: none">a. mpls ldpb. vpn-instance <i>vpn-instance-name</i>	By default, LDP is disabled.
3. Configure an LDP LSR ID.	lsr-id <i>lsr-id</i>	By default, the LDP LSR ID is the same as the MPLS LSR ID.

Enabling LDP on an interface

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type interface-number</i>	If the interface is bound to a VPN instance, you must enable LDP for the VPN instance by using the vpn-instance command in LDP view.
3. Enable LDP on the interface.	mpls ldp enable	By default, LDP is disabled on an interface.

Configuring Link Hello parameters

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter the view of the interface where you want to establish an LDP session.	interface <i>interface-type interface-number</i>	N/A
3. Configure the Link Hello hold time.	mpls ldp timer hello-hold <i>timeout</i>	By default, the Link Hello hold time is 15 seconds.

Step	Command	Remarks
4. Configure the Link Hello interval.	mpls ldp timer hello-interval <i>interval</i>	By default, the Link Hello interval is five seconds.

Configuring LDP session parameters

This task configures the following LDP session parameters:

- **Keepalive hold time and Keepalive interval.**
- **LDP transport address**—IP address for establishing TCP connections.

When you configure LDP session parameters, follow these guidelines:

- The configured LDP transport address must be the IP address of an up interface on the device. Otherwise, no LDP session can be established.
- Make sure the LDP transport addresses of the local and peer LSRs can reach each other. Otherwise, no TCP connection can be established.

To configure parameters for LDP sessions:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Configure the Keepalive hold time.	mpls ldp timer keepalive-hold <i>timeout</i>	By default, the Keepalive hold time is 45 seconds.
4. Configure the Keepalive interval.	mpls ldp timer keepalive-interval <i>interval</i>	By default, the Keepalive interval is 15 seconds.
5. Configure the LDP transport address.	mpls ldp transport-address { <i>ip-address</i> interface }	By default, the LDP transport address is the LSR ID of the local device if the interface where you want to establish an LDP session belongs to the public network. If the interface belongs to a VPN, the LDP transport address is the primary IP address of the interface. If the interface where you want to establish an LDP session is bound to a VPN instance, the interface with the IP address specified with this command must be bound to the same VPN instance.

Configuring LDP backoff

If LDP session parameters (for example, the label advertisement mode) are incompatible, two LDP peers cannot establish a session, and they will keep negotiating with each other.

The LDP backoff mechanism can mitigate this problem by using an initial delay timer and a maximum delay timer. After LDP fails to establish a session with a peer LSR for the first time, LDP does not start an attempt until the initial delay timer expires. If the session setup fails again, LDP waits for two times the initial delay before the next attempt, and so forth until the maximum delay time is reached. After that, the maximum delay time will always take effect.

To configure LDP backoff:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter LDP view or enter LDP-VPN instance view.	<ul style="list-style-type: none"> • Enter LDP view: mpls ldp • Enter LDP-VPN instance view: <ul style="list-style-type: none"> a. mpls ldp b. vpn-instance <i>vpn-instance-name</i> 	N/A
3. Configure the initial delay time and maximum delay time.	backoff initial <i>initial-time</i> maximum <i>maximum-time</i>	By default, the initial delay time is 15 seconds and the maximum delay time is 120 seconds.

Configuring LDP MD5 authentication

To improve security for LDP sessions, you can configure MD5 authentication for the underlying TCP connections to check the integrity of LDP messages.

For two LDP peers to establish an LDP session successfully, make sure the LDP MD5 authentication configurations on the LDP peers are consistent.

To configure LDP MD5 authentication:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter LDP view or enter LDP-VPN instance view.	<ul style="list-style-type: none"> • Enter LDP view: mpls ldp • Enter LDP-VPN instance view: <ul style="list-style-type: none"> a. mpls ldp b. vpn-instance <i>vpn-instance-name</i> 	N/A
3. Enable LDP MD5 authentication.	md5-authentication <i>peer-lsr-id</i> { cipher plain } <i>password</i>	By default, LDP MD5 authentication is disabled.

Configuring an LSP generation policy

An LSP generation policy controls the number of LSPs generated by LDP in one of the following ways:

- Use all routes to establish LSPs.
- Use the routes permitted by an IP prefix list to establish LSPs. For information about IP prefix list configuration, see *Layer 3—IP Routing Configuration Guide*.

- Use only host routes with a 32-bit mask to establish LSPs.

By default, LDP uses only host routes with a 32-bit mask to establish LSPs. The other two methods can result in more LSPs than the default policy. To change the policy, be sure that the system resources and bandwidth resources are sufficient.

Configure an LSP generation policy:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter LDP view or enter LDP-VPN instance view.	<ul style="list-style-type: none"> • Enter LDP view: mpls ldp • Enter LDP-VPN instance view: a. mpls ldp b. vpn-instance <i>vpn-instance-name</i> 	N/A
3. Configure an LSP generation policy.	lsp-trigger { all prefix-list <i>prefix-list-name</i> }	By default, LDP uses only host routes with a 32-bit mask to establish LSPs.

Configuring the LDP label distribution control mode

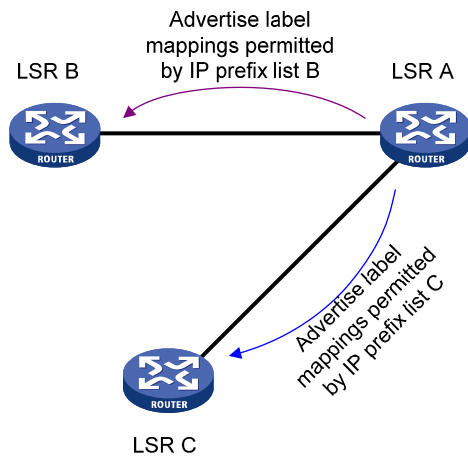
Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter LDP view or enter LDP-VPN instance view.	<ul style="list-style-type: none"> • Enter LDP view: mpls ldp • Enter LDP-VPN instance view: a. mpls ldp b. vpn-instance <i>vpn-instance-name</i> 	N/A
3. Configure the label distribution control mode.	label-distribution { independent ordered }	By default, the Ordered label distribution mode is used. To apply the new setting to LDP sessions established before the command is configured, you must reset the LDP sessions.

Configuring a label advertisement policy

A label advertisement policy uses IP prefix lists to control the FEC-label mappings advertised to peers.

As shown in [Figure 14](#), LSR A advertises label mappings for FECs permitted by IP prefix list B to LSR B and advertises label mappings for FECs permitted by IP prefix list C to LSR C.

Figure 14 Label advertisement control diagram



A label advertisement policy on an LSR and a label acceptance policy on its upstream LSR can achieve the same purpose. HP recommends that you use label advertisement policies to reduce network load if downstream LSRs support label advertisement control.

Before you configure an LDP label advertisement policy, create an IP prefix list. For information about IP prefix list configuration, see *Layer 3—IP Routing Configuration Guide*.

To configure a label advertisement policy:

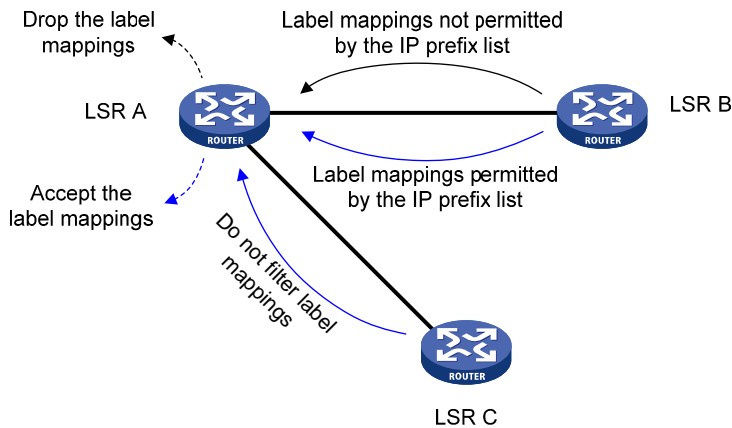
Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter LDP view or enter LDP-VPN instance view.	<ul style="list-style-type: none"> • Enter LDP view: mpls ldp • Enter LDP-VPN instance view: a. mpls ldp b. vpn-instance <i>vpn-instance-name</i> 	N/A
3. Configure a label advertisement policy.	advertise-label prefix-list <i>prefix-list-name</i> [peer <i>peer-prefix-list-name</i>]	By default, LDP advertises all label mappings permitted by the LSP generation policy to all peers.

Configuring a label acceptance policy

A label acceptance policy uses an IP prefix list to control the label mappings received from a peer.

As shown in [Figure 15](#), LSR A uses an IP prefix list to filter label mappings from LSR B, and it does not filter label mappings from LSR C.

Figure 15 Label acceptance control diagram



A label advertisement policy on an LSR and a label acceptance policy on its upstream LSR can achieve the same purpose. HP recommends using the label advertisement policy to reduce network load.

You must create an IP prefix list before you configure a label acceptance policy. For information about IP prefix list configuration, see *Layer 3—IP Routing Configuration Guide*.

To configure a label acceptance policy:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter LDP view or enter LDP-VPN instance view.	<ul style="list-style-type: none"> • Enter LDP view: mpls ldp • Enter LDP-VPN instance view: <ul style="list-style-type: none"> a. mpls ldp b. vpn-instance vpn-instance-name 	N/A
3. Configure a label acceptance policy.	accept-label peer peer-lsr-id prefix-list prefix-list-name	By default, LDP accepts all label mappings.

Configuring LDP loop detection

LDP detects and terminate LSP loops in the following ways:

- **Maximum hop count**—LDP adds a hop count in a label request or label mapping message. The hop count value increments by 1 on each LSR. When the maximum hop count is reached, LDP considers that a loop has occurred and terminates the establishment of the LSP.
- **Path vector**—LDP adds LSR ID information in a label request or label mapping message. Each LSR checks whether its LSR ID is contained in the message. If not, the LSR adds its own LSR ID into the message. If yes, the LSR considers that a loop has occurred and terminates LSP establishment. In addition, when the number of LSR IDs in the message reaches the path vector limit, LDP also considers that a loop has occurred and terminates LSP establishment.

To configure LDP loop detection:

Step	Command	Remarks
1. Enter system view.	system-view	N/A

Step	Command	Remarks
2. Enter LDP view or enter LDP-VPN instance view.	<ul style="list-style-type: none"> Enter LDP view: mpls ldp Enter LDP-VPN instance view: <ul style="list-style-type: none"> a. mpls ldp b. vpn-instance <i>vpn-instance-name</i> 	N/A
3. Enable loop detection.	loop-detect	By default, loop detection is disabled. After loop detection is enabled, the device uses both the maximum hop count and the path vector methods to detect loops.
4. Specify the maximum hop count.	maxhops <i>hop-number</i>	By default, the maximum hop count is 32.
5. Specify the path vector limit.	pv-limit <i>pv-number</i>	By default, the path vector limit is 32.

NOTE:

The LDP loop detection feature is applicable only in networks comprised of devices that do not support TTL mechanism, such as ATM switches. Do not use LDP loop detection on other networks because it only results in extra LDP overhead.

Configuring LDP GR

Before you configure LDP GR, enable LDP on the GR restarter and GR helpers.

To configure LDP GR:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter LDP view.	mpls ldp	N/A
3. Enable LDP GR.	graceful-restart	By default, LDP GR is disabled.
4. Configure the Reconnect timer for LDP GR.	graceful-restart timer reconnect <i>reconnect-time</i>	By default, the Reconnect time is 120 seconds.
5. Configure the MPLS Forwarding State Holding timer for LDP GR.	graceful-restart timer forwarding-hold <i>hold-time</i>	By default, the MPLS Forwarding State Holding time is 180 seconds.

Resetting LDP sessions

Changes to LDP session parameters, except the MD5 authentication key, do not take effect on existing LDP sessions. To validate the changes, you must reset the LDP sessions.

Execute the **reset mpls ldp** command in user view.

Task	Command
Reset LDP sessions.	<code>reset mpls ldp [vpn-instance vpn-instance-name] [peer peer-id]</code>

Enabling SNMP notifications for LDP

This feature enables generating SNMP notifications for LDP upon LDP session changes, as defined in RFC 3815. The generated SNMP notifications are sent to the SNMP module.

To enable SNMP notifications for LDP:

Step	Command	Remarks
1. Enter system view.	<code>system-view</code>	N/A
2. Enable SNMP notifications for LDP.	<code>snmp-agent trap enable ldp</code>	By default, SNMP notifications for LDP are enabled.

For more information about SNMP notifications, see *Network Management and Monitoring Configuration Guide*.

Displaying and maintaining LDP

Execute **display** commands in any view.

Task	Command
Display LDP discovery information.	<code>display mpls ldp discovery [vpn-instance vpn-instance-name] [interface interface-type interface-number peer peer-lsr-id] [verbose]</code>
Display LDP FEC-label mapping information.	<code>display mpls ldp fec [vpn-instance vpn-instance-name] [destination-address mask-length summary]</code>
Display LDP interface information.	<code>display mpls ldp interface [interface-type interface-number]</code>
Display LDP LSP information.	<code>display mpls ldp lsp [vpn-instance vpn-instance-name] [destination-address mask-length]</code>
Display LDP running parameters.	<code>display mpls ldp parameter [vpn-instance vpn-instance-name]</code>
Display LDP peer and session information.	<code>display mpls ldp peer [vpn-instance vpn-instance-name] [peer-lsr-id] [verbose]</code>
Display LDP summary information.	<code>display mpls ldp summary [all vpn-instance vpn-instance-name]</code>

LDP configuration examples

LDP LSP configuration example

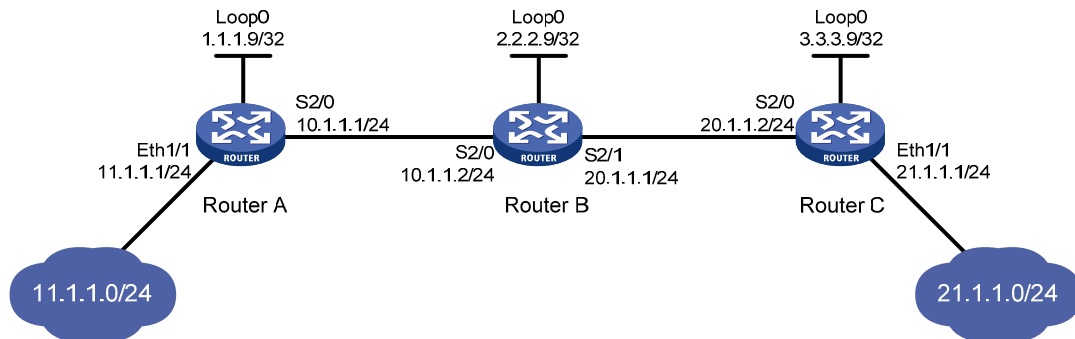
Network requirements

Router A, Router B, and Router C all support MPLS.

Configure LDP to establish LSPs between Router A and Router C, so subnets 11.1.1.0/24 and 21.1.1.0/24 can reach each other over MPLS.

Configure LDP to establish LSPs for only destinations 1.1.1.9/32, 2.2.2.9/32, 3.3.3.9/32, 11.1.1.0/24, and 21.1.1.0/24 on Router A, Router B, and Router C.

Figure 16 Network diagram



Configuration considerations

- LDP assigns labels according to routing information. To establish LDP LSPs, you must configure a routing protocol to make sure the LSRs can reach each other. This example uses OSPF.
- Enable LDP on each LSR.
- To control the number of LSPs, configure an LSP generation policy on each LSR.

Configuration procedure

1. Configure IP addresses and masks for interfaces, including the loopback interfaces, as shown in Figure 16. (Details not shown.)
2. Configure OSPF on each router to ensure IP connectivity between them:

Configure Router A.

```
<RouterA> system-view
[RouterA] ospf
[RouterA-ospf-1] area 0
[RouterA-ospf-1-area-0.0.0.0] network 1.1.1.9 0.0.0.0
[RouterA-ospf-1-area-0.0.0.0] network 10.1.1.0 0.0.0.255
[RouterA-ospf-1-area-0.0.0.0] network 11.1.1.0 0.0.0.255
[RouterA-ospf-1-area-0.0.0.0] quit
[RouterA-ospf-1] quit
```

Configure Router B.

```
<RouterB> system-view
[RouterB] ospf
[RouterB-ospf-1] area 0
[RouterB-ospf-1-area-0.0.0.0] network 2.2.2.9 0.0.0.0
[RouterB-ospf-1-area-0.0.0.0] network 10.1.1.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] network 20.1.1.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] quit
[RouterB-ospf-1] quit
```

Configure Router C.

```
<RouterC> system-view
```

```

[RouterC] ospf
[RouterC-ospf-1] area 0
[RouterC-ospf-1-area-0.0.0.0] network 3.3.3.9 0.0.0.0
[RouterC-ospf-1-area-0.0.0.0] network 20.1.1.0 0.0.0.255
[RouterC-ospf-1-area-0.0.0.0] network 21.1.1.0 0.0.0.255
[RouterC-ospf-1-area-0.0.0.0] quit
[RouterC-ospf-1] quit

```

Verify that the routers have learned the routes to each other. For example, on Router A:

```
[RouterA] display ip routing-table
```

```
Destinations : 21          Routes : 21
```

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
0.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
1.1.1.9/32	Direct	0	0	127.0.0.1	InLoop0
2.2.2.9/32	OSPF	10	1	10.1.1.2	S2/0
3.3.3.9/32	OSPF	10	2	10.1.1.2	S2/0
10.1.1.0/24	Direct	0	0	10.1.1.1	S2/0
10.1.1.0/32	Direct	0	0	10.1.1.1	S2/0
10.1.1.1/32	Direct	0	0	127.0.0.1	InLoop0
10.1.1.255/32	Direct	0	0	10.1.1.1	S2/0
11.1.1.0/24	Direct	0	0	11.1.1.1	Eth1/1
11.1.1.0/32	Direct	0	0	11.1.1.1	Eth1/1
11.1.1.1/32	Direct	0	0	127.0.0.1	InLoop0
11.1.1.255/32	Direct	0	0	11.1.1.1	Eth1/1
20.1.1.0/24	OSPF	10	2	10.1.1.2	S2/0
21.1.1.0/24	OSPF	10	3	10.1.1.2	S2/0
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0
127.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0
224.0.0.0/4	Direct	0	0	0.0.0.0	NULL0
224.0.0.0/24	Direct	0	0	0.0.0.0	NULL0
255.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0

3. Enable MPLS and LDP:

Configure Router A.

```

[RouterA] mpls lsr-id 1.1.1.9
[RouterA] mpls ldp
[RouterA-ldp] quit
[RouterA] interface serial 2/0
[RouterA-Serial2/0] mpls enable
[RouterA-Serial2/0] mpls ldp enable
[RouterA-Serial2/0] quit

```

Configure Router B.

```

[RouterB] mpls lsr-id 2.2.2.9
[RouterB] mpls ldp
[RouterB-ldp] quit
[RouterB] interface serial 2/0

```

```

[RouterB-Serial2/0] mpls enable
[RouterB-Serial2/0] mpls ldp enable
[RouterB-Serial2/0] quit
[RouterB] interface serial 2/1
[RouterB-Serial2/1] mpls enable
[RouterB-Serial2/1] mpls ldp enable
[RouterB-Serial2/1] quit

```

Configure Router C.

```

[RouterC] mpls lsr-id 3.3.3.9
[RouterC] mpls ldp
[RouterC-ldp] quit
[RouterC] interface serial 2/0
[RouterC-Serial2/0] mpls enable
[RouterC-Serial2/0] mpls ldp enable
[RouterC-Serial2/0] quit

```

4. Configure LSP generation policies:

On Router A, create IP prefix list **routera**, and configure LDP to use only the routes permitted by the prefix list to establish LSPs.

```

[RouterA] ip prefix-list routera index 10 permit 1.1.1.9 32
[RouterA] ip prefix-list routera index 20 permit 2.2.2.9 32
[RouterA] ip prefix-list routera index 30 permit 3.3.3.9 32
[RouterA] ip prefix-list routera index 40 permit 11.1.1.0 24
[RouterA] ip prefix-list routera index 50 permit 21.1.1.0 24
[RouterA] mpls ldp
[RouterA-ldp] lsp-trigger prefix-list routera
[RouterA-ldp] quit

```

On Router B, create IP prefix list **routerb**, and configure LDP to use only the routes permitted by the prefix list to establish LSPs.

```

[RouterB] ip prefix-list routerb index 10 permit 1.1.1.9 32
[RouterB] ip prefix-list routerb index 20 permit 2.2.2.9 32
[RouterB] ip prefix-list routerb index 30 permit 3.3.3.9 32
[RouterB] ip prefix-list routerb index 40 permit 11.1.1.0 24
[RouterB] ip prefix-list routerb index 50 permit 21.1.1.0 24
[RouterB] mpls ldp
[RouterB-ldp] lsp-trigger prefix-list routerb
[RouterB-ldp] quit

```

On Router C, create IP prefix list **routerc**, and configure LDP to use only the routes permitted by the prefix list to establish LSPs.

```

[RouterC] ip prefix-list routerc index 10 permit 1.1.1.9 32
[RouterC] ip prefix-list routerc index 20 permit 2.2.2.9 32
[RouterC] ip prefix-list routerc index 30 permit 3.3.3.9 32
[RouterC] ip prefix-list routerc index 40 permit 11.1.1.0 24
[RouterC] ip prefix-list routerc index 50 permit 21.1.1.0 24
[RouterC] mpls ldp
[RouterC-ldp] lsp-trigger prefix-list routerc
[RouterC-ldp] quit

```

Verifying the configuration

Execute the **display mpls ldp lsp** command on each router to view the LDP LSP information. For example, on Router A:

```
[RouterA] display mpls ldp lsp
Status Flags: * - stale, L - liberal
Statistics:
  FECs: 5      Ingress LSPs: 3      Transit LSPs: 3      Egress LSPs: 2

FEC                In/Out Label      Nexthop           OutInterface
1.1.1.9/32         3/-
                  -/1279(L)
2.2.2.9/32         -/3               10.1.1.2          S2/0
                  1279/3            10.1.1.2          S2/0
3.3.3.9/32         -/1278            10.1.1.2          S2/0
                  1278/1278        10.1.1.2          S2/0
11.1.1.0/24        1277/-
                  -/1277(L)
21.1.1.0/24        -/1276            10.1.1.2          S2/0
                  1276/1276        10.1.1.2          S2/0
```

On Router A, test the connectivity of the LDP LSP from Router A to Router C.

```
[RouterA] ping mpls -a 11.1.1.1 ipv4 21.1.1.0 24
MPLS Ping FEC: 21.1.1.0/24 : 100 data bytes
100 bytes from 20.1.1.2: Sequence=1 time=1 ms
100 bytes from 20.1.1.2: Sequence=2 time=1 ms
100 bytes from 20.1.1.2: Sequence=3 time=8 ms
100 bytes from 20.1.1.2: Sequence=4 time=2 ms
100 bytes from 20.1.1.2: Sequence=5 time=1 ms

--- FEC: 21.1.1.0/24 ping statistics ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max = 1/2/8 ms
```

On Router C, test the connectivity of the LDP LSP from Router C to Router A.

```
[RouterC] ping mpls -a 21.1.1.1 ipv4 11.1.1.0 24
MPLS Ping FEC: 11.1.1.0/24 : 100 data bytes
100 bytes from 10.1.1.1: Sequence=1 time=1 ms
100 bytes from 10.1.1.1: Sequence=2 time=1 ms
100 bytes from 10.1.1.1: Sequence=3 time=1 ms
100 bytes from 10.1.1.1: Sequence=4 time=1 ms
100 bytes from 10.1.1.1: Sequence=5 time=1 ms

--- FEC: 11.1.1.0/24 ping statistics ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max = 1/1/1 ms
```

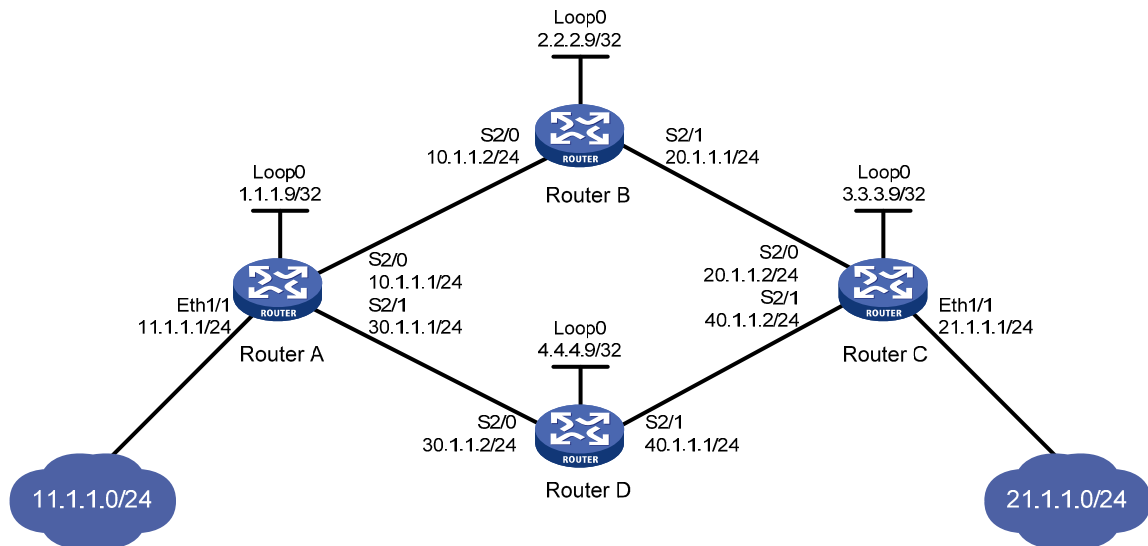
Label acceptance control configuration example

Network requirements

Two links, Router A—Router B—Router C and Router A—Router D—Router C, exist between subnets 11.1.0/24 and 21.1.0/24.

Configure label acceptance control, so LDP sets up LSPs only on the link Router A—Router B—Router C to forward traffic between subnets 11.1.0/24 and 21.1.0/24.

Figure 17 Network diagram



Configuration considerations

1. Configure a routing protocol on each router to make sure that the routers can reach each other. This example uses OSPF.
2. Enable LDP on each router.
3. Configure LSP generation policies, so LDP establishes LSPs only for the routes 11.1.0/24 and 21.1.0/24.
4. Configure label acceptance policies, so LDP sets up LSPs only over the link Router A—Router B—Router C, as follows:
 - Router A accepts only the label mapping for FEC 21.1.0/24 received from Router B. Router A denies the label mapping for FEC 21.1.0/24 received from Router D.
 - Router C accepts only the label mapping for FEC 11.1.0/24 received from Router B. Router C denies the label mapping for FEC 11.1.0/24 received from Router D.

Configuration procedure

1. Configure IP addresses and masks for interfaces, including the loopback interfaces, as shown in Figure 17. (Details not shown.)
2. Configure OSPF on each router to ensure IP connectivity between them. (Details not shown.)
3. Enable MPLS and LDP:

```
# Configure Router A.  
<RouterA> system-view  
[RouterA] mpls lsr-id 1.1.1.9
```



```
[RouterA] mpls ldp
[RouterA-ldp] quit
[RouterA] interface serial 2/0
[RouterA-Serial2/0] mpls enable
[RouterA-Serial2/0] mpls ldp enable
[RouterA-Serial2/0] quit
[RouterA] interface serial 2/1
[RouterA-Serial2/1] mpls enable
[RouterA-Serial2/1] mpls ldp enable
[RouterA-Serial2/1] quit
```

Configure Router B.

```
<RouterB> system-view
[RouterB] mpls lsr-id 2.2.2.9
[RouterB] mpls ldp
[RouterB-ldp] quit
[RouterB] interface serial 2/0
[RouterB-Serial2/0] mpls enable
[RouterB-Serial2/0] mpls ldp enable
[RouterB-Serial2/0] quit
[RouterB] interface serial 2/1
[RouterB-Serial2/1] mpls enable
[RouterB-Serial2/1] mpls ldp enable
[RouterB-Serial2/1] quit
```

Configure Router C.

```
<RouterC> system-view
[RouterC] mpls lsr-id 3.3.3.9
[RouterC] mpls ldp
[RouterC-ldp] quit
[RouterC] interface serial 2/0
[RouterC-Serial2/0] mpls enable
[RouterC-Serial2/0] mpls ldp enable
[RouterC-Serial2/0] quit
[RouterC] interface serial 2/1
[RouterC-Serial2/1] mpls enable
[RouterC-Serial2/1] mpls ldp enable
[RouterC-Serial2/1] quit
```

Configure Router D.

```
<RouterD> system-view
[RouterD] mpls lsr-id 4.4.4.9
[RouterD] mpls ldp
[RouterD-ldp] quit
[RouterD] interface serial 2/0
[RouterD-Serial2/0] mpls enable
[RouterD-Serial2/0] mpls ldp enable
[RouterD-Serial2/0] quit
[RouterD] interface serial 2/1
[RouterD-Serial2/1] mpls enable
[RouterD-Serial2/1] mpls ldp enable
```

```
[RouterD-Serial2/1] quit
```

4. Configure LSP generation policies:

On Router A, create IP prefix list **routera**, and configure LDP to use only the routes permitted by the prefix list to establish LSPs.

```
[RouterA] ip prefix-list routera index 10 permit 11.1.1.0 24
[RouterA] ip prefix-list routera index 20 permit 21.1.1.0 24
[RouterA] mpls ldp
[RouterA-ldp] lsp-trigger prefix-list routera
[RouterA-ldp] quit
```

On Router B, create IP prefix list **routerb**, and configure LDP to use only the routes permitted by the prefix list to establish LSPs.

```
[RouterB] ip prefix-list routerb index 10 permit 11.1.1.0 24
[RouterB] ip prefix-list routerb index 20 permit 21.1.1.0 24
[RouterB] mpls ldp
[RouterB-ldp] lsp-trigger prefix-list routerb
[RouterB-ldp] quit
```

On Router C, create IP prefix list **routerc**, and configure LDP to use only the routes permitted by the prefix list to establish LSPs.

```
[RouterC] ip prefix-list routerc index 10 permit 11.1.1.0 24
[RouterC] ip prefix-list routerc index 20 permit 21.1.1.0 24
[RouterC] mpls ldp
[RouterC-ldp] lsp-trigger prefix-list routerc
[RouterC-ldp] quit
```

On Router D, create IP prefix list **routerd**, and configure LDP to use only the routes permitted by the prefix list to establish LSPs.

```
[RouterD] ip prefix-list routerd index 10 permit 11.1.1.0 24
[RouterD] ip prefix-list routerd index 20 permit 21.1.1.0 24
[RouterD] mpls ldp
[RouterD-ldp] lsp-trigger prefix-list routerd
[RouterD-ldp] quit
```

5. Configure label acceptance policies:

On Router A, create an IP prefix list **prefix-from-b** that permits subnet 21.1.1.0/24. Router A uses this list to filter FEC-label mappings received from Router B.

```
[RouterA] ip prefix-list prefix-from-b index 10 permit 21.1.1.0 24
```

On Router A, create an IP prefix list **prefix-from-d** that denies subnet 21.1.1.0/24. Router A uses this list to filter FEC-label mappings received from Router D.

```
[RouterA] ip prefix-list prefix-from-d index 10 deny 21.1.1.0 24
```

On Router A, configure label acceptance policies to filter FEC-label mappings received from Router B and Router D.

```
[RouterA] mpls ldp
[RouterA-ldp] accept-label peer 2.2.2.9 prefix-list prefix-from-b
[RouterA-ldp] accept-label peer 4.4.4.9 prefix-list prefix-from-d
[RouterA-ldp] quit
```

On Router C, create an IP prefix list **prefix-from-b** that permits subnet 11.1.1.0/24. Router C uses this list to filter FEC-label mappings received from Router B.

```
[RouterC] ip prefix-list prefix-from-b index 10 permit 11.1.1.0 24
```

On Router C, create an IP prefix list **prefix-from-d** that denies subnet 11.1.1.0/24. Router A uses this list to filter FEC-label mappings received from Router D.

```
[RouterC] ip prefix-list prefix-from-d index 10 deny 11.1.1.0 24
```

On Router C, configure label acceptance policies to filter FEC-label mappings received from Router B and Router D.

```
[RouterC] mpls ldp
```

```
[RouterC-ldp] accept-label peer 2.2.2.9 prefix-list prefix-from-b
```

```
[RouterC-ldp] accept-label peer 4.4.4.9 prefix-list prefix-from-d
```

```
[RouterC-ldp] quit
```

Verifying the configuration

Execute the **display mpls ldp lsp** command on each router to view the LDP LSP information. For example, on Router A:

```
[RouterA] display mpls ldp lsp
```

```
Status Flags: * - stale, L - liberal
```

```
Statistics:
```

```
  FECs: 2          Ingress LSPs: 1          Transit LSPs: 1          Egress LSPs: 1
```

FEC	In/Out Label	Nexthop	OutInterface
11.1.1.0/24	1277/-		
	-/1148(L)		
21.1.1.0/24	-/1149(L)		
	-/1276	10.1.1.2	S2/0
	1276/1276	10.1.1.2	S2/0

The output shows that the next hop of the LSP for FEC 21.1.1.0/24 is Router B (10.1.1.2). The LSP has been set up over the link Router A—Router B—Router C, not over the link Router A—Router D—Router C.

On Router A, test the connectivity of the LDP LSP from Router A to Router C.

```
[RouterA] ping mpls -a 11.1.1.1 ipv4 21.1.1.0 24
```

```
MPLS Ping FEC: 21.1.1.0/24 : 100 data bytes
```

```
100 bytes from 20.1.1.2: Sequence=1 time=1 ms
```

```
100 bytes from 20.1.1.2: Sequence=2 time=1 ms
```

```
100 bytes from 20.1.1.2: Sequence=3 time=8 ms
```

```
100 bytes from 20.1.1.2: Sequence=4 time=2 ms
```

```
100 bytes from 20.1.1.2: Sequence=5 time=1 ms
```

```
--- FEC: 21.1.1.0/24 ping statistics ---
```

```
5 packets transmitted, 5 packets received, 0.0% packet loss
```

```
round-trip min/avg/max = 1/2/8 ms
```

On Router C, test the connectivity of the LDP LSP from Router C to Router A.

```
[RouterC] ping mpls -a 21.1.1.1 ipv4 11.1.1.0 24
```

```
MPLS Ping FEC: 11.1.1.0/24 : 100 data bytes
```

```
100 bytes from 10.1.1.1: Sequence=1 time=1 ms
```

```
100 bytes from 10.1.1.1: Sequence=2 time=1 ms
```

```
100 bytes from 10.1.1.1: Sequence=3 time=1 ms
```

```
100 bytes from 10.1.1.1: Sequence=4 time=1 ms
```

```
100 bytes from 10.1.1.1: Sequence=5 time=1 ms
```

```

--- FEC: 11.1.1.0/24 ping statistics ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max = 1/1/1 ms

```

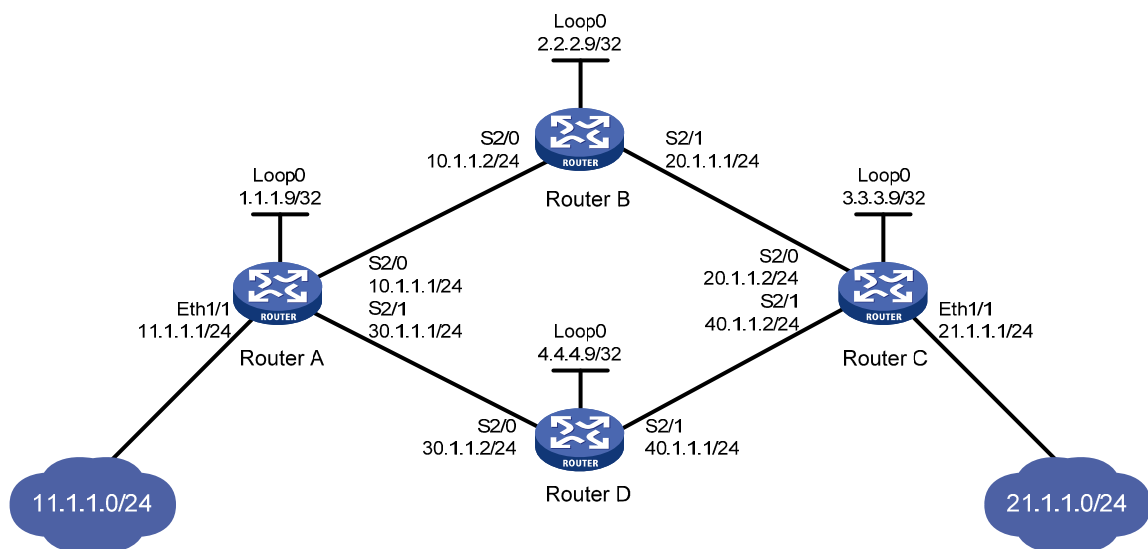
Label advertisement control configuration example

Network requirements

Two links, Router A—Router B—Router C and Router A—Router D—Router C, exist between subnets 11.1.1.0/24 and 21.1.1.0/24.

Configure label advertisement control, so LDP sets up LSPs only on the link Router A—Router B—Router C to forward traffic between subnets 11.1.1.0/24 and 21.1.1.0/24.

Figure 18 Network diagram



Configuration considerations

1. Configure a routing protocol on each router to make sure that the routers can reach each other. This example uses OSPF.
2. Enable LDP on each router.
3. Configure LSP generation policies so LDP uses only the routes 11.1.1.0/24 and 21.1.1.0/24 to establish LSPs.
4. Configure label advertisement policies, so LDP sets up LSPs only over the link Router A—Router B—Router C, as follows:
 - o Router A advertises only the label mapping for FEC 11.1.1.0/24 to Router B.
 - o Router C advertises only the label mapping for FEC 21.1.1.0/24 to Router B.
 - o Router D does not advertise label mapping for FEC 21.1.1.0/24 to Router A. Router D does not advertise label mapping for FEC 11.1.1.0/24 to Router C.

Configuration procedure

1. Configure IP addresses and masks for interfaces, including the loopback interfaces, as shown in Figure 18. (Details not shown.)
2. Configure OSPF on each router to ensure IP connectivity between them. (Details not shown.)

3. Enable MPLS and LDP:

Configure Router A.

```
<RouterA> system-view
[RouterA] mpls lsr-id 1.1.1.9
[RouterA] mpls ldp
[RouterA-ldp] quit
[RouterA] interface serial 2/0
[RouterA-Serial2/0] mpls enable
[RouterA-Serial2/0] mpls ldp enable
[RouterA-Serial2/0] quit
[RouterA] interface serial 2/1
[RouterA-Serial2/1] mpls enable
[RouterA-Serial2/1] mpls ldp enable
[RouterA-Serial2/1] quit
```

Configure Router B.

```
<RouterB> system-view
[RouterB] mpls lsr-id 2.2.2.9
[RouterB] mpls ldp
[RouterB-ldp] quit
[RouterB] interface serial 2/0
[RouterB-Serial2/0] mpls enable
[RouterB-Serial2/0] mpls ldp enable
[RouterB-Serial2/0] quit
[RouterB] interface serial 2/1
[RouterB-Serial2/1] mpls enable
[RouterB-Serial2/1] mpls ldp enable
[RouterB-Serial2/1] quit
```

Configure Router C.

```
<RouterC> system-view
[RouterC] mpls lsr-id 3.3.3.9
[RouterC] mpls ldp
[RouterC-ldp] quit
[RouterC] interface serial 2/0
[RouterC-Serial2/0] mpls enable
[RouterC-Serial2/0] mpls ldp enable
[RouterC-Serial2/0] quit
[RouterC] interface serial 2/1
[RouterC-Serial2/1] mpls enable
[RouterC-Serial2/1] mpls ldp enable
[RouterC-Serial2/1] quit
```

Configure Router D.

```
<RouterD> system-view
[RouterD] mpls lsr-id 4.4.4.9
[RouterD] mpls ldp
[RouterD-ldp] quit
[RouterD] interface serial 2/0
[RouterD-Serial2/0] mpls enable
```

```
[RouterD-Serial2/0] mpls ldp enable
[RouterD-Serial2/0] quit
[RouterD] interface serial 2/1
[RouterD-Serial2/1] mpls enable
[RouterD-Serial2/1] mpls ldp enable
[RouterD-Serial2/1] quit
```

4. Configure LSP generation policies:

On Router A, create IP prefix list **routera**, and configure LDP to use only the routes permitted by the prefix list to establish LSPs.

```
[RouterA] ip prefix-list routera index 10 permit 11.1.1.0 24
[RouterA] ip prefix-list routera index 20 permit 21.1.1.0 24
[RouterA] mpls ldp
[RouterA-ldp] lsp-trigger prefix-list routera
[RouterA-ldp] quit
```

On Router B, create IP prefix list **routerb**, and configure LDP to use only the routes permitted by the prefix list to establish LSPs.

```
[RouterB] ip prefix-list routerb index 10 permit 11.1.1.0 24
[RouterB] ip prefix-list routerb index 20 permit 21.1.1.0 24
[RouterB] mpls ldp
[RouterB-ldp] lsp-trigger prefix-list routerb
[RouterB-ldp] quit
```

On Router C, create IP prefix list **routerc**, and configure LDP to use only the routes permitted by the prefix list to establish LSPs.

```
[RouterC] ip prefix-listx routerc index 10 permit 11.1.1.0 24
[RouterC] ip prefix-list routerc index 20 permit 21.1.1.0 24
[RouterC] mpls ldp
[RouterC-ldp] lsp-trigger prefix-list routerc
[RouterC-ldp] quit
```

On Router D, create IP prefix list **routerd**, and configure LDP to use only the routes permitted by the prefix list to establish LSPs.

```
[RouterD] ip prefix-list routerd index 10 permit 11.1.1.0 24
[RouterD] ip prefix-list routerd index 20 permit 21.1.1.0 24
[RouterD] mpls ldp
[RouterD-ldp] lsp-trigger prefix-list routerd
[RouterD-ldp] quit
```

5. Configure label advertisement policies:

On Router A, create an IP prefix list **prefix-to-b** that permits subnet 11.1.1.0/24. Router A uses this list to filter FEC-label mappings advertised to Router B.

```
[RouterA] ip prefix-list prefix-to-b index 10 permit 11.1.1.0 24
```

On Router A, create an IP prefix list **peer-b** that permits 2.2.2.9/32. Router A uses this list to filter peers.

```
[RouterA] ip prefix-list peer-b index 10 permit 2.2.2.9 32
```

On Router A, configure a label advertisement policy to advertise only the label mapping for FEC 11.1.1.0/24 to Router B.

```
[RouterA] mpls ldp
[RouterA-ldp] advertise-label prefix-list prefix-to-b peer peer-b
[RouterA-ldp] quit
```

On Router C, create an IP prefix list **prefix-to-b** that permits subnet 21.1.1.0/24. Router C uses this list to filter FEC-label mappings advertised to Router B.

```
[RouterC] ip prefix-list prefix-to-b index 10 permit 21.1.1.0 24
```

On Router C, create an IP prefix list **peer-b** that permits 2.2.2.9/32. Router C uses this list to filter peers.

```
[RouterC] ip prefix-list peer-b index 10 permit 2.2.2.9 32
```

On Router C, configure a label advertisement policy to advertise only the label mapping for FEC 21.1.1.0/24 to Router B.

```
[RouterC] mpls ldp
```

```
[RouterC-ldp] advertise-label prefix-list prefix-to-b peer peer-b
```

```
[RouterC-ldp] quit
```

On Router D, create an IP prefix list **prefix-to-a** that denies subnet 21.1.1.0/24. Router D uses this list to filter FEC-label mappings to be advertised to Router A.

```
[RouterD] ip prefix-list prefix-to-a index 10 deny 21.1.1.0 24
```

```
[RouterD] ip prefix-list prefix-to-a index 20 permit 0.0.0.0 0 less-equal 32
```

On Router D, create an IP prefix list **peer-a** that permits 1.1.1.9/32. Router D uses this list to filter peers.

```
[RouterD] ip prefix-list peer-a index 10 permit 1.1.1.9 32
```

On Router D, create an IP prefix list **prefix-to-c** that denies subnet 11.1.1.0/24. Router D uses this list to filter FEC-label mappings to be advertised to Router C.

```
[RouterD] ip prefix-list prefix-to-c index 10 deny 11.1.1.0 24
```

```
[RouterD] ip prefix-list prefix-to-c index 20 permit 0.0.0.0 0 less-equal 32
```

On Router D, create an IP prefix list **peer-c** that permits subnet 3.3.3.9/32. Router D uses this list to filter peers.

```
[RouterD] ip prefix-list peer-c index 10 permit 3.3.3.9 32
```

On Router D, configure a label advertisement policy, so Router D does not advertise label mappings for FEC 21.1.1.0/24 to Router A, and does not advertise label mappings for FEC 11.1.1.0/24 to Router C.

```
[RouterD] mpls ldp
```

```
[RouterD-ldp] advertise-label prefix-list prefix-to-a peer peer-a
```

```
[RouterD-ldp] advertise-label prefix-list prefix-to-c peer peer-c
```

```
[RouterD-ldp] quit
```

Verifying the configuration

Execute the **display mpls ldp lsp** command on each router to view the LDP LSP information.

```
[RouterA] display mpls ldp lsp
```

```
Status Flags: * - stale, L - liberal
```

```
Statistics:
```

```
FECs: 2          Ingress LSPs: 1          Transit LSPs: 1          Egress LSPs: 1
```

FEC	In/Out Label	Nexthop	OutInterface
11.1.1.0/24	1277/- -/1151(L) -/1277(L)		
21.1.1.0/24	-/1276 1276/1276	10.1.1.2 10.1.1.2	S2/0 S2/0

```
[RouterB] display mpls ldp lsp
```

```
Status Flags: * - stale, L - liberal
```

```
Statistics:
  FECs: 2      Ingress LSPs: 2      Transit LSPs: 2      Egress LSPs: 0
```

FEC	In/Out Label	Nexthop	OutInterface
11.1.1.0/24	-/1277	10.1.1.1	S2/0
	1277/1277	10.1.1.1	S2/0
21.1.1.0/24	-/1149	20.1.1.2	S2/1
	1276/1149	20.1.1.2	S2/1

```
[RouterC] display mpls ldp lsp
Status Flags: * - stale, L - liberal
```

```
Statistics:
  FECs: 2      Ingress LSPs: 1      Transit LSPs: 1      Egress LSPs: 1
```

FEC	In/Out Label	Nexthop	OutInterface
11.1.1.0/24	-/1277	20.1.1.1	S2/0
	1148/1277	20.1.1.1	S2/0
21.1.1.0/24	1149/-		
	-/1276(L)		
	-/1150(L)		

```
[RouterD] display mpls ldp lsp
Status Flags: * - stale, L - liberal
```

```
Statistics:
  FECs: 2      Ingress LSPs: 0      Transit LSPs: 0      Egress LSPs: 2
```

FEC	In/Out Label	Nexthop	OutInterface
11.1.1.0/24	1151/-		
	-/1277(L)		
21.1.1.0/24	1150/-		

The output shows that Router A and Router C has received FEC-label mappings only from Router B. Router B has received FEC-label mappings from both Router A and Router C. Router D does not receive FEC-label mappings from Router A or Router C. LDP has set up an LSP only over the link Router A—Router B—Router C.

On Router A, test the connectivity of the LDP LSP from Router A to Router C.

```
[RouterA] ping mpls -a 11.1.1.1 ipv4 21.1.1.0 24
```

```
MPLS Ping FEC: 21.1.1.0/24 : 100 data bytes
100 bytes from 20.1.1.2: Sequence=1 time=1 ms
100 bytes from 20.1.1.2: Sequence=2 time=1 ms
100 bytes from 20.1.1.2: Sequence=3 time=8 ms
100 bytes from 20.1.1.2: Sequence=4 time=2 ms
100 bytes from 20.1.1.2: Sequence=5 time=1 ms
```

```
--- FEC: 21.1.1.0/24 ping statistics ---
```

```
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max = 1/2/8 ms
```

On Router C, test the connectivity of the LDP LSP from Router C to Router A.

```
[RouterC] ping mpls -a 21.1.1.1 ipv4 11.1.1.0 24
```

```
MPLS Ping FEC: 11.1.1.0/24 : 100 data bytes
100 bytes from 10.1.1.1: Sequence=1 time=1 ms
```



```
100 bytes from 10.1.1.1: Sequence=2 time=1 ms
100 bytes from 10.1.1.1: Sequence=3 time=1 ms
100 bytes from 10.1.1.1: Sequence=4 time=1 ms
100 bytes from 10.1.1.1: Sequence=5 time=1 ms
```

```
--- FEC: 11.1.1.0/24 ping statistics ---
```

```
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max = 1/1/1 ms
```

Configuring MPLS TE

Overview

TE and MPLS TE

Network congestion can degrade the network backbone performance. It might occur when network resources are inadequate or when load distribution is unbalanced. Traffic engineering (TE) is intended to avoid the latter situation where partial congestion might occur because of improper resource allocation.

TE can make the best use of network resources and avoid uneven load distribution by real-time monitoring of traffic and traffic load on network elements and dynamic tuning of traffic management attributes, routing parameters, and resources constraints.

MPLS TE combines the MPLS technology and traffic engineering. It reserves resources by establishing LSP tunnels along the specified paths, allowing traffic to bypass congested nodes to achieve appropriate load distribution.

MPLS TE features simplicity and good scalability. With MPLS TE, a service provider can deploy traffic engineering on the existing MPLS backbone to provide various services and optimize network resources management.

MPLS TE basic concepts

- **CRLSP**—Constraint-based Routed Label Switched Path. To establish a CRLSP, you must configure routing (as you do for a normal LSP), and specify constraints, such as the bandwidth and explicit paths.
- **MPLS TE tunnel**—A virtual point-to-point connection from the ingress node to the egress node. Usually, an MPLS TE tunnel consists of one CRLSP. To deploy CRLSP backup or fast reroute, or transmit traffic over multiple paths, you need to establish multiple CRLSPs for one class of traffic. In this case, an MPLS TE tunnel consists of a set of CRLSPs. An MPLS TE tunnel is identified by an MPLS TE tunnel interface on the ingress node. When the outgoing interface of a traffic flow is an MPLS TE tunnel interface, the traffic flow is forwarded through the CRLSP of the MPLS TE tunnel.

Static CRLSP establishment

A static CRLSP is established by manually specifying the incoming label, outgoing label, and other constraints (such as bandwidth) on each hop (including the ingress, transit, and egress nodes) along the path that the traffic travels. Static CRLSPs feature simple configuration, but they cannot automatically adapt to network changes.

For more information about static CRLSPs, see "Configuring a static CRLSP."

Dynamic CRLSP establishment

Dynamic CRLSPs are dynamically established as follows:

1. An IGP advertises TE attributes for links.
2. MPLS TE uses the CSPF algorithm to calculate the shortest path that meets the constraints (such as bandwidth and explicit routing) to the tunnel destination.
3. A label distribution protocol (such as RSVP-TE) advertises labels to establish CRLSPs and reserve bandwidth resources on each node along the calculated path.

Dynamic CRLSPs adapt to network changes and support CRLSP backup and fast reroute, but they require complicated configurations.

Advertising TE attributes

MPLS TE uses extended link state IGPs, such as OSPF and IS-IS, to advertise TE attributes for links.

TE attributes include the maximum bandwidth, maximum reservable bandwidth, non-reserved bandwidth for each priority, and the link attribute. The IGP floods TE attributes on the network. Each node collects the TE attributes of all links on all routers within the local area or at the same level to build up a TE database (TEDB).

Calculating paths

Based on the TEDB, MPLS TE uses the Constraint-based Shortest Path First (CSPF) algorithm, an improved SPF algorithm, to calculate the shortest, TE constraints-compliant path to the tunnel destination.

CSPF first prunes TE constraints-incompliant links from the TEDB and then performs SPF calculation to identify the shortest path (a set of LSR addresses) to an egress. CSPF calculation is usually performed on the ingress node of an MPLS TE tunnel.

TE constraints include the bandwidth, affinity, setup and holding priorities, and explicit path. They are configured on the ingress node of an MPLS TE tunnel.

- Bandwidth

Bandwidth constraints specify the class of service and the required bandwidth for the traffic to be forwarded along the MPLS TE tunnel. A link complies with the bandwidth constraints when the reservable bandwidth for the class type is greater than or equal to the bandwidth required by the class type.

- Affinity

Affinity determines which links a tunnel can use. The affinity attribute and its mask, and the link attribute are all 32-bit long. A link is available for a tunnel if the link attribute meets the following requirements:

- The link attribute bits corresponding to the affinity attribute's 1 bits whose mask bits are 1 must have at least one bit set to 1.
- The link attribute bits corresponding to the affinity attribute's 0 bits whose mask bits are 1 must have no bit set to 1.

The link attribute bits corresponding to the 0 bits in the affinity mask are not checked.

For example, if the affinity attribute is 0xFFFFFFFF0 and its mask is 0x0000FFFF, a link is available for the tunnel when its link attribute bits meet the following requirements: the highest 16 bits each can be 0 or 1 (no requirements), the 17th through 28th bits must have at least one bit whose value is 1, and the lowest four bits must be 0.

- Setup priority and holding priority

If MPLS TE cannot find a qualified path for an MPLS TE tunnel, it can remove an existing MPLS TE tunnel and preempt its bandwidth to set up the new MPLS TE tunnel.

MPLS TE uses the setup priority and holding priority to make preemption decisions. For a new MPLS TE tunnel to preempt an existing MPLS TE tunnel, the setup priority of the new tunnel must be

higher than the holding priority of the existing tunnel. Both setup and holding priorities are in the range of 0 to 7. A smaller value indicates a higher priority.

To avoid flapping caused by improper preemptions, the setup priority of a tunnel must not be higher than its holding priority, namely, the setup priority value must be equal to or greater than the holding priority value.

- **Explicit path**

Explicit path specifies the nodes to pass and the nodes to not pass for a tunnel.

Explicit paths include the following types:

- **Strict explicit path**—Among the nodes that the path must traverse, a node and its previous hop must be connected directly.
- **Loose explicit path**—Among the nodes that the path must traverse, a node and its previous hop can be connected indirectly.

Strict explicit path precisely specifies the path that an MPLS TE tunnel must traverse. Loose explicit path vaguely specifies the path that an MPLS TE tunnel must traverse. Strict explicit path and loose explicit path can be used together to specify that some nodes are directly connected and some nodes have other nodes in between.

Setting up a CRLSP through RSVP-TE

After calculating a path by using CSPF, MPLS TE uses a label distribution protocol to set up the CRLSP and reserves resources on each node of the path.

The device supports the label distribution protocol of RSVP-TE for MPLS TE. Resource Reservation Protocol (RSVP) reserves resources on each node along a path. Extended RSVP can support MPLS label distribution and allow resource reservation information to be transmitted with label bindings. This extended RSVP is called "RSVP-TE."

For more information about RSVP, see "Configuring RSVP."

Traffic forwarding

After an MPLS TE tunnel is established, traffic is not forwarded on the tunnel automatically. You must direct the traffic to the tunnel by using one of the following methods.

Static routing

You can direct traffic to an MPLS TE tunnel by creating a static route that reaches the destination through the tunnel interface. This is the easiest way to implement MPLS TE tunnel forwarding. However, when the traffic to multiple networks is to be forwarded through the MPLS TE tunnel, you must configure multiple static routes, which are complicated to configure and difficult to maintain.

For more information about static routing, see *Layer 3—IP Routing Configuration Guide*.

Policy-based routing

You can configure PBR on the ingress interface of traffic to direct the traffic that matches an ACL to the MPLS TE tunnel interface.

PBR can match the traffic to be forwarded on the tunnel not only by destination IP address, but also by source IP address, protocol type, and other criteria. Compared with static routing, PBR is more flexible but requires more complicated configuration.

For more information about policy-based routing, see *Layer 3—IP Routing Configuration Guide*.

Make-before-break

Make-before-break is a mechanism to change an MPLS TE tunnel with minimum data loss and without using extra bandwidth.

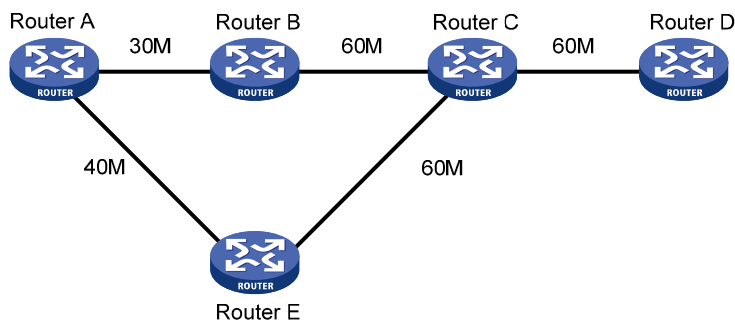
In cases of tunnel reoptimization, traffic forwarding is interrupted if the existing CRLSP is removed before a new CRLSP is established. The make-before-break mechanism makes sure that the existing CRLSP is removed after the new CRLSP is established and the traffic is switched to the new CRLSP. However, this might waste bandwidth resources if some links on the old and new CRLSPs are the same, because you need to reserve bandwidth on these links for both the old and new CRLSPs. The make-before-break mechanism uses the SE resource reservation style to address this problem.

The resource reservation style refers to the style in which RSVP-TE reserves bandwidth resources during CRLSP establishment. The resource reservation style used by an MPLS TE tunnel is determined by the ingress node, and is advertised to other nodes through RSVP.

The device supports the following resource reservation styles:

- **FF**—Fixed-filter style, where resources are reserved for individual senders and cannot be shared among senders on the same session.
- **SE**—Shared-explicit style, where resources are reserved for senders on the same session and shared among them. SE is mainly used for make-before-break.

Figure 19 Diagram for make-before-break



As shown in [Figure 19](#), a CRLSP with 30 M reserved bandwidth has been set up from Router A to Router D through the path Router A—Router B—Router C—Router D.

To increase the reserved bandwidth to 40 M, a new CRLSP must be set up through the path Router A—Router E—Router C—Router D. To achieve this purpose, RSVP-TE needs to reserve 30M bandwidth for the old CRLSP and 40M bandwidth for the new CRLSP on the link Router C—Router D, but the link bandwidth is not enough.

Using the make-before-break mechanism, the new CRLSP can share the bandwidth reserved for the old CRLSP. After the new CRLSP is set up, traffic is switched to the new CRLSP without service interruption, and then the old CRLSP is removed.

Route pinning

Route pinning enables CRLSPs to always use the original optimal path even if a new optimal route has been learned.

On a network where route changes frequently occur, you can use route pinning to avoid re-establishing CRLSPs upon route changes.

Tunnel reoptimization

Tunnel reoptimization allows you to manually or dynamically trigger the ingress node to recalculate a path. If the ingress node recalculates a better path, it creates a new CRLSP, switches traffic from the old CRLSP to the new, and then deletes the old CRLSP.

MPLS TE uses the tunnel reoptimization function to implement dynamic CRLSP optimization. For example, when MPLS TE sets up a tunnel, if a link on the optimal path does not have enough reservable bandwidth, MPLS TE sets up the tunnel on another path. When the link has enough bandwidth, the tunnel optimization function can switch the MPLS TE tunnel to the optimal path.

CRLSP backup

CRLSP backup uses a CRLSP to back up a primary CRLSP. When the ingress detects that the primary CRLSP fails, it switches traffic to the backup CRLSP. When the primary CRLSP recovers, the ingress switches traffic back.

CRLSP backup has the following modes:

- **Hot standby**—A backup CRLSP is created immediately after a primary CRLSP is created.
- **Ordinary**—A backup CRLSP is created after the primary CRLSP fails.

FRR

Fast reroute (FRR) protects CRLSPs from link and node failures. FRR can implement 50-millisecond CRLSP failover.

After FRR is enabled for an MPLS TE tunnel, once a link or node fails on the primary CRLSP, FRR reroutes the traffic to a bypass CRLSP, and the ingress node attempts to set up a new CRLSP. After the new CRLSP is set up, traffic is forwarded on the new CRLSP.

CRLSP backup provides end-to-end path protection for a CRLSP without time limitation. FRR provides quick but temporary protection for a link or node on a CRLSP.

Basic concepts

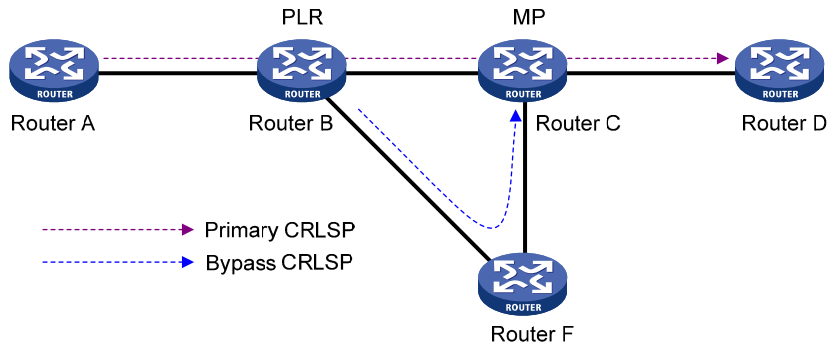
- **Primary CRLSP**—Protected CRLSP.
- **Bypass CRLSP**—Used to protect the primary CRLSP.
- **Point of local repair**—An PLR is the ingress node of the bypass CRLSP. It must be located on the primary CRLSP but must not be the egress node of the primary CRLSP.
- **Merge point**—An MP is the egress node of the bypass CRLSP. It must be located on the primary CRLSP but must not be the ingress node of the primary CRLSP.

Protection modes

FRR provides the following protection modes:

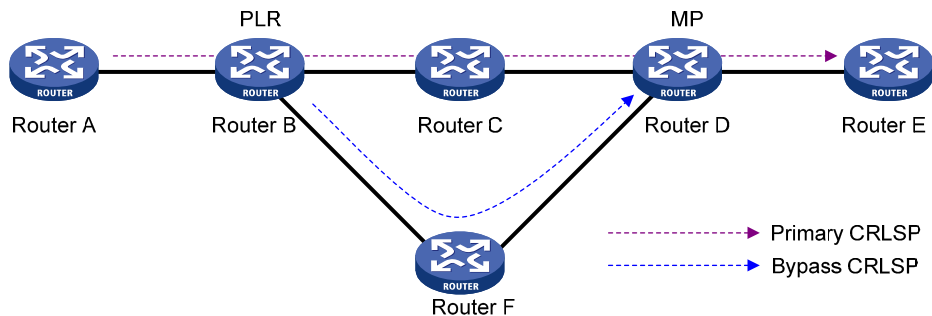
- **Link protection**—The PLR and the MP are connected through a direct link and the primary CRLSP traverses this link. When the link fails, traffic is switched to the bypass CRLSP. As shown in [Figure 20](#), the primary CRLSP is Router A—Router B—Router C—Router D, and the bypass CRLSP is Router B—Router F—Router C.

Figure 20 FRR link protection



- Node protection**—The PLR and the MP are connected through a device and the primary CRLSP traverses this device. When the device fails, traffic is switched to the bypass CRLSP. As shown in Figure 21, the primary CRLSP is Router A—Router B—Router C—Router D—Router E, and the bypass CRLSP is Router B—Router F—Router D. Router C is the protected device.

Figure 21 FRR node protection



FRR deployment

Following these guidelines to deploy FRR:

- Make sure the protected link or node is not on the bypass CRLSP.
- FRR requires extra bandwidth because bypass CRLSPs must be pre-established. When network bandwidth is insufficient, use FRR only for crucial nodes or links.

DiffServ-aware TE

DiffServ is a model that provides differentiated QoS guarantees based on class of service. MPLS TE is a traffic engineering solution that focuses on optimizing network resources allocation.

DiffServ-aware TE (DS-TE) combines DiffServ and TE to optimize network resources allocation on a per-service class basis. DS-TE defines different bandwidth constraints for class types. It maps each traffic class type to the CRLSP that is constraints compliant for the class type.

The device supports these DS-TE modes:

- Prestandard mode**—HP proprietary DS-TE.
- IETF mode**—Complies with RFC 4124, RFC 4125, and RFC 4127.

Basic concepts

- **CT**—Class Type. DS-TE allocates link bandwidth, implements constraint-based routing, and performs admission control on a per class type basis. A given traffic flow belongs to the same CT on all links.
- **BC**—Bandwidth Constraint. BC restricts the bandwidth for one or more CTs.
- **Bandwidth constraint model**—Algorithm for implementing bandwidth constraints on different CTs. A BC model comprises two factors, the maximum number of BCs (MaxBC) and the mappings between BCs and CTs. DS-TE supports two BC models, Russian Dolls Model (RDM) and Maximum Allocation Model (MAM).
- **TE class**—Defines a CT and a priority. The setup priority or holding priority of an MPLS TE tunnel for a CT must be the same as the priority of the TE class.

The prestandard and IETF modes of DS-TE have the following differences:

- The prestandard mode supports two CTs (CT 0 and CT 1), eight priorities, and up to 16 TE classes. The IETF mode supports four CTs (CT 0 through CT 3), eight priorities, and up to eight TE classes.
- The prestandard mode does not allow you to configure TE classes. The IETF mode allows for TE class configuration.
- The prestandard mode supports only RDM. The IETF mode supports both RDM and MAM.
- A device operating in prestandard mode cannot communicate with devices from some vendors. A device operating in IETF mode can communicate with devices from other vendors.

How DS-TE operates

A device takes the following steps to establish an MPLS TE tunnel for a CT:

1. Determines the CT.

A device classifies traffic according to your configuration:

- When configuring a dynamic MPLS TE tunnel, you can use the **mpls te bandwidth** command on the tunnel interface to specify a CT for the traffic to be forwarded by the tunnel.
- When configuring a static MPLS TE tunnel, you can use the **bandwidth** keyword to specify a CT for the traffic to be forwarded along the tunnel.

2. Checks whether bandwidth is enough for the CT.

You can use the **mpls te max-reservable-bandwidth** command on an interface to configure the bandwidth constraints of the interface. The device determines whether the bandwidth is enough to establish an MPLS TE tunnel for the CT.

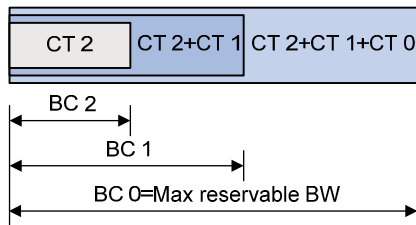
The relation between BCs and CTs varies with different BC models:

In RDM model, a BC constrains the total bandwidth of multiple CTs, as shown in [Figure 22](#):

- BC 2 is for CT 2. The total bandwidth for CT 2 cannot exceed BC 2.
- BC 1 is for CT 2 and CT 1. The total bandwidth for CT 2 and CT 1 cannot exceed BC 1.
- BC 0 is for CT 2, CT 1, and CT 0. The total bandwidth for CT 2, CT 1, and CT 0 cannot exceed BC 0. In this model, BC 0 equals the maximum reservable bandwidth of the link.

In cooperation with priority preemption, the RDM model can also implement bandwidth isolation between CTs. RDM is suitable for networks where traffic is unstable and traffic bursts might occur.

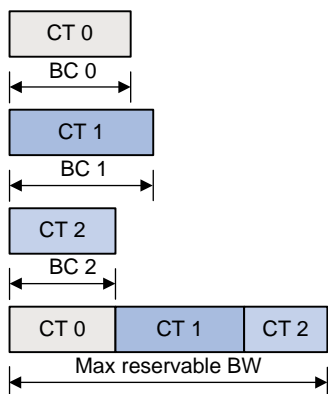
Figure 22 RDM bandwidth constraints model



In MAM model, a BC constrains the bandwidth for only one CT. This ensures bandwidth isolation among CTs no matter whether preemption is used or not. Compared with RDM, MAM is easier to configure. MAM is suitable for networks where traffic of each CT is stable and no traffic bursts occur. Figure 23 shows an example:

- BC 0 is for CT 0. The bandwidth occupied by the traffic of CT 0 cannot exceed BC 0.
- BC 1 is for CT 1. The bandwidth occupied by the traffic of CT 1 cannot exceed BC 1.
- BC 2 is for CT 2. The bandwidth occupied by the traffic of CT 2 cannot exceed BC 2.
- The total bandwidth occupied by CT 0, CT 1, and CT 2 cannot exceed the maximum reservable bandwidth.

Figure 23 MAM bandwidth constraints model



3. Checks whether the CT and the LSP setup/holding priority match an existing TE class.
An MPLS TE tunnel can be established for the CT only when the following conditions are met:
 - Every node along the tunnel has a TE class that matches the CT and the LSP setup priority.
 - Every node along the tunnel has a TE class that matches the CT and the LSP holding priority.

Bidirectional MPLS TE tunnel

MPLS Transport Profile (MPLS-TP) uses bidirectional MPLS TE tunnels to implement 1:1 and 1+1 protection switching and support in-band detection tools and signaling protocols such as OAM and PSC.

A bidirectional MPLS TE tunnel includes two CRLSPs in opposite directions. It can be established in the following modes:

- **Co-routed mode**—Uses the extended RSVP-TE protocol to establish a bidirectional MPLS TE tunnel. RSVP-TE uses a Path message to advertise the labels assigned by the upstream LSR to the downstream LSR and a Resv message to advertise the labels assigned by the downstream LSR to the upstream LSR. During the delivery of the path message, a CRLSP in one direction is established.

During the delivery of the Resv message, a CRLSP in the other direction is established. The CRLSPs of a bidirectional MPLS TE tunnel established in co-routed mode use the same path.

- **Associated mode**—In this mode, you establish a bidirectional MPLS TE tunnel by binding two unidirectional CRLSPs in opposite directions. The two CRLSPs can be established in different modes and use different paths. For example, one CRLSP is established statically and the other CRLSP is established dynamically by RSVP-TE.

For more information about establishing MPLS TE tunnel through RSVP-TE, the Path message, and the Resv message, see "Configuring RSVP."

Protocols and standards

- RFC 2702, *Requirements for Traffic Engineering Over MPLS*
- RFC 3564, *Requirements for Support of Differentiated Service-aware MPLS Traffic Engineering*
- RFC 4124, *Protocol Extensions for Support of Diffserv-aware MPLS Traffic Engineering*
- RFC 4125, *Maximum Allocation Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering*
- RFC 4127, *Russian Dolls Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering*
- ITU-T Recommendation Y.1720, *Protection switching for MPLS networks*

MPLS TE configuration task list

To configure an MPLS TE tunnel to use a static CRLSP, complete the following tasks:

1. Enable MPLS TE on each node and interface that the MPLS TE tunnel traverses.
2. Create a tunnel interface on the ingress node of the MPLS TE tunnel, and specify the tunnel destination address—the address of the egress node.
3. Create a static CRLSP on each node that the MPLS TE tunnel traverses.
For information about creating a static CRLSP, see "Configuring a static CRLSP."
4. On the ingress node of the MPLS TE tunnel, configure the tunnel interface to reference the created static CRLSP.
5. On the ingress node of the MPLS TE tunnel, configure static routing or PBR to direct traffic to the MPLS TE tunnel.

To configure an MPLS TE tunnel to use a CRLSP dynamically established by RSVP-TE, complete the following tasks:

6. Enable MPLS TE and RSVP on each node and interface that the MPLS TE tunnel traverses.
For information about enabling RSVP, see "Configuring RSVP."
7. Create a tunnel interface on the ingress node of the MPLS TE tunnel, specify the tunnel destination address—the address of the egress node, and configure the MPLS TE tunnel constraints (such as the tunnel bandwidth constraints and affinity) on the tunnel interface.
8. Configure the link TE attributes (such as the maximum link bandwidth and link attribute) on each interface that the MPLS TE tunnel traverses.
9. Configure an IGP on each node that the MPLS TE tunnel traverses, and configure the IGP to support MPLS TE, so that the nodes advertise the link TE attributes through the IGP.

10. On the ingress node of the MPLS TE tunnel, configure RSVP-TE to establish a CRLSP based on the tunnel constraints and link TE attributes.
11. On the ingress node of the MPLS TE tunnel, configure static routing or PBR to direct traffic to the MPLS TE tunnel.

You can also configure other MPLS TE functions such as the DS-TE and FRR as needed.

To configure MPLS TE, perform the following tasks:

Tasks at a glance
(Required.) Enabling MPLS TE
(Required.) Configuring a tunnel interface
(Optional.) Configuring DS-TE
(Required.) Perform at least one of the following tasks to configure an MPLS TE tunnel: <ul style="list-style-type: none"> • Configuring an MPLS TE tunnel to use a static CRLSP • Configuring an MPLS TE tunnel to use a dynamic CRLSP
(Required.) Perform either task to configure traffic forwarding: <ul style="list-style-type: none"> • Configuring static routing to direct traffic to an MPLS TE tunnel • Configuring PBR to direct traffic to an MPLS TE tunnel
(Optional.) Configuring a bidirectional MPLS TE tunnel
(Optional.) Configuring CRLSP backup Only MPLS TE tunnels established by RSVP-TE support this configuration.
(Optional.) Configuring MPLS TE FRR Only MPLS TE tunnels established by RSVP-TE support this configuration.

Enabling MPLS TE

Enable MPLS TE on each node and interface that the MPLS TE tunnel traverses.

Before you enable MPLS TE, complete the following tasks:

- Configure static routing or IGP to make sure all LSRs can reach each other.
- Configure basic MPLS. For information about basic MPLS configurations, see "Configuring basic MPLS."

To enable MPLS TE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable MPLS TE and enter MPLS TE view.	mpls te	By default, MPLS TE is disabled.
3. Return to system view.	quit	N/A
4. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A

Step	Command	Remarks
5. Enable MPLS TE for the interface.	mpls te enable	By default, MPLS TE is disabled on an interface.

Configuring a tunnel interface

To configure an MPLS TE tunnel, you must create an MPLS TE tunnel interface and enter tunnel interface view. All MPLS TE tunnel attributes are configured in tunnel interface view. For more information about tunnel interfaces, see *Layer 3—IP Services Configuration Guide*.

Perform this task on the ingress node of the MPLS TE tunnel.

To configure a tunnel interface:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create an MPLS TE tunnel interface and enter tunnel interface view.	interface tunnel <i>tunnel-number</i> mode mpls-te	By default, no tunnel interface is created.
3. Configure an IP address for the tunnel interface.	ip address <i>ip-address</i> { <i>mask-length</i> <i>mask</i> }	By default, a tunnel interface does not have an IP address.
4. Specify the tunnel destination address.	destination <i>ip-address</i>	By default, no tunnel destination address is specified.

Configuring DS-TE

DS-TE is configurable on any node that an MPLS TE tunnel traverses.

To configure DS-TE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter MPLS TE view.	mpls te	N/A
3. (Optional.) Configure the DS-TE mode as IETF.	ds-te mode ietf	By default, the DS-TE mode is prestandard .
4. (Optional.) Configure the BC model of IETF DS-TE as MAM.	ds-te bc-model mam	By default, the BC model of IETF DS-TE is RDM.
5. Configure a TE class.	ds-te te-class <i>te-class-index</i> class-type <i>class-type-number</i> priority <i>pri-number</i>	The default TE classes for IETF mode are shown in Table 1 . In prestandard mode, you cannot configure TE classes.

Table 1 Default TE classes in IETF mode

TE Class	CT	Priority
0	0	7
1	1	7
2	2	7
3	3	7
4	0	0
5	1	0
6	2	0
7	3	0

Configuring an MPLS TE tunnel to use a static CRLSP

To configure an MPLS TE tunnel to use a static CRLSP, establish the static CRLSP, specify the MPLS TE tunnel establishment mode as static, and configure the MPLS TE tunnel to reference the static CRLSP. Other configurations, such as tunnel constraints, IGP extension and CSPF, are not needed.

To configure an MPLS TE tunnel to use a static CRLSP:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a static CRLSP.	See "Configuring a static CRLSP."	N/A
3. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	Execute this command on the ingress node.
4. Specify the MPLS TE tunnel establishment mode as static.	mpls te signaling static	By default, MPLS TE uses RSVP-TE to establish a tunnel.
5. Apply the static CRLSP to the tunnel interface.	mpls te static-cr-lsp <i>lsp-name</i>	By default, a tunnel does not reference any static CRLSP.

Configuring an MPLS TE tunnel to use a dynamic CRLSP

To configure an MPLS TE tunnel to use a CRLSP dynamically established by RSVP-TE, complete the following tasks:

- Configure MPLS TE attributes for the links.
- Configure IGP TE extension to advertise link TE attributes, so as to generate a TEDB on each node.
- Configure tunnel constraints.
- Use the CSPF to calculate a preferred path based on the TEDB and tunnel constraints.
- Establish the CRLSP by using the signaling protocol RSVP-TE.

You must configure the IGP TE extension to form a TEDB. Otherwise, the path is created based on IGP routing rather than computed by CSPF.

Configuration task list

To establish an MPLS TE tunnel by using a dynamic CRLSP, perform the following tasks:

Tasks at a glance
(Required.) Configuring MPLS TE attributes for a link
(Required.) Configuring MPLS TE tunnel constraints
(Required.) Establishing an MPLS TE tunnel by using RSVP-TE
(Optional.) Controlling CRLSP path selection
(Optional.) Controlling MPLS TE tunnel setup

Configuring MPLS TE attributes for a link

MPLS TE attributes for a link include the maximum link bandwidth, the maximum reservable bandwidth, and the link attribute.

Perform this task on each interface that the MPLS TE tunnel traverses.

To configure the link TE attributes:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type interface-number</i>	N/A
3. Configure the maximum link bandwidth for MPLS TE traffic.	mpls te max-link-bandwidth <i>bandwidth-value</i>	By default, the maximum link bandwidth for MPLS TE traffic is 0.

Step	Command	Remarks
4. Configure the maximum reservable bandwidth.	<ul style="list-style-type: none"> Configure the maximum reservable bandwidth of the link (BC 0) and BC 1 in RDM model of the prestandard DS-TE: mpls te max-reservable-bandwidth <i>bandwidth-value [bc1 bc1-bandwidth]</i> Configure the maximum reservable bandwidth of the link and the BCs in MAM model of the IETF DS-TE: mpls te max-reservable-bandwidth mam <i>bandwidth-value { bc0 bc0-bandwidth bc1 bc1-bandwidth bc2 bc2-bandwidth bc3 bc3-bandwidth } *</i> Configure the maximum reservable bandwidth of the link and the BCs in RDM model of the IETF DS-TE: mpls te max-reservable-bandwidth rdm <i>bandwidth-value [bc1 bc1-bandwidth] [bc2 bc2-bandwidth] [bc3 bc3-bandwidth]</i> 	<p>Use one command according to the DS-TE mode and BC model configured in "Configuring DS-TE."</p> <p>By default, the maximum reservable bandwidth of a link is 0 kbps and each BC is 0 kbps.</p> <p>In RDM model, BC 0 is the maximum reservable bandwidth of a link.</p>
5. Configure the link attribute.	mpls te link-attribute <i>attribute-value</i>	By default, the link attribute value is 0x00000000.

Configuring MPLS TE tunnel constraints

Perform this task on the ingress node of the MPLS TE tunnel.

Configuring bandwidth constraints for an MPLS TE tunnel

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	N/A
3. Configure bandwidth required for the tunnel, and specify a CT for the tunnel's traffic.	mpls te bandwidth <i>[ct0 ct1 ct2 ct3] bandwidth</i>	By default, no bandwidth is assigned, and the class type is CT 0.

Configuring the affinity attribute for an MPLS TE tunnel

The associations between the link attribute and the affinity attribute might vary by vendor. To ensure the successful establishment of a tunnel between two devices from different vendors, correctly configure their respective link attribute and affinity attribute.

To configure the affinity attribute for an MPLS TE tunnel:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	N/A

Step	Command	Remarks
3. Configure an affinity for the MPLS TE tunnel.	mpls te affinity-attribute <i>attribute-value</i> [mask <i>mask-value</i>]	By default, the affinity is 0x00000000, and the mask is 0x00000000. The default affinity matches all link attributes.

Configuring a setup priority and a holding priority for an MPLS TE tunnel

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode <i>mpls-te</i>]	N/A
3. Configure a setup priority and a holding priority for the MPLS TE tunnel.	mpls te priority <i>setup-priority</i> [<i>hold-priority</i>]	By default, the setup priority and the holding priority are both 7 for an MPLS TE tunnel.

Configuring an explicit path for an MPLS TE tunnel

An explicit path is a set of nodes. The relationship between any two neighboring nodes on an explicit path can be either strict or loose.

- **Strict**—The two nodes must be directly connected.
- **Loose**—The two nodes can have devices in between.

When establishing an MPLS TE tunnel between areas or ASs, you must use a loose explicit path, specify the ABR or ASBR as the next hop of the path, and make sure the tunnel's ingress node and the ABR or ASBR can reach each other.

Configure an explicit path for a MPLS TE tunnel:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create an explicit path and enter its view.	explicit-path <i>path-name</i>	By default, no explicit path exists on the device.
3. Enable the explicit path.	undo disable	By default, an explicit path is enabled.
4. Add or modify a node in the explicit path.	nexthop [index <i>index-number</i>] <i>ip-address</i> [exclude include] [loose strict]	By default, an explicit path does not include any node. You can specify the include keyword to have the CRLSP traverse the specified node or the exclude keyword to have the CRLSP bypass the specified node.
5. Return to system view.	quit	N/A
6. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode <i>mpls-te</i>]	N/A

Step	Command	Remarks
7. Configure the MPLS TE tunnel interface to use the explicit path, and specify a preference value for the explicit path.	mpls te path preference <i>value</i> explicit-path <i>path-name</i>	By default, MPLS TE uses the calculated path to establish a CRLSP.

Establishing an MPLS TE tunnel by using RSVP-TE

Before you configure this task, you must use the **rsvp** command and the **rsvp enable** command to enable RSVP on all nodes and interfaces that the MPLS TE tunnel traverses.

Perform this task on the ingress node of the MPLS TE tunnel.

To configure RSVP-TE to establish an MPLS TE tunnel:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	N/A
3. Configure MPLS TE to use RSVP-TE to establish the tunnel.	mpls te signaling rsvp-te	By default, MPLS TE uses RSVP-TE to establish a tunnel.
4. Specify an explicit path for the MPLS TE tunnel, and specify the path preference value.	mpls te path preference <i>value</i> { dynamic explicit-path <i>path-name</i> }	By default, MPLS TE uses the calculated path to establish a CRLSP.

Controlling CRLSP path selection

Before performing the configuration tasks in this section, be aware of each configuration objective and its impact on your device.

MPLS TE uses CSPF to calculate a path according to the TEDB and constraints and sets up the CRLSP through RSVP-TE. MPLS TE provides measures that affect the CSPF calculation. You can use these measures to tune the path selection for CRLSP.

Configuring the metric type for path selection

Each MPLS TE link has a metric. The IGP metric represents a link delay (a smaller IGP metric value indicates a lower link delay). By using the IGP metric for MPLS TE links, you can make sure delay-sensitive traffic such as voice traffic travels through the path that has lower delay.

To configure the metric type for tunnel path selection:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter MPLS TE view.	mpls te	N/A

Step	Command	Remarks
3. Specify the metric type to use when no metric type is explicitly configured for a tunnel.	path-metric-type igp	Execute this command on the ingress node of an MPLS TE tunnel.
4. Return to system view.	quit	N/A
5. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	N/A
6. Specify the metric type for path selection.	mpls te path-metric-type igp	By default, no link metric type is specified and the one specified in MPLS TE view is used. Execute this command on the ingress node of an MPLS TE tunnel.
7. Return to system view.	quit	N/A

Configuring route pinning

When route pinning is enabled, MPLS TE tunnel reoptimization is not available.

Perform this task on the ingress node of an MPLS TE tunnel.

To configure route pinning:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	N/A
3. Enable route pinning.	mpls te route-pinning	By default, route pinning is disabled.

Configuring tunnel reoptimization

Tunnel reoptimization allows you to manually or dynamically trigger the ingress node to recalculate a path. If the ingress node recalculates a better path, it creates a new CRLSP, switches the traffic from the old CRLSP to the new CRLSP, and then deletes the old CRLSP.

Perform this task on the ingress node of an MPLS TE tunnel.

To configure tunnel reoptimization:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	N/A
3. Enable tunnel reoptimization.	mpls te reoptimization [frequency <i>seconds</i>]	By default, tunnel reoptimization is disabled.
4. Return to user view.	return	N/A

Step	Command	Remarks
5. (Optional.) Immediately reoptimize all MPLS TE tunnels that are enabled with the tunnel reoptimization function.	mpls te reoptimization	N/A

Configuring TE flooding thresholds and interval

When the bandwidth of an MPLS TE link changes, IGP floods the new bandwidth information, so the ingress node can use CSPF to recalculate the path.

To prevent such recalculations from consuming too many resources, you can configure IGP to flood only significant bandwidth changes by setting the following flooding thresholds:

- **Up threshold**—When the percentage of the reservable-bandwidth increase to the maximum reservable bandwidth reaches the threshold, IGP floods the TE information.
- **Down threshold**—When the percentage of the reservable-bandwidth decrease to the maximum reservable bandwidth reaches the threshold, IGP floods the TE information.

You can also configure the flooding interval at which bandwidth changes that cannot trigger immediate flooding are flooded.

This task can be performed on all nodes that the MPLS TE tunnel traverses.

To configure TE flooding thresholds and the flooding interval:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Configure the up/down threshold.	mpls te bandwidth change thresholds { down up } <i>percent</i>	By default, the up/down threshold is 10% of the link reservable bandwidth.
4. Return to system view.	quit	N/A
5. Enter MPLS TE view.	mpls te	N/A
6. Configure the flooding interval.	link-management periodic-flooding timer <i>interval</i>	By default, the flooding interval is 180 seconds.

Controlling MPLS TE tunnel setup

Before performing the configuration tasks in this section, be aware of each configuration objective and its impact on your device.

Perform the tasks in this section on the ingress node of the MPLS TE tunnel.

Enabling route and label recording

Perform this task to record the nodes that an MPLS TE tunnel traverses and the label assigned by each node. The recorded information helps you know about the path used by the MPLS TE tunnel and the label distribution information, and when the tunnel fails, it helps you locate the fault.

To enable route and label recording:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	N/A
3. Record routes or record both routes and labels.	<ul style="list-style-type: none"> To record routes: mpls te record-route To record both routes and labels: mpls te record-route label 	By default, both route recording and label recording are disabled.

Enabling loop detection

Enabling loop detection also enables the route recording function, regardless of whether you have configured the **mpls te record-route** command. Loop detection enables each node of the tunnel to detect whether a loop has occurred according to the recorded route information.

To enable loop detection:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	N/A
3. Enable loop detection.	mpls te loop-detection	By default, loop detection is disabled.

Configuring tunnel setup retry

If the ingress node fails to establish an MPLS TE tunnel, it waits for the retry interval, and then tries to set up the tunnel again. It repeats this process until the tunnel is established or until the number of attempts reaches the maximum. If the tunnel cannot be established when the number of attempts reaches the maximum, the ingress waits for a longer period and then repeats the previous process.

To configure tunnel setup retry:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	N/A
3. Configure maximum number of tunnel setup attempts.	mpls te retry <i>times</i>	By default, the maximum number of attempts is 3.
4. Configure the retry interval.	mpls te timer retry <i>seconds</i>	By default, the retry interval is 2 seconds.

Configuring RSVP resource reservation style

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	N/A

Step	Command	Remarks
3. Configure the resources reservation style for the tunnel.	mpls te resv-style { ff se }	By default, the resource reservation style is SE. In current MPLS TE applications, tunnels are established usually by using the make-before-break mechanism. Therefore, HP recommends that you use the SE style.

Configuring traffic forwarding

Perform the tasks in this section on the ingress node of the MPLS TE tunnel.

Configuring static routing to direct traffic to an MPLS TE tunnel

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure a static route to direct traffic to an MPLS TE tunnel.	For information about static routing commands, see <i>Layer 3—IP Routing Command Reference</i> .	By default, no static route exists on the device. The interface specified in this command must be an MPLS TE tunnel interface.

Configuring PBR to direct traffic to an MPLS TE tunnel

For more information about the commands in this task, see *Layer 3—IP Routing Command Reference*.

To configure PBR to direct traffic to an MPLS TE tunnel:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a PBR policy node and enter policy node view.	policy-based-route <i>policy-name</i> [deny permit] node <i>node-number</i>	By default, no PBR policy node is created.
3. Configure an ACL match criterion.	if-match acl { <i>acl-number</i> name <i>acl-name</i> }	By default, no ACL match criterion is configured.
4. Specify a tunnel interface as the packet output interface.	apply output-interface { tunnel <i>tunnel-number</i> [track <i>track-entry-number</i>] }<1-n>	N/A
5. Return to system view.	quit	N/A

Step	Command	Remarks
6. Apply the PBR policy.	<ul style="list-style-type: none"> To apply the policy to the local device: ip local policy-based-route <i>policy-name</i> To apply the policy to an interface: <ul style="list-style-type: none"> interface <i>interface-type interface-number</i> ip policy-based-route <i>policy-name</i> 	<p>Use either method.</p> <p>By default, no policy is applied.</p>

Configuring a bidirectional MPLS TE tunnel

Before you create a bidirectional MPLS TE tunnel, complete the following tasks:

- Disable the PHP function on both ends of the tunnel to assign a non-null label to the penultimate hop.
- To set up a bidirectional MPLS TE tunnel in co-routed mode, you must specify the signaling protocol as RSVP-TE, and use the **mpls te resv-style** command to configure the resources reservation style as FF for the tunnel.
- To set up a bidirectional MPLS TE tunnel in associated mode and use RSVP-TE to set up one CRLSP of the tunnel, you must use the **mpls te resv-style** command to configure the resources reservation style as FF for the CR-LSP.

To create a bidirectional MPLS TE tunnel, create an MPLS TE tunnel interface on both ends of the tunnel and enable the bidirectional tunnel function on the tunnel interfaces:

- For a co-routed bidirectional tunnel, configure one end of the tunnel as the active end and the other end as the passive end, and specify the reverse CR-LSP at the passive end.
- For an associated bidirectional tunnel, specify a reverse CR-LSP at both ends of the tunnel.

To configure the active end of a co-routed bidirectional MPLS TE tunnel:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	N/A
3. Configure a co-routed bidirectional MPLS TE tunnel and specify the local end as the active end of the tunnel.	mpls te bidirectional co-routed active	By default, no bidirectional tunnel is configured, and tunnels established on the tunnel interface are unidirectional MPLS TE tunnels.

To configure the passive end of a co-routed bidirectional MPLS TE tunnel:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	N/A
3. Configure a co-routed bidirectional MPLS TE tunnel and specify the local end as the passive end of the tunnel.	mpls te bidirectional co-routed passive reverse-lsp <i>lsp-id</i> <i>ingress-lsp-id tunnel-id tunnel-id</i>	By default, no bidirectional tunnel is configured, and tunnels established on the tunnel interface are unidirectional MPLS TE tunnels.

To configure an associated bidirectional MPLS TE tunnel:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	N/A
3. Configure an associated bidirectional MPLS TE tunnel.	mpls te bidirectional associated reverse-lsp { lsp-name <i>lsp-name</i> lsp-id <i>ingress-lsp-id</i> tunnel-id <i>tunnel-id</i> }	By default, no bidirectional tunnel is configured, and tunnels established on the tunnel interface are unidirectional MPLS TE tunnels.

Configuring CRLSP backup

CRLSP backup provides end-to-end CRLSP protection. Only MPLS TE tunnels established through RSVP-TE support CRLSP backup.

Perform this task on the ingress node of an MPLS TE tunnel.

To configure CRLSP backup:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter MPLS TE tunnel interface view.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	N/A
3. Enable CRLSP backup and specify the backup mode.	mpls te backup { hot-standby ordinary }	By default, tunnel backup is disabled.
4. Specify a path for the primary CRLSP and set the preference of the path.	mpls te path preference <i>value</i> { dynamic explicit-path <i>path-name</i> }	By default, MPLS TE uses the dynamically calculated path to set up the primary CRLSP.
5. Specify a path for the backup CRLSP and set the preference of the path.	mpls te backup-path preference <i>value</i> explicit-path <i>path-name</i>	N/A

Configuring MPLS TE FRR

MPLS TE FRR provides temporary link or node protection on a CRLSP. When you configure FRR, note the following restrictions and guidelines:

- Do not configure both FRR and RSVP authentication on the same interface.
- Only MPLS TE tunnels established through RSVP-TE support FRR.
- Use bypass tunnels to protect only crucial interfaces or links when bandwidth is insufficient because bypass tunnels are pre-established and require extra bandwidth.
- You can specify which type of CRLSPs can use a bypass tunnel, whether a bypass tunnel provides bandwidth protection, and the sum of protected bandwidth. Make sure the bandwidth assigned to the bypass tunnel is no less than the total bandwidth needed by all primary CRLSPs. Otherwise, some primary CRLSPs might not be protected by the bypass tunnel.

- Usually, a bypass tunnel does not forward data when the primary CRLSP works. For a bypass tunnel to also forward data during tunnel protection, you must assign adequate bandwidth to the bypass tunnel.
- A bypass tunnel cannot be used for services like VPN.

Enabling FRR

Perform this task on the ingress node of a primary CRLSP.

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter tunnel interface view of the primary CRLSP.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	N/A
3. Enable FRR.	mpls te fast-reroute	By default, FRR is disabled.

Configuring a bypass tunnel on the PLR

To configure FRR, you must create an MPLS TE tunnel on the PLR, specify the bandwidth and CRLSP types that the tunnel can protect, and bind the tunnel to the egress interface of the primary CRLSP. This MPLS TE tunnel serves as the bypass tunnel of the primary CRLSP. When the link or node connected to the egress interface fails, MPLS TE switches the traffic to the bypass tunnel to ensure uninterrupted traffic forwarding.

The bypass tunnel setup method is the same as a normal MPLS TE tunnel. This section describes only FRR-related configurations.

When configuring a bypass tunnel, follow these guidelines:

- You cannot configure FRR for a bypass tunnel. A bypass tunnel cannot act as a primary CRLSP.
- The protected interface cannot be the egress interface of the bypass tunnel.
- You can specify up to three bypass tunnels for a protected interface. If multiple bypass tunnels are specified, the one that can provide node protection is preferred.
- Make sure the bandwidth assigned to the bypass tunnel is no less than the total bandwidth needed by all primary CRLSPs. Otherwise, some primary CRLSPs might not be protected by the bypass tunnel.

To configure a bypass tunnel on the PLR:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter tunnel interface view of the bypass tunnel.	interface tunnel <i>tunnel-number</i> [mode mpls-te]	N/A
3. Specify the destination address of the bypass tunnel.	destination <i>ip-address</i>	The bypass tunnel destination address is the LSR ID of the MP.

Step	Command	Remarks
4. Configure the bandwidth and the CT that the bypass tunnel can protect.	mpls te backup bandwidth { <i>bandwidth</i> { ct0 ct1 ct2 ct3 } { <i>bandwidth</i> un-limited } }	By default, the bypass tunnel does not provide bandwidth protection. You must execute this command to configure the bandwidth that the bypass tunnel can protect. Otherwise, the primary CRLSP cannot be bound to the bypass tunnel successfully.
5. Return to system view.	quit	N/A
6. Enter interface view of the egress interface of the primary CRLSP.	interface <i>interface-type</i> <i>interface-number</i>	N/A
7. Specify a bypass tunnel for the protected interface (the current interface).	rsvp fast-reroute bypass-tunnel tunnel <i>tunnel-number</i>	By default, no bypass tunnel is specified for an interface.

Configuring node fault detection

Perform this task to configure the RSVP hello mechanism or BFD on the PLR and the protected node to detect the node faults caused by signaling protocol faults. FRR does not need to use the RSVP hello mechanism or BFD to detect the node faults caused by the link faults between the PLR and the protected node.

You do not need to perform this task for FRR link protection.

To configure node fault detection:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view of the connecting interface between the PLR and the protected node.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Configure node fault detection.	<ul style="list-style-type: none"> (Method 1) Enable RSVP hello extension on the interface: rsvp hello enable (Method 2) Enable BFD on the interface: rsvp bfd enable 	Use either method. By default, RSVP hello extension is disabled, and BFD is not configured. For more information about the rsvp hello enable command and the rsvp bfd enable command, see "Configuring RSVP."

Configuring the optimal bypass tunnel selection interval

If you have specified multiple bypass tunnels for a primary CRLSP, RSVP selects an optimal bypass tunnel to protect the primary CRLSP. Sometimes, a bypass tunnel might become better than the current optimal

bypass tunnel because, for example, the reservable bandwidth changes. Therefore, RSVP needs to poll the bypass tunnels periodically to update the optimal bypass tunnel.

You can perform this task on the PLR to configure the interval for selecting an optimal bypass tunnel:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter RSVP view.	rsvp	N/A
3. Configure the interval for selecting an optimal bypass tunnel.	fast-reroute timer <i>interval</i>	By default, the interval is 300 seconds.

Displaying and maintaining MPLS TE

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display information about explicit paths.	display explicit-path [<i>path-name</i>]
Display DS-TE information.	display mpls te ds-te
Display the bandwidth information on MPLS TE-enabled interfaces.	display mpls te link-management bandwidth-allocation [interface <i>interface-type interface-number</i>]
Display information about MPLS TE tunnel interfaces.	display mpls te tunnel-interface [tunnel <i>number</i>]

MPLS TE configuration examples

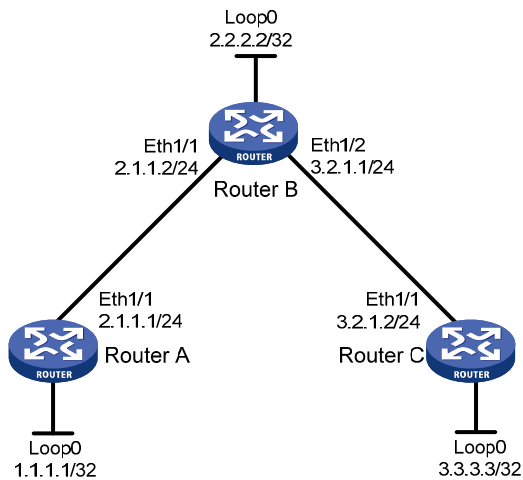
Establishing an MPLS TE tunnel over a static CRLSP

Network requirements

Router A, Router B, and Router C run IS-IS.

Establish an MPLS TE tunnel over a static CRLSP from Router A to Router C.

Figure 24 Network diagram



Configuration procedure

1. Configure IP addresses and masks for interfaces. (Details not shown.)
2. Configure IS-IS to advertise interface addresses, including the Loopback interface address:

Configure Router A.

```
<RouterA> system-view
[RouterA] isis 1
[RouterA-isis-1] network-entity 00.0005.0000.0000.0001.00
[RouterA-isis-1] quit
[RouterA] interface ethernet 1/1
[RouterA-Ethernet1/1] isis enable 1
[RouterA-Ethernet1/1] quit
[RouterA] interface loopback 0
[RouterA-LoopBack0] isis enable 1
[RouterA-LoopBack0] quit
```

Configure Router B.

```
<RouterB> system-view
[RouterB] isis 1
[RouterB-isis-1] network-entity 00.0005.0000.0000.0002.00
[RouterB-isis-1] quit
[RouterB] interface ethernet 1/1
[RouterB-Ethernet1/1] isis enable 1
[RouterB-Ethernet1/1] quit
[RouterB] interface ethernet 1/2
[RouterB-Ethernet1/2] isis enable 1
[RouterB-Ethernet1/2] quit
[RouterB] interface loopback 0
[RouterB-LoopBack0] isis enable 1
[RouterB-LoopBack0] quit
```

Configure Router C.

```
<RouterC> system-view
[RouterC] isis 1
```

```

[RouterC-isis-1] network-entity 00.0005.0000.0000.0003.00
[RouterC-isis-1] quit
[RouterC] interface ethernet 1/1
[RouterC-Ethernet1/1] isis enable 1
[RouterC-Ethernet1/1] quit
[RouterC] interface loopback 0
[RouterC-LoopBack0] isis enable 1
[RouterC-LoopBack0] quit

```

After the previous configuration, execute the **display ip routing-table** command on each router. You can see that the routers have learned the routes to one another, including the routes to the Loopback interfaces.

3. Configure an LSR ID, and enable MPLS and MPLS TE:

Configure Router A.

```

[RouterA] mpls lsr-id 1.1.1.1
[RouterA] mpls te
[RouterA-te] quit
[RouterA] interface ethernet 1/1
[RouterA-Ethernet1/1] mpls enable
[RouterA-Ethernet1/1] mpls te enable
[RouterA-Ethernet1/1] quit

```

Configure Router B.

```

[RouterB] mpls lsr-id 2.2.2.2
[RouterB] mpls te
[RouterB-te] quit
[RouterB] interface ethernet 1/1
[RouterB-Ethernet1/1] mpls enable
[RouterB-Ethernet1/1] mpls te enable
[RouterB-Ethernet1/1] quit
[RouterB] interface ethernet 1/2
[RouterB-Ethernet1/2] mpls enable
[RouterB-Ethernet1/2] mpls te enable
[RouterB-Ethernet1/2] quit

```

Configure Router C.

```

[RouterC] mpls lsr-id 3.3.3.3
[RouterC] mpls te
[RouterC-mpls] quit
[RouterC] interface ethernet 1/1
[RouterC-Ethernet1/1] mpls enable
[RouterC-Ethernet1/1] mpls te enable
[RouterC-Ethernet1/1] quit

```

4. Configure an MPLS TE tunnel:

On Router A, configure the MPLS TE tunnel interface Tunnel 0, specify the tunnel destination address as the LSR ID of Router C, and configure MPLS TE to use a static CRLSP to establish the tunnel.

```

[RouterA] interface tunnel 0 mode mpls-te
[RouterA-Tunnel0] ip address 6.1.1.1 255.255.255.0
[RouterA-Tunnel0] destination 3.3.3.3

```

```
[RouterA-Tunnel0] mpls te signaling static
[RouterA-Tunnel0] quit
```

5. Create a static CRLSP:

Configure Router A as the ingress node of the static CRLSP, and specify the next hop address as 2.1.1.2 and outgoing label as 20.

```
[RouterA] static-cr-lsp ingress static-cr-lsp-1 nexthop 2.1.1.2 out-label 20
```

On Router A, configure tunnel 0 to reference the static CRLSP **static-cr-lsp-1**.

```
[RouterA] interface tunnel0
[RouterA-Tunnel0] mpls te static-cr-lsp static-cr-lsp-1
[RouterA-Tunnel0] quit
```

Configure Router B as the transit node of the static CRLSP, and specify the incoming label as 20, the next hop address as 3.2.1.2, and outgoing label as 30.

```
[RouterB] static-cr-lsp transit static-cr-lsp-1 in-label 20 nexthop 3.2.1.2 out-label 30
```

Configure Router C as the egress node of the static CRLSP, and specify the incoming label as 30.

```
[RouterC] static-cr-lsp egress static-cr-lsp-1 in-label 30
```

6. Configure a static route on Router A to direct traffic destined for subnet 3.2.1.0/24 to MPLS TE tunnel 0:

```
[RouterA] ip route-static 3.2.1.2 24 tunnel 0 preference 1
```

Verifying the configuration

Execute the **display interface tunnel** command on Router A. The output shows that the tunnel interface is up.

```
[RouterA] display interface tunnel
Tunnel0
Current state: UP
Line protocol state: UP
Description: Tunnel0 Interface
Maximum Transmit Unit: 64000
Internet Address is 6.1.1.1/24 Primary
Tunnel source unknown, destination 3.3.3.3
Tunnel bandwidth 64 (kbps)
Tunnel TTL 255
Tunnel protocol/transport CR_LSP
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops
```

Execute the **display mpls te tunnel-interface** command on Router A to display detailed information about the MPLS TE tunnel.

```
[RouterA] display mpls te tunnel-interface
Tunnel Name           : Tunnel 0
Tunnel State          : Up (Main CRLSP up)
Tunnel Attributes     :
  LSP ID              : 1           Tunnel ID           : 0
  Admin State         : Normal
```

```

Ingress LSR ID      : 1.1.1.1          Egress LSR ID      : 3.3.3.3
Signaling           : Static           Static CRLSP Name   : static-cr-lsp-1
Resv Style          : -
Tunnel mode         : -
Reverse-LSP name    : -
Reverse-LSP LSR ID : -                Reverse-LSP Tunnel ID: -
Class Type          : -                Tunnel Bandwidth    : -
Reserved Bandwidth  : -
Setup Priority       : 0                Holding Priority     : 0
Affinity Attr/Mask  : -/-
Explicit Path       : -
Backup Explicit Path : -
Metric Type         : IGP
Record Route        : -                Record Label        : -
FRR Flag            : -                Backup Bandwidth Flag: -
Backup Bandwidth Type: -              Backup Bandwidth    : -
Route Pinning       : -
Retry Limit         : 10               Retry Interval       : 2 sec
Reoptimization      : -                Reoptimization Freq : -
Backup Type         : -                Backup LSP ID       : -
Auto Bandwidth      : -                Auto Bandwidth Freq : -
Min Bandwidth       : -                Max Bandwidth       : -
Collected Bandwidth : -

```

Execute the **display mpls lsp** command or the **display mpls static-cr-lsp** command on each router to display the static CRLSP information.

```
[RouterA] display mpls lsp
```

```

FEC                Proto    In/Out Label    Interface/Out NHLFE
1.1.1.1/0/1        StaticCR -/20        Eth1/1
2.1.1.2            Local    -/-            Eth1/1

```

```
[RouterB] display mpls lsp
```

```

FEC                Proto    In/Out Label    Interface/Out NHLFE
-                  StaticCR 20/30        Eth1/2
3.2.1.2            Local    -/-            Eth1/2

```

```
[RouterC] display mpls lsp
```

```

FEC                Proto    In/Out Label    Interface/Out NHLFE
-                  StaticCR 30/-         -

```

```
[RouterA] display mpls static-cr-lsp
```

```

Name          LSR Type    In/Out Label    Out Interface    State
static-cr-lsp-1 Ingress    Null/20        Eth1/1           Up

```

```
[RouterB] display mpls static-cr-lsp
```

```

Name          LSR Type    In/Out Label    Out Interface    State
static-cr-lsp-1 Transit    20/30         Eth1/2           Up

```

```
[RouterC] display mpls static-cr-lsp
```

```

Name          LSR Type    In/Out Label    Out Interface    State
static-cr-lsp-1 Egress     30/Null        -                Up

```

Execute the **display ip routing-table** command on Router A. You can see a static route entry with interface Tunnel0 as the egress interface.

Establishing an inter-AS MPLS TE tunnel with RSVP-TE

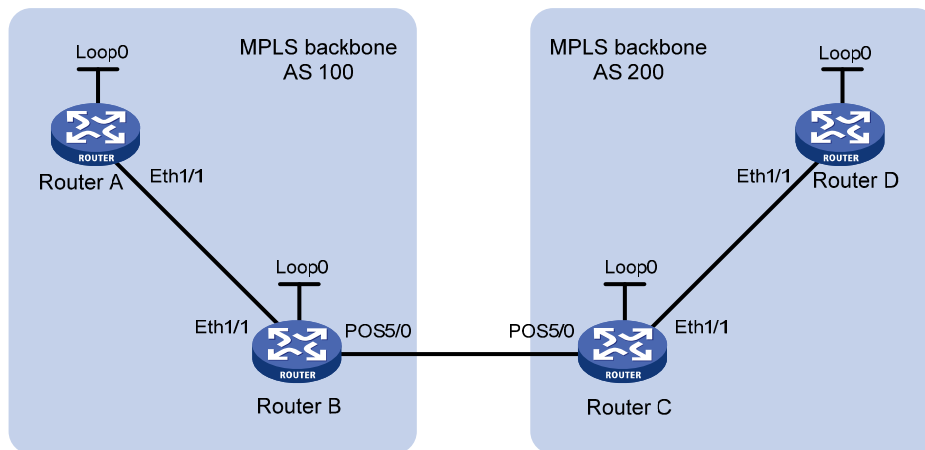
Network requirements

Router A and Router B are in AS 100. Router C and Router D are in AS 200. AS 100 and AS 200 use OSPF as the IGP.

Establish an EBGP connection between ASBRs Router B and Router C. Redistribute BGP routes into OSPF and OSPF routes into BGP, so that AS 100 and AS 200 can reach each other.

Establish an MPLS TE tunnel from Router A to Router D. The tunnel requires a bandwidth of 2000 kbps. The maximum bandwidth of the link that the tunnel traverses is 10000 kbps, and the maximum reservable bandwidth of the link is 5000 kbps.

Figure 25 Network diagram



Device	Interface	IP address	Device	Interface	IP address
Router A	Loop0	1.1.1.9/32	Router C	Loop0	3.3.3.9/32
	Eth1/1	10.1.1.1/24		Eth1/1	30.1.1.1/24
Router B	Loop0	2.2.2.9/32	Router D	Loop0	4.4.4.9/32
	Eth1/1	10.1.1.2/24		Eth1/1	30.1.1.2/24
	POS5/0	20.1.1.1/24			

Configuration procedure

1. Configure IP addresses and masks for interfaces. (Details not shown.)
2. Configure OSPF to advertise routes within the ASs, and redistribute the direct and BGP routes into OSPF on Router B and Router C:

Configure Router A.

```
<RouterA> system-view
[RouterA] ospf
[RouterA-ospf-1] area 0
[RouterA-ospf-1-area-0.0.0.0] network 10.1.1.0 0.0.0.255
[RouterA-ospf-1-area-0.0.0.0] network 1.1.1.9 0.0.0.0
[RouterA-ospf-1-area-0.0.0.0] quit
[RouterA-ospf-1] quit
```

Configure Router B.

```
<RouterB> system-view
```

```
[RouterB] ospf
[RouterB-ospf-1] import-route direct
[RouterB-ospf-1] import-route bgp
[RouterB-ospf-1] area 0
[RouterB-ospf-1-area-0.0.0.0] network 10.1.1.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] network 2.2.2.9 0.0.0.0
[RouterB-ospf-1-area-0.0.0.0] quit
[RouterB-ospf-1] quit
```

Configure Router C.

```
<RouterC> system-view
[RouterC] ospf
[RouterC-ospf-1] import-route direct
[RouterC-ospf-1] import-route bgp
[RouterC-ospf-1] area 0
[RouterC-ospf-1-area-0.0.0.0] network 30.1.1.0 0.0.0.255
[RouterC-ospf-1-area-0.0.0.0] network 3.3.3.9 0.0.0.0
[RouterC-ospf-1-area-0.0.0.0] quit
[RouterC-ospf-1] quit
```

Configure Router D.

```
<RouterD> system-view
[RouterD] ospf
[RouterD-ospf-1] area 0
[RouterD-ospf-1-area-0.0.0.0] network 30.1.1.0 0.0.0.255
[RouterD-ospf-1-area-0.0.0.0] network 4.4.4.9 0.0.0.0
[RouterD-ospf-1-area-0.0.0.0] quit
[RouterD-ospf-1] quit
```

After the previous configuration, execute the **display ip routing-table** command on each router. You can see that the routers have learned the routes to one another, including the routes to the Loopback interfaces. Take Router A as an example:

```
[RouterA] display ip routing-table
```

```
Destinations : 6          Routes : 6

Destination/Mask    Proto  Pre  Cost           NextHop         Interfac
1.1.1.9/32          Direct  0    0             127.0.0.1       InLoop0
2.2.2.9/32          OSPF   10   1             10.1.1.2        Eth1/1
10.1.1.0/24         Direct  0    0             10.1.1.1        Eth1/1
10.1.1.1/32         Direct  0    0             127.0.0.1       InLoop0
127.0.0.0/8         Direct  0    0             127.0.0.1       InLoop0
127.0.0.1/32        Direct  0    0             127.0.0.1       InLoop0
```

3. Configure BGP on Router B and Router C to make sure the ASs can communicate with each other:

Configure Router B.

```
[RouterB] bgp 100
[RouterB-bgp] peer 20.1.1.2 as-number 200
[RouterB-bgp] ipv4-family unicast
[RouterB-bgp-ipv4] peer 20.1.1.2 enable
[RouterB-bgp-ipv4] import-route ospf
```



```
[RouterB-bgp-ipv4] import-route direct
[RouterB-bgp-ipv4] quit
[RouterB-bgp] quit
```

Configure Router C.

```
[RouterC] bgp 200
[RouterC-bgp] peer 20.1.1.1 as-number 100
[RouterC-bgp] ipv4-family unicast
[RouterC-bgp-ipv4] peer 20.1.1.1 enable
[RouterC-bgp-ipv4] import-route ospf
[RouterC-bgp-ipv4] import-route direct
[RouterC-bgp-ipv4] quit
[RouterC-bgp] quit
```

After the previous configuration, execute the **display ip routing-table** command on each router. You can see that the routers have learned the AS-external routes. Take Router A as an example:

```
[RouterA] display ip routing-table
```

```
Destinations : 10          Routes : 10
```

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
1.1.1.9/32	Direct	0	0	127.0.0.1	InLoop0
2.2.2.9/32	OSPF	10	1	10.1.1.2	Eth1/1
3.3.3.9/32	O_ASE	150	1	10.1.1.2	Eth1/1
4.4.4.9/32	O_ASE	150	1	10.1.1.2	Eth1/1
10.1.1.0/24	Direct	0	0	10.1.1.1	Eth1/1
10.1.1.1/32	Direct	0	0	127.0.0.1	InLoop0
20.1.1.0/24	O_ASE	150	1	10.1.1.2	Eth1/1
30.1.1.0/24	O_ASE	150	1	10.1.1.2	Eth1/1
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0

4. Configure an LSR ID, and enable MPLS, MPLS TE, and RSVP-TE:

Configure Router A.

```
[RouterA] mpls lsr-id 1.1.1.9
[RouterA] mpls te
[RouterA-te] quit
[RouterA] rsvp
[RouterA-rsvp] quit
[RouterA] interface ethernet 1/1
[RouterA-Ethernet1/1] mpls enable
[RouterA-Ethernet1/1] mpls te enable
[RouterA-Ethernet1/1] rsvp enable
[RouterA-Ethernet1/1] quit
```

Configure Router B.

```
[RouterB] mpls lsr-id 2.2.2.9
[RouterB] mpls te
[RouterB-te] quit
[RouterB] rsvp
[RouterB-rsvp] quit
```

```
[RouterB] interface ethernet 1/1
[RouterB-Ethernet1/1] mpls enable
[RouterB-Ethernet1/1] mpls te enable
[RouterB-Ethernet1/1] rsvp enable
[RouterB-Ethernet1/1] quit
[RouterB] interface pos 5/0
[RouterB-POS5/0] mpls enable
[RouterB-POS5/0] mpls te enable
[RouterB-POS5/0] rsvp enable
[RouterB-POS5/0] quit
```

Configure Router C.

```
[RouterC] mpls lsr-id 3.3.3.9
[RouterC] mpls te
[RouterC-te] quit
[RouterC] rsvp
[RouterC-rsvp] quit
[RouterC] interface ethernet 1/1
[RouterC-Ethernet1/1] mpls enable
[RouterC-Ethernet1/1] mpls te enable
[RouterC-Ethernet1/1] rsvp enable
[RouterC-Ethernet1/1] quit
[RouterC] interface pos 5/0
[RouterC-POS5/0] mpls enable
[RouterC-POS5/0] mpls te enable
[RouterC-POS5/0] rsvp enable
[RouterC-POS5/0] quit
```

Configure Router D.

```
[RouterD] mpls lsr-id 4.4.4.9
[RouterD] mpls te
[RouterD-te] quit
[RouterD] rsvp
[RouterD-rsvp] quit
[RouterD] interface ethernet 1/1
[RouterD-Ethernet1/1] mpls enable
[RouterD-Ethernet1/1] mpls te enable
[RouterD-Ethernet1/1] rsvp enable
[RouterD-Ethernet1/1] quit
```

5. Configure a loose explicit route on Router A:

```
[RouterA] explicit-path atod
[RouterA-explicit-path-atod] nexthop 10.1.1.2 include loose
[RouterA-explicit-path-atod] nexthop 20.1.1.2 include loose
[RouterA-explicit-path-atod] nexthop 30.1.1.2 include loose
[RouterA-explicit-path-atod] quit
```

6. Configure MPLS TE attributes of links:

Configure the maximum link bandwidth and maximum reservable bandwidth on Router A.

```
[RouterA] interface ethernet 1/1
[RouterA-Ethernet1/1] mpls te max-link-bandwidth 10000
```

```
[RouterA-Ethernet1/1] mpls te max-reservable-bandwidth 5000
[RouterA-Ethernet1/1] quit
```

Configure the maximum link bandwidth and maximum reservable bandwidth on Router B.

```
[RouterB] interface ethernet 1/1
[RouterB-Ethernet1/1] mpls te max-link-bandwidth 10000
[RouterB-Ethernet1/1] mpls te max-reservable-bandwidth 5000
[RouterB-Ethernet1/1] quit
[RouterB] interface pos 5/0
[RouterB-POS5/0] mpls te max-link-bandwidth 10000
[RouterB-POS5/0] mpls te max-reservable-bandwidth 5000
[RouterB-POS5/0] quit
```

Configure the maximum link bandwidth and maximum reservable bandwidth on Router C.

```
[RouterC] interface ethernet 1/1
[RouterC-Ethernet1/1] mpls te max-link-bandwidth 10000
[RouterC-Ethernet1/1] mpls te max-reservable-bandwidth 5000
[RouterC-Ethernet1/1] quit
[RouterC] interface pos 5/0
[RouterC-POS5/0] mpls te max-link-bandwidth 10000
[RouterC-POS5/0] mpls te max-reservable-bandwidth 5000
[RouterC-POS5/0] quit
```

Configure the maximum link bandwidth and maximum reservable bandwidth on Router D.

```
[RouterD] interface ethernet 1/1
[RouterD-Ethernet1/1] mpls te max-link-bandwidth 10000
[RouterD-Ethernet1/1] mpls te max-reservable-bandwidth 5000
[RouterD-Ethernet1/1] quit
```

7. Configure an MPLS TE tunnel:

On Router A, configure MPLS TE tunnel interface Tunnel 1, specify the tunnel destination address as the LSR ID of Router D, configure MPLS TE to use RSVP-TE to establish the tunnel, assign 2000 kbps bandwidth to the tunnel, and specify the explicit path **atod** for the tunnel.

```
[RouterA] interface tunnel 1 mode mpls-te
[RouterA-Tunnell] ip address 7.1.1.1 255.255.255.0
[RouterA-Tunnell] destination 4.4.4.9
[RouterA-Tunnell] mpls te signaling rsvp-te
[RouterA-Tunnell] mpls te bandwidth 2000
[RouterA-Tunnell] mpls te path preference 5 explicit-path atod
[RouterA-Tunnell] quit
```

8. Configure a static route on Router A to direct the traffic destined for subnet 30.1.1.0/24 to MPLS TE tunnel 1:

```
[RouterA] ip route-static 30.1.1.2 24 tunnel 1 preference 1
```

Verifying the configuration

Execute the **display interface tunnel** command on Router A. The output shows that the tunnel interface is up.

```
[RouterA] display interface tunnel 1
Tunnell current state: UP
Line protocol current state: UP
Description: Tunnell Interface
The Maximum Transmit Unit is 64000
```

```

Internet Address is 7.1.1.1/24 Primary
Tunnel source unknown, destination 4.4.4.9
Tunnel bandwidth 64 (kbps)
Tunnel TTL 255
Tunnel protocol/transport CR_LSP
Last clearing of counters: Never
  Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
  Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
  0 packets input, 0 bytes, 0 drops
  3077 packets output, 197028 bytes, 0 drops

```

Execute the **display mpls te tunnel-interface** command on Router A to display detailed information about the MPLS TE tunnel.

```

[RouterA] display mpls te tunnel-interface
Tunnel Name           : Tunnel 1
Tunnel State          : Up (Main CRLSP up, Shared-resource CRLSP down)
Tunnel Attributes     :
  LSP ID               : 23549           Tunnel ID              : 1
  Admin State          : Normal
  Ingress LSR ID       : 1.1.1.9         Egress LSR ID          : 4.4.4.9
  Signaling             : RSVP-TE        Static CRLSP Name      : -
  Resv Style           : SE
  Tunnel mode          : -
  Reverse-LSP name     : -
  Reverse-LSP LSR ID  : -               Reverse-LSP Tunnel ID : -
  Class Type           : CT0             Tunnel Bandwidth       : 2000 kbps
  Reserved Bandwidth  : 2000 kbps
  Setup Priority       : 7               Holding Priority        : 7
  Affinity Attr/Mask  : 0/0
  Explicit Path       : atod
  Backup Explicit Path : -
  Metric Type         : IGP
  Record Route        : Disabled        Record Label           : Disabled
  FRR Flag            : Disabled        Backup Bandwidth Flag  : Disabled
  Backup Bandwidth Type : -             Backup Bandwidth       : -
  Route Pinning       : Disabled
  Retry Limit         : 10              Retry Interval         : 2 sec
  Reoptimization      : Disabled        Reoptimization Freq   : -
  Backup Type         : None            Backup LSP ID          : -
  Auto Bandwidth      : Disabled        Auto Bandwidth Freq   : -
  Min Bandwidth       : -               Max Bandwidth          : -
  Collected Bandwidth : -

```

Execute the **display ip routing-table** command on Router A. You can see a static route entry with interface Tunnel1 as the egress interface.

```

[RouterA] display ip routing-table

Destinations : 14          Routes : 14

Destination/Mask    Proto  Pre  Cost           NextHop           Interface

```

1.1.1.9/32	Direct	0	0	127.0.0.1	InLoop0
2.2.2.9/32	OSPF	10	1	10.1.1.2	Eth1/1
3.3.3.9/32	O_ASE	150	1	10.1.1.2	Eth1/1
4.4.4.9/32	O_ASE	150	1	10.1.1.2	Eth1/1
7.1.1.0/24	Direct	0	0	7.1.1.1	Tun1
7.1.1.1/32	Direct	0	0	127.0.0.1	InLoop0
10.1.1.0/24	Direct	0	0	10.1.1.1	Eth1/1
10.1.1.1/32	Direct	0	0	127.0.0.1	InLoop0
20.1.1.0/24	O_ASE	150	1	10.1.1.2	Eth1/1
30.1.1.0/24	Static	1	0	7.1.1.1	Tun1
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0

FRR configuration example

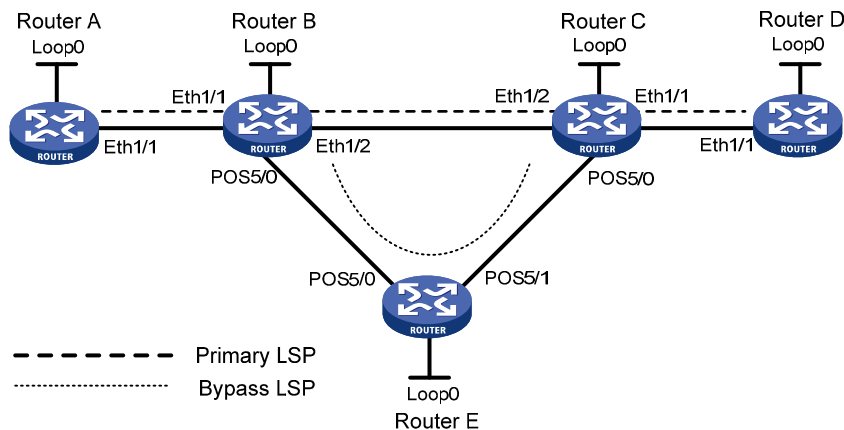
Network requirements

On the primary CRLSP Router A—Router B—Router C—Router D, use FRR to protect the link Router B—Router C.

Use RSVP-TE to establish the primary CRLSP and bypass CRLSP of the MPLS TE tunnel based on the constraints of the explicit paths. The bypass CRLSP uses path Router B—Router E—Router C. Router B is the PLR and Router C is the MP.

Configure BFD for RSVP-TE between Router B and Router C. When the link between Router B and Router C fails, BFD can detect the failure quickly and notify RSVP-TE of the failure, so RSVP-TE can switch traffic to the bypass CRLSP.

Figure 26 Network diagram



Device	Interface	IP address	Device	Interface	IP address
Router A	Loop0	1.1.1.1/32	Router E	Loop0	5.5.5.5/32
	Eth1/1	2.1.1.1/24		POS5/0	3.2.1.2/24
Router B	Loop0	2.2.2.2/32		POS5/1	3.3.1.1/24
	Eth1/1	2.1.1.2/24	Router C	Loop0	3.3.3.3/32
	Eth1/2	3.1.1.1/24		Eth1/1	4.1.1.1/24
	POS5/0	3.2.1.1/24		Eth1/2	3.1.1.2/24
Router D	Loop0	4.4.4.4/32		POS5/0	3.3.1.2/24

Device	Interface	IP address	Device	Interface	IP address
	Eth1/1	4.1.1.2/24			

Configuration procedure

1. Configure IP addresses and masks for interfaces. (Details not shown.)
2. Configure IS-IS to advertise interface addresses, including the Loopback interface address. (Details not shown.)
3. Configure an LSR ID, and enable MPLS, MPLS TE, RSVP-TE, and CSPF on each router, and enable BFD for RSVP-TE on Router B and Router C:

Configure Router A.

```
<RouterA> system-view
[RouterA] mpls lsr-id 1.1.1.1
[RouterA] mpls te
[RouterA-te] quit
[RouterA] rsvp
[RouterA-rsvp] quit
[RouterA] interface ethernet 1/1
[RouterA-Ethernet1/1] mpls enable
[RouterA-Ethernet1/1] mpls te enable
[RouterA-Ethernet1/1] rsvp enable
[RouterA-Ethernet1/1] quit
```

Configure Router B.

```
<RouterB> system-view
[RouterB] mpls lsr-id 2.2.2.2
[RouterB] mpls te
[RouterB-te] quit
[RouterB] rsvp
[RouterB-rsvp] quit
[RouterB] interface ethernet 1/1
[RouterB-Ethernet1/1] mpls enable
[RouterB-Ethernet1/1] mpls te enable
[RouterB-Ethernet1/1] rsvp enable
[RouterB-Ethernet1/1] quit
[RouterB] interface ethernet 1/2
[RouterB-Ethernet1/2] mpls enable
[RouterB-Ethernet1/2] mpls te enable
[RouterB-Ethernet1/2] rsvp enable
[RouterB-Ethernet1/2] rsvp bfd enable
[RouterB-Ethernet1/2] quit
[RouterB] interface pos 5/0
[RouterB-POS5/0] mpls enable
[RouterB-POS5/0] mpls te enable
[RouterB-POS5/0] rsvp enable
[RouterB-POS5/0] quit
```

Perform the same configurations on Router C as on Router B. Perform the same configurations on Router D and Router E as on Router A. (Details not shown.)

4. Configure an MPLS TE tunnel on Router A, the ingress node of the primary CRLSP:

Configure an explicit path for the primary CRLSP.

```
[RouterA] explicit-path pri-path
[RouterA-explicit-path-pri-path] nexthop 2.1.1.2
[RouterA-explicit-path-pri-path] nexthop 3.1.1.2
[RouterA-explicit-path-pri-path] nexthop 4.1.1.2
[RouterA-explicit-path-pri-path] nexthop 4.4.4.4
[RouterA-explicit-path-pri-path] quit
```

Configure the MPLS TE tunnel for the primary CRLSP: Create MPLS TE tunnel interface Tunnel 4, and specify the tunnel destination address as the LSR ID of Router D, the tunnel signaling protocol as RSVP-TE, and the explicit path as **pri-path**.

```
[RouterA] interface tunnel 4 mode mpls-te
[RouterA-Tunnel4] ip address 10.1.1.1 255.255.255.0
[RouterA-Tunnel4] destination 4.4.4.4
[RouterA-Tunnel4] mpls te signaling rsvp-te
[RouterA-Tunnel4] mpls te path preference 1 explicit-path pri-path
```

Enable FRR for the MPLS TE tunnel.

```
[RouterA-Tunnel4] mpls te fast-reroute
[RouterA-Tunnel4] quit
```

After the previous configuration, execute the **display interface tunnel** command on Router A. The output shows that the tunnel interface Tunnel 4 is up.

```
[RouterA] display interface tunnel
Tunnel4 current state: UP
Line protocol current state: UP
Description: Tunnel3 Interface
The Maximum Transmit Unit is 64000
Internet Address is 9.1.1.1/24 Primary
Tunnel source unknown, destination 3.3.3.9
Tunnel bandwidth 64 (kbps)
Tunnel TTL 255
Tunnel protocol/transport CR_LSP
Last clearing of counters: Never
  Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
  Last 300 seconds output rate: 1911 bytes/sec, 15288 bits/sec, 0 packets/sec
  0 packets input, 0 bytes, 0 drops
  1526 packets output, 22356852 bytes, 0 drops
```

Execute the **display mpls te tunnel-interface** command on Router A to display detailed information about the MPLS TE tunnel.

```
[RouterA] display mpls te tunnel-interface
Tunnel Name          : Tunnel 4
Tunnel State         : Up (Main CRLSP up, Shared-resource CRLSP down)
Tunnel Attributes    :
  LSP ID              : 48960                Tunnel ID           : 4
  Admin State         : Normal
  Ingress LSR ID      : 1.1.1.1              Egress LSR ID       : 3.3.3.3
  Signaling            : RSVP-TE              Static CRLSP Name   : -
```

```

Resv Style          : SE
Tunnel mode        : -
Reverse-LSP name   : -
Reverse-LSP LSR ID : -           Reverse-LSP Tunnel ID: -
Class Type         : CT0           Tunnel Bandwidth    : 0 kbps
Reserved Bandwidth : 0 kbps
Setup Priority     : 7             Holding Priority    : 7
Affinity Attr/Mask : 0/0
Explicit Path      : pri-path
Backup Explicit Path : -
Metric Type       : IGP
Record Route      : Disabled       Record Label        : Disabled
FRR Flag          : Enabled        Backup Bandwidth Flag: Disabled
Backup Bandwidth Type: -           Backup Bandwidth    : -
Route Pinning     : Disabled
Retry Limit       : 10             Retry Interval      : 2 sec
Reoptimization    : Disabled       Reoptimization Freq : -
Backup Type       : None           Backup LSP ID       : -
Auto Bandwidth    : Disabled       Auto Bandwidth Freq : -
Min Bandwidth     : -              Max Bandwidth       : -
Collected Bandwidth : -

```

5. Configure a bypass tunnel on Router B, the PLR:

Configure an explicit path for the bypass tunnel.

```

[RouterB] explicit-path by-path
[RouterB-explicit-path-by-path] nexthop 3.2.1.2
[RouterB-explicit-path-by-path] nexthop 3.3.1.2
[RouterB-explicit-path-by-path] nexthop 3.3.3.3
[RouterB-explicit-path-by-path] quit

```

Configure the bypass tunnel: Create MPLS TE tunnel interface Tunnel 5, and specify the tunnel destination address as LSR ID of Router C, the tunnel signaling protocol as RSVP-TE, and the explicit path to be used as **by-path**.

```

[RouterB] interface tunnel 5 mode mpls-te
[RouterB-Tunnel5] ip address 11.1.1.1 255.255.255.0
[RouterB-Tunnel5] destination 3.3.3.3
[RouterB-Tunnel5] mpls te signaling rsvp-te
[RouterB-Tunnel5] mpls te path preference 1 explicit-path by-path

```

Configure the bandwidth that the bypass tunnel can protect.

```

[RouterB-Tunnel5] mpls te backup bandwidth 10000
[RouterB-Tunnel5] quit

```

Bind the bypass tunnel to the protected interface.

```

[RouterB] interface ethernet 1/2
[RouterB-Ethernet1/2] rsvp fast-reroute bypass-tunnel tunnel 5
[RouterB-Ethernet1/2] quit

```

Execute the **display interface tunnel** command on Router B. You can see that the tunnel interface Tunnel 5 is up.

6. Configure a static route on Router A to direct the traffic destined for subnet 4.1.1.0/24 to MPLS TE tunnel 4.


```
[RouterA] ip route-static 4.1.1.2 24 tunnel 4 preference 1
```

Verifying the configuration

Execute the **display mpls lsp** command on each router. You can see the LSP entries. Router B and Router C each have two CRLSPs. The bypass CRLSP backs up the primary CRLSP.

```
[RouterA] display mpls lsp
```

FEC	Proto	In/Out Label	Interface/Out NHLFE
1.1.1.1/4/48960	RSVP	-/1245	Eth1/1
2.1.1.2	Local	-/-	Eth1/1

```
[RouterB] display mpls lsp
```

FEC	Proto	In/Out Label	Interface/Out NHLFE
1.1.1.1/4/48960	RSVP	1245/3	ETH1/2
Backup		1245/3	Tun5
2.2.2.2/5/31857	RSVP	-/3	ETH1/2
3.2.1.2	Local	-/-	POS5/0
3.1.1.2	Local	-/-	ETH1/2

Shut down the protected interface Ethernet 1/2 on the PLR (Router B).

```
[RouterB] interface ethernet 1/2
```

```
[RouterB-Ethernet1/2] shutdown
```

```
[RouterB-Ethernet1/2] quit
```

Execute the **display interface tunnel 4** command on Router A to display information about the primary CRLSP. You can see that the tunnel interface is still up.

Execute the **display mpls te tunnel-interface** command on Router A to display the detailed information of the tunnel interface.

```
[RouterA] display mpls te tunnel-interface
```

```
Tunnel Name          : Tunnel 4
Tunnel State         : Up (Main CRLSP up, Shared-resource CRLSP being set up)
Tunnel Attributes    :
  LSP ID              : 18753                Tunnel ID           : 4
  Admin State         : Normal
  Ingress LSR ID     : 1.1.1.1              Egress LSR ID      : 3.3.3.3
  Signaling           : RSVP-TE             Static CRLSP Name   : -
  Resv Style          : SE
  Tunnel mode         : -
  Reverse-LSP name    : -
  Reverse-LSP LSR ID : -                  Reverse-LSP Tunnel ID: -
  Class Type          : CT0                 Tunnel Bandwidth    : 0 kbps
  Reserved Bandwidth : 0 kbps
  Setup Priority       : 7                  Holding Priority     : 7
  Affinity Attr/Mask  : 0/0
  Explicit Path       : pri-path
  Backup Explicit Path : -
  Metric Type         : IGP
  Record Route        : Disabled            Record Label        : Disabled
  FRR Flag            : Enabled             Backup Bandwidth Flag: Disabled
  Backup Bandwidth Type: -                  Backup Bandwidth    : -
  Route Pinning       : Disabled
  Retry Limit         : 10                  Retry Interval       : 2 sec
```

```

Reoptimization      : Disabled      Reoptimization Freq : -
Backup Type        : None           Backup LSP ID       : -
Auto Bandwidth     : Disabled       Auto Bandwidth Freq : -
Min Bandwidth      : -              Max Bandwidth       : -
Collected Bandwidth : -

```

NOTE:

If you execute the **display mpls te tunnel-interface** command immediately after an FRR, you can see two CRLSPs in up state. This is because FRR uses the **make-before-break** mechanism to set up a new LSP, and the old LSP is deleted after the new one has been established for a while.

Execute the **display mpls lsp** command on Router B. The output shows that the bypass tunnel is in use.

```

[RouterB] display mpls lsp
FEC                Proto   In/Out Label   Interface/Out NHLFE
1.1.1.1/4/18753    RSVP   1122/3    Tun5
2.2.2.2/5/40312    RSVP   -/1150    GE0/1/4
3.2.1.2            Local   -/-       GE0/1/4

```

On the PLR, configure the interval for selecting an optimal bypass tunnel as 5 seconds.

```

[RouterB] rsvp
[RouterB-rsvp] fast-reroute timer 5
[RouterB-rsvp] quit

```

On the PLR, bring up the protected interface Ethernet 1/2.

```

[RouterB] interface ethernet 1/2
[RouterB-Ethernet1/2] undo shutdown
[RouterB-Ethernet1/2] quit

```

On Router A, execute the **display interface tunnel 4** command to display information about the primary CRLSP. You can see that the tunnel interface is in up state.

Wait for about 5 seconds, execute the **display mpls lsp verbose** command on Router B. You can see that Tunnel5 is bound to interface Ethernet 1/2 but not in use.

Execute the **display ip routing-table** command on Router A. You can see a static route entry with interface Tunnel4 as the egress interface.

Configuring a static CRLSP

Overview

A static Constraint-based Routed Label Switched Path (CRLSP) is established by manually specifying the incoming label, outgoing label, and required bandwidth on each node (ingress, transit, or egress node) of the forwarding path. If the device does not have enough bandwidth resources required by a CRLSP, the CRLSP cannot be established.

Static CRLSPs consume fewer resources, but they cannot automatically adapt to network topology changes. Therefore, static CRLSPs are suitable for small and stable networks with simple topologies.

Follow these guidelines to establish a static CRLSP:

- On the ingress node, specify the outgoing label for the CRLSP, the next hop or the outgoing interface to the next hop, and the required bandwidth, create an MPLS TE tunnel interface, and reference the static CRLSP for the tunnel interface. The tunnel interface adds the outgoing label of the static CRLSP to each packet, and forwards the packet to the next hop or out of the outgoing interface.
- A transit node swaps the label carried in a received packet with a specific label, and forwards the packet to the next hop or out of the outgoing interface. You must specify on each transit node the incoming label, the outgoing label, the next hop or the outgoing interface, and the required bandwidth.
- If it is not configured with the penultimate hop popping function, an egress node pops the incoming label of a packet, and performs label forwarding according to the inner label or IP forwarding. You are only required to specify the incoming label on the egress node.
- The outgoing label specified on an LSR must be the same as the incoming label specified on the directly connected downstream LSR.

Configuration procedure

Static CRLSPs are special static LSPs. They use the same label space as static LSPs. On a device, a static CRLSP and a static LSP cannot use the same incoming label.

A static CRLSP can be used to forward MPLS TE traffic only after you create an MPLS TE tunnel interface on the ingress node and reference the static CRLSP for the tunnel interface. For more information about MPLS TE, see "Configuring MPLS TE."

Before you configure a static CRLSP, complete the following tasks:

- Identify the ingress node, transit nodes, and egress node of the CRLSP.
- Enable MPLS on all interfaces that participate in MPLS forwarding. For more information, see "Configuring basic MPLS."
- Enable MPLS TE for each node and interface that the CRLSP traverses. For more information, see "Configuring MPLS TE."

To configure a static CRLSP:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a static CRLSP.	<ul style="list-style-type: none"> Configure the ingress node: static-cr-lsp ingress <i>lsp-name</i> { next-hop <i>next-hop-addr</i> outgoing-interface <i>interface-type interface-number</i> } out-label <i>out-label-value</i> [bandwidth [<i>ct0</i> <i>ct1</i> <i>ct2</i> <i>ct3</i>] <i>bandwidth-value</i>] Configure a transit node: static-cr-lsp transit <i>lsp-name</i> in-label <i>in-label-value</i> { next-hop <i>next-hop-addr</i> outgoing-interface <i>interface-type interface-number</i> } out-label <i>out-label-value</i> [bandwidth [<i>ct0</i> <i>ct1</i> <i>ct2</i> <i>ct3</i>] <i>bandwidth-value</i>] Configure the egress node: static-cr-lsp egress <i>lsp-name</i> in-label <i>in-label-value</i> 	<p>Use one command according to the position of a device on the network.</p> <p>By default, no static CRLSP exists.</p> <p>Do not configure the next hop address as a local public IP address when configuring the static CRLSP on the ingress node or a transit node.</p> <p>You do not need to execute the static-cr-lsp egress command on the egress node if the outgoing label configured on the penultimate hop of the static CRLSP is 0 or 3.</p>

Displaying static CRLSPs

Execute **display** commands in any view.

Task	Command
Display static CRLSP information.	display mpls static-cr-lsp [<i>lsp-name</i> <i>lsp-name</i>] [verbose]

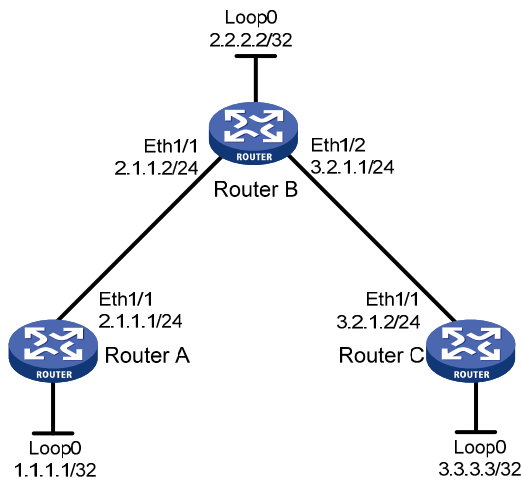
Static CRLSP configuration example

Network requirements

Router A, Router B, and Router C run IS-IS.

Establish an MPLS TE tunnel over a static CRLSP from Router A to Router C.

Figure 27 Network diagram



Configuration procedure

1. Configure IP addresses and masks for interfaces. (Details not shown.)
2. Configure IS-IS to advertise interface addresses, including the Loopback interface address:

Configure Router A.

```
<RouterA> system-view
[RouterA] isis 1
[RouterA-isis-1] network-entity 00.0005.0000.0000.0001.00
[RouterA-isis-1] quit
[RouterA] interface ethernet 1/1
[RouterA-Ethernet1/1] isis enable 1
[RouterA-Ethernet1/1] quit
[RouterA] interface loopback 0
[RouterA-LoopBack0] isis enable 1
[RouterA-LoopBack0] quit
```

Configure Router B.

```
<RouterB> system-view
[RouterB] isis 1
[RouterB-isis-1] network-entity 00.0005.0000.0000.0002.00
[RouterB-isis-1] quit
[RouterB] interface ethernet 1/1
[RouterB-Ethernet1/1] isis enable 1
[RouterB-Ethernet1/1] quit
[RouterB] interface ethernet 1/2
[RouterB-Ethernet1/2] isis enable 1
[RouterB-Ethernet1/2] quit
[RouterB] interface loopback 0
[RouterB-LoopBack0] isis enable 1
[RouterB-LoopBack0] quit
```

Configure Router C.

```
<RouterC> system-view
[RouterC] isis 1
```

```

[RouterC-isis-1] network-entity 00.0005.0000.0000.0003.00
[RouterC-isis-1] quit
[RouterC] interface ethernet 1/1
[RouterC-Ethernet1/1] isis enable 1
[RouterC-Ethernet1/1] quit
[RouterC] interface loopback 0
[RouterC-LoopBack0] isis enable 1
[RouterC-LoopBack0] quit

```

After the previous configuration, execute the **display ip routing-table** command on each router. The output shows that the routers have learned the routes to one another, including the routes to the Loopback interfaces.

3. Configure an LSR ID, and enable MPLS and MPLS TE:

Configure Router A.

```

[RouterA] mpls lsr-id 1.1.1.1
[RouterA] mpls te
[RouterA-te] quit
[RouterA] interface ethernet 1/1
[RouterA-Ethernet1/1] mpls enable
[RouterA-Ethernet1/1] mpls te enable
[RouterA-Ethernet1/1] quit

```

Configure Router B.

```

[RouterB] mpls lsr-id 2.2.2.2
[RouterB] mpls te
[RouterB-te] quit
[RouterB] interface ethernet 1/1
[RouterB-Ethernet1/1] mpls enable
[RouterB-Ethernet1/1] mpls te enable
[RouterB-Ethernet1/1] quit
[RouterB] interface ethernet 1/2
[RouterB-Ethernet1/2] mpls enable
[RouterB-Ethernet1/2] mpls te enable
[RouterB-Ethernet1/2] quit

```

Configure Router C.

```

[RouterC] mpls lsr-id 3.3.3.3
[RouterC] mpls te
[RouterC-mpls] quit
[RouterC] interface ethernet 1/1
[RouterC-Ethernet1/1] mpls enable
[RouterC-Ethernet1/1] mpls te enable
[RouterC-Ethernet1/1] quit

```

4. Configure an MPLS TE tunnel:

On Router A, configure the MPLS TE tunnel interface Tunnel0, specify the tunnel destination address as the LSR ID of Router C, and configure MPLS TE to use a static CRLSP to establish the tunnel.

```

[RouterA] interface tunnel 0 mode mpls-te
[RouterA-Tunnel0] ip address 6.1.1.1 255.255.255.0
[RouterA-Tunnel0] destination 3.3.3.3

```

```
[RouterA-Tunnel0] mpls te signaling static
[RouterA-Tunnel0] quit
```

5. Create a static CRLSP:

Configure Router A as the ingress node of the static CRLSP, and specify the next hop address as 2.1.1.2 and outgoing label as 20.

```
[RouterA] static-cr-lsp ingress static-cr-lsp-1 nexthop 2.1.1.2 out-label 20
```

On Router A, configure tunnel 0 to reference the static CRLSP **static-cr-lsp-1**.

```
[RouterA] interface Tunnel0
[RouterA-Tunnel0] mpls te static-cr-lsp static-cr-lsp-1
[RouterA-Tunnel0] quit
```

Configure Router B as the transit node of the static CRLSP, and specify the incoming label as 20, the next hop address as 3.2.1.2, and outgoing label as 30.

```
[RouterB] static-cr-lsp transit static-cr-lsp-1 in-label 20 nexthop 3.2.1.2 out-label 30
```

Configure Router C as the egress node of the static CRLSP, and specify the incoming label as 30.

```
[RouterC] static-cr-lsp egress static-cr-lsp-1 in-label 30
```

6. Configure a static route on Router A to direct traffic destined for subnet 3.2.1.0/24 to MPLS TE tunnel 0:

```
[RouterA] ip route-static 3.2.1.2 24 tunnel 0 preference 1
```

Verifying the configuration

Execute the **display interface tunnel** command on Router A. The output shows that the tunnel interface is up.

```
[RouterA] display interface tunnel
Tunnel0
Current state: UP
Line protocol state: UP
Description: Tunnel0 Interface
Maximum Transmit Unit: 64000
Internet Address is 6.1.1.1/24 Primary
Tunnel source unknown, destination 3.3.3.3
Tunnel bandwidth 64 (kbps)
Tunnel TTL 255
Tunnel protocol/transport CR_LSP
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 0 packets, 0 bytes, 0 drops
Output: 0 packets, 0 bytes, 0 drops
```

Execute the **display mpls te tunnel-interface** command on Router A to display detailed information about the MPLS TE tunnel.

```
[RouterA] display mpls te tunnel-interface
Tunnel Name           : Tunnel 0
Tunnel State          : Up (Main CRLSP up)
Tunnel Attributes     :
  LSP ID              : 1           Tunnel ID           : 0
  Admin State         : Normal
```

```

Ingress LSR ID      : 1.1.1.1          Egress LSR ID      : 3.3.3.3
Signaling           : Static           Static CRLSP Name   : static-cr-lsp-1
Resv Style          : -
Tunnel mode         : -
Reverse-LSP name    : -
Reverse-LSP LSR ID  : -                Reverse-LSP Tunnel ID: -
Class Type          : -                Tunnel Bandwidth    : -
Reserved Bandwidth  : -
Setup Priority       : 0                Holding Priority     : 0
Affinity Attr/Mask  : -/-
Explicit Path       : -
Backup Explicit Path : -
Metric Type         : TE
Record Route        : -                Record Label        : -
FRR Flag            : -                Backup Bandwidth Flag: -
Backup Bandwidth Type: -                Backup Bandwidth    : -
Route Pinning       : -
Retry Limit         : 10                Retry Interval      : 2 sec
Reoptimization      : -                Reoptimization Freq : -
Backup Type         : -                Backup LSP ID       : -
Auto Bandwidth      : -                Auto Bandwidth Freq : -
Min Bandwidth       : -                Max Bandwidth       : -
Collected Bandwidth : -

```

Execute the **display mpls lsp** command or the **display mpls static-cr-lsp** command on each router to display the static CRLSP information.

```
[RouterA] display mpls lsp
```

```

FEC                Proto    In/Out Label    Interface/Out NHLFE
1.1.1.1/0/1        StaticCR -/20         Eth1/1
2.1.1.2            Local    -/-            Eth1/1

```

```
[RouterB] display mpls lsp
```

```

FEC                Proto    In/Out Label    Interface/Out NHLFE
-                  StaticCR 20/30         Eth1/2
3.2.1.2            Local    -/-            Eth1/2

```

```
[RouterC] display mpls lsp
```

```

FEC                Proto    In/Out Label    Interface/Out NHLFE
-                  StaticCR 30/-          -

```

```
[RouterA] display mpls static-cr-lsp
```

```

Name              LSR Type  In/Out Label    Out Interface    State
static-cr-lsp-1  Ingress  Null/20         Eth1/1           Up

```

```
[RouterB] display mpls static-cr-lsp
```

```

Name              LSR Type  In/Out Label    Out Interface    State
static-cr-lsp-1  Transit  20/30          Eth1/2           Up

```

```
[RouterC] display mpls static-cr-lsp
```

```

Name              LSR Type  In/Out Label    Out Interface    State
static-cr-lsp-1  Egress   30/Null        -                Up

```

Execute the **display ip routing-table** command on Router A. The output shows a static route entry with interface Tunnel0 as the egress interface.

Configuring RSVP

Overview

The Resource Reservation Protocol (RSVP) is a signaling protocol that reserves resources on a network. Extended RSVP supports MPLS label distribution and allows resource reservation information to be transmitted with label bindings. This extended RSVP is called "RSVP-TE." RSVP-TE is a label distribution protocol for MPLS TE. It distributes MPLS labels and reserve resources on the nodes of a specific path to establish a CRLSP.

RSVP messages

RSVP uses the following types of messages:

- **Path messages**—Sent by the sender downstream along the data transmission path to save path state information on each node along the path.
- **Resv messages**—Sent by the receiver upstream towards the sender to request resource reservation and to create and maintain reservation state on each node along the reverse of the data transmission path.
- **PathTear messages**—Sent downstream by the sender or a transit node to remove the path state and related reservation state on each node along the path.
- **ResvTear messages**—Sent upstream by the receiver or a transit node to remove the reservation state on each node along the path.
- **PathErr messages**—Sent upstream by the receiver or a transit node to report Path message processing errors to the sender. They do not affect the state of the nodes along the path.
- **ResvErr messages**—Sent downstream by the sender or a transit node to notify the downstream nodes that an error has occurred during Resv message processing or that a reservation error has occurred because of preemption.
- **ResvConf messages**—Sent to the receiver to confirm Resv messages.
- **Hello messages**—Sent between any two directly connected RSVP neighbors to set up and maintain the neighbor relationship. Hello messages are sent only when the RSVP hello extension has been enabled.

RSVP-TE extends RSVP by adding new objects to Path and Resv messages. In addition to label bindings, these objects also carry routing constraints to support CRLSP and FRR.

New objects added to the Path message include:

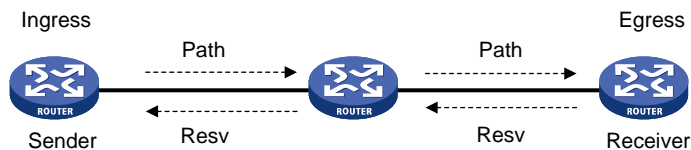
- **LABEL_REQUEST**—Requests the downstream node to allocate a label.
- **EXPLICIT_ROUTE**—Carries the path information calculated by the ingress node, making sure the CRLSP is set up along that path.
- **RECORD_ROUTE**—Records the path that the CRLSP actually traverses and the label allocated by each node on the path.
- **SESSION_ATTRIBUTE**—Carries the MPLS TE tunnel attributes, such as the setup priority, holding priority, and affinity.

New objects added to the Resv message include:

- **LABEL**—Advertises the label allocated by the downstream node to the upstream node.
- **RECORD_ROUTE**—Records the path that the CRLSP actually traverses and the label allocated by each node on the path.

CRLSP setup procedure

Figure 28 Setting up a CRLSP



As shown in Figure 28, a CRLSP is set up using the following steps:

1. The ingress LSR generates a Path message that carries LABEL_REQUEST, and then forwards the message along the path calculated by CSPF hop-by-hop towards the egress LSR.
2. After receiving the Path message, the egress LSR generates a Resv message carrying the reservation information and the LABEL object, and forwards the Resv message to the ingress LSR along the reverse direction of the path that the Path message traveled. The Resv message advertises labels, reserves resources, and creates a reserve state on each LSR it passes, so QoS can be guaranteed for services transmitted on the CRLSP.
3. When the ingress LSR receives the Resv message, the CRLSP is established.

RSVP refresh mechanism

Refresh messages

RSVP maintains resource reservation states on a node by periodically sending messages.

The resource reservation states include path states and reservation states. A path state is saved in a path state block (PSB), and a reservation state is saved in a reservation state block (RSB). A PSB is created by a Path message and saves the LABEL_REQUEST object. A RSB is created by a Resv message and saves the LABEL object.

The path states and reservation states are refreshed periodically by Path and Resv messages. A state is removed if no refresh messages for the state are received in a certain interval, and the CRLSP established based on this state is also removed.

The Path and Resv messages for refreshing the resource reservation states are collectively referred to as refresh messages. Refresh messages can also be used to recover from lost RSVP messages.

When multiple RSVP sessions exist on a network, the periodically sent refresh messages can cause network degradation. In this case, the refreshing interval of Path and Resv messages should not be too short. However, delay sensitive applications want to recover from lost RSVP messages through the refresh messages as soon as possible. In this case, the refreshing interval should not be too long. You can use the summary refresh (Srefresh) and the reliable RSVP message delivery functions to find the appropriate balance.

Srefresh

Srefresh is implemented by adding a Message_ID object to a Path or Resv message to uniquely identify the message. To refresh Path and Resv states, RSVP does not need to send standard Path and Resv messages. Instead, it sends an Srefresh message carrying a set of Message_ID objects that identify the Path and Resv states to be refreshed. The Srefresh function reduces the number of refresh messages on the network and speeds up refresh message processing.

Reliable RSVP message delivery

An RSVP sender cannot know or retransmit lost RSVP messages. The reliable RSVP message delivery mechanism is designed to ensure reliable transmission.

This mechanism requires the peer device to acknowledge each RSVP message received from the local device. If no acknowledgement is received, the local device retransmits the message.

To implement reliable RSVP message delivery, a node sends an RSVP message that includes a Message_ID object in which the ACK_Desired flag is set. The receiver needs to confirm the delivery by sending back a message that includes the Message_ID_ACK object. If the sender does not receive a Message_ID_ACK within the retransmission interval (Rf), it retransmits the message when Rf expires and sets the next transmission interval to $(1 + \text{delta}) \times Rf$. The sender repeats this process until it receives the Message_ID_ACK before the retransmission time expires or it has transmitted the message three times.

RSVP authentication

RSVP authentication ensures integrity of RSVP messages, and prevents false resource reservation requests from occupying network resources.

With RSVP authentication, the sender uses the MD5 algorithm and the authentication key to calculate a message digest for an RSVP message, and inserts the message digest to the RSVP message. When the receiver receives the message, it performs the same calculation and compares the result with the message digest. If they match, the receiver accepts the message. Otherwise, it drops the message.

By carrying a sequence number in a message, RSVP authentication can also prevent packet replay attacks. The device records the sequence number of a received RSVP message, and determines whether the subsequent messages are valid according to the recorded sequence number. If the sequence number of a subsequent message is within the valid range, the device accepts the message. Otherwise, it drops the message.

RSVP GR

RSVP GR preserves the soft state and label forwarding information when the signaling protocol or control plane fails, so that LSRs can still forward packets according to forwarding entries.

RSVP GR defines two roles:

- **GR restarter**—Router that gracefully restarts due to a manually configured command or a fault. It must be GR-capable.
- **GR helper**—Neighbor of the GR restarter. A GR helper maintains the neighbor relationship with the GR restarter and helps the GR restarter restore its LFIB information. A GR helper must be GR-capable.

The router can act only as a GR helper.

The RSVP GR function depends on the extended hello capability of RSVP. A GR-capable device advertises its GR capability and relevant time parameters to its neighbors in RSVP hello packets. If a

device and all its neighbor have the RSVP GR capability and have exchanged GR parameters, each of them can function as the GR helper of another device.

A GR helper considers that a GR restarter is rebooting when it receives no hello packets from the restarter in a specific period of time. When a GR restarter is rebooting, the GR helpers retain soft state information about the GR restarter and continue sending hello packets periodically to the GR restarter until the restart timer expires.

If a GR helper receives a hello message from the GR restarter before the restart timer expires, the recovery timer is started and signaling packet exchange is triggered to restore the original soft state. Otherwise, all RSVP soft state information and forwarding entries relevant to the neighbor are removed. When the recovery timer expires, soft state information and forwarding entries that are not restored are removed.

Protocols and standards

- RFC 2205, *Resource ReSerVation Protocol*
- RFC 3209, *RSVP-TE: Extensions to RSVP for LSP Tunnels*
- RFC 2961, *RSVP Refresh Overhead Reduction Extensions*

RSVP configuration task list

To configure RSVP, perform the following tasks:

Tasks at a glance

(Required.) [Enabling RSVP](#)

(Optional.) Perform the following tasks on each node of an MPLS TE tunnel according to your network requirements:

- [Configuring RSVP refresh](#)
 - [Configuring RSVP Srefresh and reliable RSVP message delivery](#)
 - [Configuring RSVP hello extension](#)
 - [Configuring RSVP authentication](#)
 - [Configuring RSVP GR](#)
 - [Enabling BFD for RSVP](#)
-

Enabling RSVP

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable global RSVP and enter RSVP view.	rsvp	By default, global RSVP is disabled.
3. Return to system view.	quit	N/A
4. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
5. Enable RSVP for the interface.	rsvp enable	By default, RSVP is disabled on an interface.

Configuring RSVP refresh

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter RSVP view.	rsvp	N/A
3. Configure the refresh interval for Path and Resv messages.	refresh interval <i>interval</i>	By default, the refresh interval is 30 seconds for both path and Resv messages.
4. Configure the PSB and RSB timeout multiplier.	keep-multiplier <i>number</i>	By default, the PSB and RSB timeout multiplier is 3.

Configuring RSVP Srefresh and reliable RSVP message delivery

After Srefresh is enabled, RSVP maintains the path and reservation states by sending Srefresh messages rather than standard refresh messages.

To configure Srefresh and reliable RSVP message delivery:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Enable Srefresh and reliable RSVP message delivery.	rsvp reduction srefresh [reliability]	By default, Srefresh and reliable RSVP message delivery are disabled.
4. Configure the retransmission increment value for reliable RSVP message delivery.	rsvp reduction retransmit increment <i>increment-value</i>	By default, the RSVP message retransmission increment is 1. This command takes effect after reliable RSVP message delivery is enabled by using the rsvp reduction srefresh reliability command.
5. Configure the retransmission interval for reliable RSVP message delivery.	rsvp reduction retransmit interval <i>retrans-timer-value</i>	By default, the RSVP message retransmission interval is 500 milliseconds. This command takes effect after reliable RSVP message delivery is enabled by using the rsvp reduction srefresh reliability command.

Configuring RSVP hello extension

When RSVP hello extension is enabled on an interface, the device receives and sends hello messages through the interface to detect the neighbor's status.

If the device receives a hello request from the neighbor, the device replies with a hello ACK message. If the device receives no hello request from the neighbor within the interval specified by the **hello interval** command, the device sends hello requests to the neighbor.

When the number of consecutive lost hellos or erroneous hellos from the neighbor reaches the maximum (specified by the **hello lost** command), the device determines the neighbor is in fault. If GR is configured, the device serves as a GR helper to help the neighbor to restart. If FRR is configured, the device performs an FRR. For more information about FRR, see "Configuring MPLS TE."

To configure RSVP hello extension:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter RSVP view.	rsvp	N/A
3. Configure the maximum number of consecutive lost or erroneous hellos.	hello lost times	By default, the maximum number is 4.
4. Configure the interval for sending hello requests.	hello interval interval	By default, hello requests are sent every 5 seconds.
5. Return to system view.	quit	N/A
6. Enter interface view.	interface interface-type interface-number	N/A
7. Enable RSVP hello extension.	resvp hello enable	By default, RSVP hello extension is disabled.

Configuring RSVP authentication

RSVP adopts hop-by-hop authentication to prevent fake resource reservation requests from occupying network resources. The interfaces at the two ends of a link must use the same authentication key.

RSVP authentication can be configured in the following views:

- **RSVP view**—Configuration in this view applies to all RSVP messages.
- **RSVP neighbor view**—Configuration in this view applies only to RSVP messages received from and sent to the specified neighbor.
- **Interface view**—Configuration in this view applies only to RSVP messages received and sent by the current interface.

Configurations in RSVP neighbor view, interface view, and RSVP view are in descending order of priority.

To configure RSVP authentication in RSVP neighbor view:

Step	Command	Remarks
1. Enter system view.	system-view	N/A

Step	Command	Remarks
2. Enter RSVP view.	rsvp	N/A
3. Create an RSVP authentication neighbor and enter RSVP neighbor view.	peer ip-address	By default, the device does not have any RSVP authentication neighbors.
4. Enable RSVP authentication for the RSVP neighbor and specify the authentication key.	authentication key { cipher plain } auth-key	By default, RSVP authentication is disabled.
5. Enable challenge-response handshake for the RSVP neighbor.	authentication challenge	By default, the challenge-response handshake function is disabled.
6. Configure the idle timeout for the RSVP security associations with the RSVP neighbor.	authentication lifetime life-time	By default, the idle timeout is 1800 seconds (30 minutes).
7. Specify the maximum number of out-of-sequence authenticated RSVP messages that can be received from the RSVP neighbor.	authentication window-size number	By default, only one authenticated RSVP message can be received out of sequence.

To configure RSVP authentication in interface view:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface interface-type interface-number	N/A
3. Enable RSVP authentication on the interface and configure the authentication key.	rsvp authentication key { cipher plain } auth-key	By default, RSVP authentication is disabled. Do not enable both RSVP authentication and FRR on the same interface.
4. Enable challenge-response handshake on the interface.	rsvp authentication challenge	By default, the challenge-response handshake function is disabled.
5. Configure the idle timeout for RSVP security associations on the interface.	rsvp authentication lifetime life-time	By default, the idle timeout is 1800 seconds (30 minutes).
6. Specify the maximum number of out-of-sequence authenticated RSVP messages that can be received on the interface.	rsvp authentication window-size number	By default, only one authenticated RSVP message can be received out of sequence.

To configure RSVP authentication in RSVP view:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter RSVP view.	rsvp	N/A

Step	Command	Remarks
3. Enable RSVP authentication globally and configure the authentication key.	authentication key { cipher plain } <i>auth-key</i>	By default, RSVP authentication is disabled.
4. Enable challenge-response handshake globally.	authentication challenge	By default, the challenge-response handshake function is disabled.
5. Configure the global idle timeout for RSVP security associations.	authentication lifetime <i>life-time</i>	By default, the idle timeout is 1800 seconds (30 minutes).
6. Specify the global RSVP authentication window size—the maximum number of authenticated RSVP messages that can be received out of sequence.	authentication window-size <i>number</i>	By default, only one authenticated RSVP message can be received out of sequence.

Configuring RSVP GR

RSVP GR depends on the RSVP hello extension function. When configuring RSVP GR, you must enable RSVP hello extension.

Perform this task on GR-capable devices.

To configure RSVP GR:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter RSVP view.	rsvp	N/A
3. Enable GR for RSVP.	graceful-restart enable	By default, RSVP GR is disabled.
4. Return to system view.	quit	N/A
5. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
6. Enable RSVP hello extension.	rsvp hello enable	By default, RSVP hello extension is disabled.

Enabling BFD for RSVP

If a link fails, MPLS TE tunnels over the link fail to forward packets. MPLS TE cannot quickly detect a link failure. To address this issue, you can enable BFD for RSVP so MPLS TE can quickly switch data from the primary path to the backup path upon a link failure.

To enable BFD for RSVP:

Step	Command	Remarks
1. Enter system view.	system-view	N/A

Step	Command	Remarks
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	You must enable RSVP on the interface.
3. Enable BFD for the RSVP neighbor on the interface.	rsvp bfd enable	By default, RSVP BFD is disabled.

Displaying and maintaining RSVP

Execute **display** commands in any view and **reset** commands in user view.

Task	Command
Display RSVP information.	display rsvp [interface [<i>interface-type</i> <i>interface-number</i>]]
Display information about the security associations established with RSVP neighbors.	display rsvp authentication [from <i>ip-address</i>] [to <i>ip-address</i>] [verbose]
Display information about CRLSPs established through RSVP.	display rsvp lsp [destination <i>ip-address</i>] [source <i>ip-address</i>] [tunnel-id <i>tunnel-id</i>] [lsp-id <i>lsp-id</i>] [verbose]
Display information about RSVP neighbors.	display rsvp peer [interface <i>interface-type</i> <i>interface-number</i>] [ip <i>ip-address</i>] [verbose]
Display information about RSVP resource reservation requests sent to upstream devices.	display rsvp request [destination <i>ip-address</i>] [source <i>ip-address</i>] [tunnel-id <i>tunnel-id</i>] [prev-hop <i>ip-address</i>] [verbose]
Display information about RSVP resource reservation states.	display rsvp reservation [destination <i>ip-address</i>] [source <i>ip-address</i>] [tunnel-id <i>tunnel-id</i>] [nexthop <i>ip-address</i>] [verbose]
Display information about RSVP path states.	display rsvp sender [destination <i>ip-address</i>] [source <i>ip-address</i>] [tunnel-id <i>tunnel-id</i>] [lsp-id <i>lsp-id</i>] [verbose]
Display RSVP statistics.	display rsvp statistics [interface [<i>interface-type</i> <i>interface-number</i>]]
Clear RSVP security associations.	reset rsvp authentication [from <i>ip-address</i> to <i>ip-address</i>]
Clear RSVP statistics.	reset rsvp statistics [interface [<i>interface-type</i> <i>interface-number</i>]]

Configuring tunnel policies

Overview

Tunnel policies enable a PE to forward traffic for each MPLS VPN over a preferred tunnel or over multiple tunnels when the PE has multiple tunnels to the peer PE. The tunnels supported by MPLS VPN include MPLS LSPs, MPLS TE tunnels, and GRE tunnels.

For more information about MPLS TE, see "Configuring MPLS TE." For more information about GRE, see *Layer 3—IP Services Configuration Guide*. For more information about MPLS VPNs, see "Configuring MPLS L3VPN."

Configuring a tunnel policy

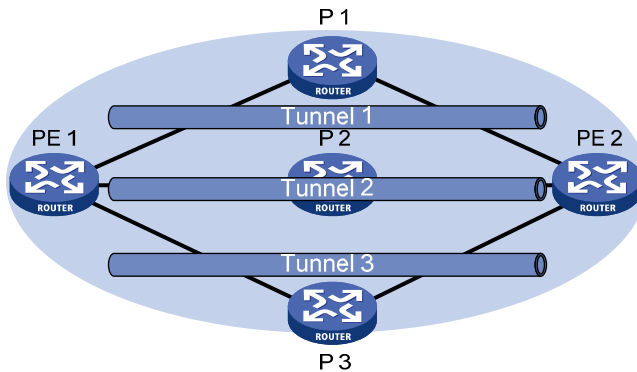
Configuration guidelines

- To select a preferred tunnel, create a tunnel policy and specify the preferred tunnel with the **preferred-path** command. The destination address of the preferred tunnel identifies a peer PE so the PE will forward traffic destined for that peer PE over the preferred tunnel. If you specify multiple preferred tunnels that have the same destination address in a tunnel policy, only the first configured tunnel takes effect. If the first tunnel is not available, the second tunnel is used, and so forth. No load balancing will be performed on these tunnels. This method explicitly specifies an MPLS TE or GRE tunnel for an MPLS VPN, facilitating traffic planning. HP recommends that you use this method.
- To select multiple tunnels for load sharing, create a tunnel policy and specify the tunnel selection order and the number of tunnels by using the **select-seq load-balance-number** command. A tunnel type closer to the **select-seq** keyword has a higher priority. For example, the **select-seq lsp gre load-balance-number 3** command gives LSP higher priority over GRE. If no LSP is available or the number of LSPs is less than 3, VPN uses GRE tunnels. The tunnels selected by this method are not fixed, complicating traffic planning. HP recommends not using this method.

If you configure both methods for a tunnel policy, the tunnel policy selects tunnels in the following steps:

1. If the destination address of a preferred tunnel identifies a peer PE, the tunnel policy uses the preferred tunnel to forward traffic destined for the peer PE without using any other tunnels.
2. If not, the tunnel policy selects tunnels as configured by the **select-seq load-balance-number** command.

Figure 29 MPLS VPN tunnel selection diagram



As shown in [Figure 29](#), PE 1 and PE 2 have multiple tunnels in between and they are connected to multiple MPLS VPNs. You can control the paths for VPN traffic by using one of the following methods:

- Configure multiple tunnel policies, and specify a preferred tunnel for each policy by using the **preferred-path** command. Apply these policies to different MPLS VPNs to forward the traffic of each VPN over a specific tunnel.
- Configure one tunnel policy, and use the **select-seq load-balance-number** command to specify the tunnel selection order and the number of tunnels for load balancing. Apply the tunnel policy to MPLS VPNs to forward the traffic of every VPN over multiple tunnels.

The second method distributes traffic of a single VPN to multiple tunnels. The transmission delays on different tunnels can greatly vary. Therefore, the destination device or the upper layer application might take a great time to sequence the packets. HP recommends not using the second method.

Configuration procedure

To configure a tunnel policy:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a tunnel policy, and enter tunnel policy view.	tunnel-policy <i>tunnel-policy-name</i>	By default, no tunnel policy is configured.
3. Configure tunnel selection methods.	<ul style="list-style-type: none"> • (Method 1) Specify a preferred tunnel: preferred-path tunnel <i>number</i> • (Method 2) Configure the tunnel selection order and the number of tunnels for load balancing: select-seq { cr-lsp gre lsp } * load-balance-number <i>number</i> 	<p>Configure one or both methods.</p> <p>By default, no preferred tunnel is specified.</p> <p>By default, only one tunnel is selected in LSP—GRE—CR-LSP order.</p>

NOTE:

For a VPN to exclusively use a tunnel, you can specify the tunnel as the preferred tunnel in a tunnel policy, and apply the policy only to that VPN.

Displaying tunnel information

Execute the **display** command in any view.

Task	Command
Display tunnel information.	display mpls tunnel { all statistics [vpn-instance <i>vpn-instance-name</i>] destination { <i>tunnel-ipv4-dest</i> <i>tunnel-ipv6-dest</i> }

Preferred tunnel configuration example

Network requirements

PE 1 has multiple tunnels to reach PE 2: one MPLS TE tunnel on the interface Tunnel1, one GRE tunnel on the interface Tunnel2, and one LDP LSP tunnel.

Two MPLS VPN instances, **vpna** and **vpnb**, exist on PE 1. Configure PE 1 to use the MPLS TE tunnel to forward traffic for both VPNs.

Configuration procedure

1. Create a tunnel policy named **preferredte1**, and configure tunnel 1 as the preferred tunnel:

```
<PE1> system-view
[PE1] tunnel-policy preferredte1
[PE1-tunnel-policy-preferredte1] preferred-path tunnel 1
[PE1-tunnel-policy-preferredte1] quit
```

2. Configure MPLS VPN instances and apply the tunnel policy to the VPN instances:

Create MPLS VPN instance **vpna**, and apply tunnel policy **preferredte1** to it.

```
[PE1] ip vpn-instance vpna
[PE1-vpn-instance-vpna] route-distinguisher 100:1
[PE1-vpn-instance-vpna] vpn-target 100:1
[PE1-vpn-instance-vpna] tnl-policy preferredte1
[PE1-vpn-instance-vpna] quit
```

Create MPLS VPN instance **vpnb**, and apply tunnel policy **preferredte1** to it.

```
[PE1] ip vpn-instance vpb
[PE1-vpn-instance-vpb] route-distinguisher 100:2
[PE1-vpn-instance-vpb] vpn-target 100:2
[PE1-vpn-instance-vpb] tnl-policy preferredte1
```

Exclusive tunnel configuration example

Network requirements

PE 1 has multiple tunnels to reach PE 2: one MPLS TE tunnel on the interface Tunnel1, one GRE tunnel on the interface Tunnel2, and one LDP LSP tunnel.

Two MPLS VPNs, **vpna** and **vpnb**, exist on PE 1. The VPN **vpna** exclusively uses the MPLS TE tunnel, and the VPN **vpnb** exclusively uses the GRE tunnel.

Configuration procedure

1. Configure tunnel policies on PE 1:

Create tunnel policy **preferredte1**, and configure tunnel 1 as the preferred tunnel.

```
<PE1> system-view
[PE1] tunnel-policy preferredte1
[PE1-tunnel-policy-preferredte1] preferred-path tunnel 1
[PE1-tunnel-policy-preferredte1] quit
```

Create tunnel policy **preferredgre2**, and configure tunnel 2 as the preferred tunnel.

```
[PE1] tunnel-policy preferredgre2
[PE1-tunnel-policy-preferredgre2] preferred-path tunnel 2
[PE1-tunnel-policy-preferredgre2] quit
```

2. Configure MPLS VPN instances and apply tunnel policies to the VPN instances:

Create MPLS VPN instance **vpna**, and apply tunnel policy **preferredte1** to it.

```
[PE1] ip vpn-instance vpna
[PE1-vpn-instance-vpna] route-distinguisher 100:1
[PE1-vpn-instance-vpna] vpn-target 100:1
[PE1-vpn-instance-vpna] tnl-policy preferredte1
[PE1-vpn-instance-vpna] quit
```

Create MPLS VPN instance **vpnb**, and apply tunnel policy **preferredgre2** to it.

```
[PE1] ip vpn-instance vpb
[PE1-vpn-instance-vpb] route-distinguisher 100:2
[PE1-vpn-instance-vpb] vpn-target 100:2
[PE1-vpn-instance-vpb] tnl-policy preferredgre2
```

Tunnel selection order configuration example

Network requirements

PE 1 has multiple tunnels to reach PE 2: one MPLS TE tunnel on the interface Tunnel1, one GRE tunnel on the interface Tunnel2, and one LDP LSP tunnel.

Only one MPLS VPN, **vpna**, exists on PE 1. Select only one tunnel in LDP LSP-MPLS TE-GRE order for this VPN.

Configuration procedure

1. Create tunnel policy **seq-lsp-te-gre**, specify the tunnel selection order and set the number of tunnels for load balancing to 1—no load balancing.

```
<PE1> system-view
[PE1] tunnel-policy seq-lsp-te-gre
[PE1-tunnel-policy-seq-lsp-te-gre] select-seq lsp cr-lsp gre load-balance-number 1
[PE1-tunnel-policy-seq-lsp-te-gre] quit
```

2. Create MPLS VPN instance **vpna**, and apply tunnel policy **seq-lsp-te-gre** to it.

```
[PE1] ip vpn-instance vpna
[PE1-vpn-instance-vpna] route-distinguisher 100:1
[PE1-vpn-instance-vpna] vpn-target 100:1
[PE1-vpn-instance-vpna] tnl-policy seq-lsp-te-gre
```

Preferred tunnel and tunnel selection order configuration example

Network requirements

PE 1 has multiple tunnels to reach PE 2: two MPLS TE tunnels on the interface Tunnel1 and Tunnel3, one GRE tunnel on the interface Tunnel2, and one LDP LSP tunnel.

PE 1 has multiple MPLS VPN instances: vpna, vpnb, vpnc, vpnd, vpne, vpnf, and vpng. [Table 2](#) shows the tunnel policy that PE 1 uses for each VPN instance.

Table 2 Tunnel policies used for VPN instances

VPN instance	Tunnel policy
vpna, vpnb	Use MPLS TE tunnel Tunnel1 as the preferred tunnel.
vpnc, vpnd	Use MPLS TE tunnel Tunnel3 as the preferred tunnel.
vpne, vpnf	Use GRE tunnel Tunnel2 as the preferred tunnel.
vpng	Uses one tunnel selected in LDP LSP-GRE-MPLS TE order.

Configuration procedure

1. Configure tunnel policies on PE 1:

Create tunnel policy **preferredte1**, and configure tunnel 1 as the preferred tunnel.

```
<PE1> system-view
[PE1] tunnel-policy preferredte1
[PE1-tunnel-policy-preferredte1] preferred-path tunnel 1
[PE1-tunnel-policy-preferredte1] quit
```

Create tunnel policy **preferredte3**, and configure tunnel 3 as the preferred tunnel.

```
[PE1] tunnel-policy preferredte3
[PE1-tunnel-policy-preferredte3] preferred-path tunnel 3
[PE1-tunnel-policy-preferredte3] quit
```

Create tunnel policy **preferredgre2**, and configure tunnel 2 as the preferred tunnel.

```
[PE1] tunnel-policy preferredgre2
[PE1-tunnel-policy-preferredgre2] preferred-path tunnel 2
[PE1-tunnel-policy-preferredgre2] quit
```

Create tunnel policy **select-lsp**, and configure the policy to select only one tunnel in LDP LSP-GRE-MPLS TE order.

```
[PE1] tunnel-policy select-lsp
[PE1-tunnel-policy-select-lsp] select-seq lsp gre cr-lsp
[PE1-tunnel-policy-select-lsp] quit
```

2. Configure MPLS VPN instances and apply tunnel policies to the VPN instances:

Create MPLS VPN instances **vpna** and **vpnb**, and apply tunnel policy **preferredte1** to them.

```
[PE1] ip vpn-instance vpna
[PE1-vpn-instance-vpna] route-distinguisher 100:1
[PE1-vpn-instance-vpna] vpn-target 100:1
[PE1-vpn-instance-vpna] tnl-policy preferredte1
[PE1-vpn-instance-vpna] quit
```

```

[PE1] ip vpn-instance vpb
[PE1-vpn-instance-vpb] route-distinguisher 100:2
[PE1-vpn-instance-vpb] vpn-target 100:2
[PE1-vpn-instance-vpb] tnl-policy preferredte1
[PE1-vpn-instance-vpb] quit
# Create MPLS VPN instances vpnc and vpnd, and apply tunnel policy preferredte3 to them.
[PE1] ip vpn-instance vpnc
[PE1-vpn-instance-vpnc] route-distinguisher 100:3
[PE1-vpn-instance-vpnc] vpn-target 100:3
[PE1-vpn-instance-vpnc] tnl-policy preferredte3
[PE1-vpn-instance-vpnc] quit
[PE1] ip vpn-instance vpnd
[PE1-vpn-instance-vpnd] route-distinguisher 100:4
[PE1-vpn-instance-vpnd] vpn-target 100:4
[PE1-vpn-instance-vpnd] tnl-policy preferredte3
[PE1-vpn-instance-vpnd] quit
# Create MPLS VPN instances vpne and vpnf, and apply tunnel policy preferredgre2 to them.
[PE1] ip vpn-instance vpne
[PE1-vpn-instance-vpne] route-distinguisher 100:5
[PE1-vpn-instance-vpne] vpn-target 100:5
[PE1-vpn-instance-vpne] tnl-policy preferredgre2
[PE1-vpn-instance-vpne] quit
[PE1] ip vpn-instance vpnf
[PE1-vpn-instance-vpnf] route-distinguisher 100:6
[PE1-vpn-instance-vpnf] vpn-target 100:6
[PE1-vpn-instance-vpnf] tnl-policy preferredgre2
[PE1-vpn-instance-vpnf] quit
# Create MPLS VPN instance vpng and apply tunnel policy select-lsp to it.
[PE1] ip vpn-instance vpng
[PE1-vpn-instance-vpng] route-distinguisher 100:7
[PE1-vpn-instance-vpng] vpn-target 100:7
[PE1-vpn-instance-vpng] tnl-policy select-lsp

```

Configuring MPLS L3VPN

This chapter describes MPLS L3VPN configuration.

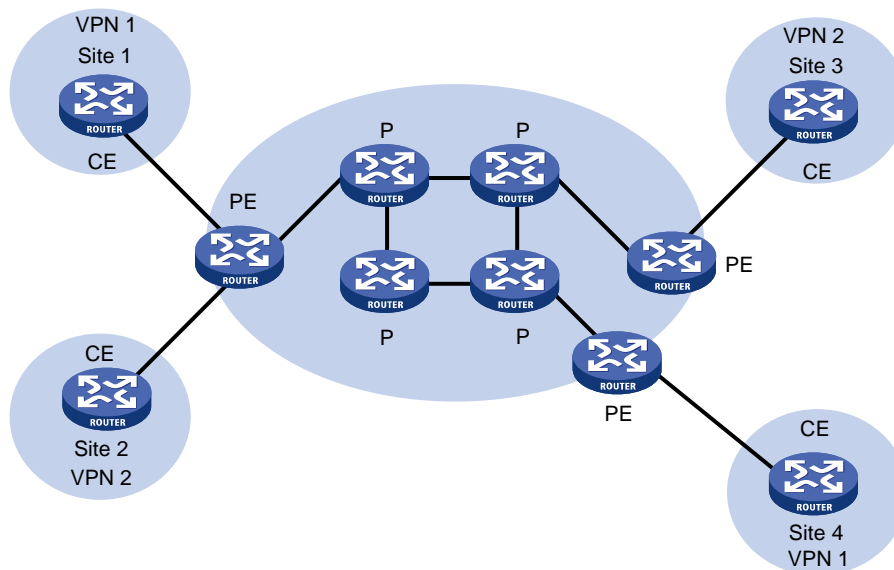
Overview

MPLS L3VPN is a L3VPN technology used to interconnect geographically dispersed VPN sites. MPLS L3VPN uses BGP to advertise VPN routes and uses MPLS to forward VPN packets over a service provider backbone.

MPLS L3VPN provides flexible networking modes, excellent scalability, and convenient support for MPLS QoS and MPLS TE.

Basic MPLS L3VPN architecture

Figure 30 Basic MPLS L3VPN architecture



A basic MPLS L3VPN architecture has the following types of devices:

- **Customer edge device**—A CE device resides on a customer network and has one or more interfaces directly connected to a service provider network. It does not support VPN or MPLS.
- **Provider edge device**—A PE device resides at the edge of a service provider network and connects to one or more CEs. All MPLS VPN services are processed on PEs.
- **Provider device**—A P device is a core device on a service provider network. It is not directly connected to any CE. A P device has only basic MPLS forwarding capability and does not handle VPN routing information.

MPLS L3VPN concepts

Site

A site has the following features:

- A site is a group of IP systems with IP connectivity that does not rely on any service provider network.
- The classification of a site depends on the topology relationship of the devices, rather than the geographical positions, though the devices at a site are, in most cases, adjacent to each other geographically.
- The devices at a site can belong to multiple VPNs, which means that a site can belong to multiple VPNs.
- A site is connected to a provider network through one or more CEs. A site can contain multiple CEs, but a CE can belong to only one site.

Sites connected to the same provider network can be classified into different sets by policies. Only the sites in the same set can access each other through the provider network. Such a set is called a VPN.

VPN instance

VPN instances, also called virtual routing and forwarding (VRF) instances, implement route isolation, data independence, and data security for VPNs.

A VPN instance has the following components:

- A separate Label Forwarding Information Base (LFIB).
- An IP routing table.
- Interfaces bound to the VPN instance.
- VPN instance administration information, including route distinguishers (RDs), route targets (RTs), and route filtering policies.

To associate a site with a VPN instance, bind the VPN instance to the PE's interface connected to the site. A site can be associated with only one VPN instance, and different sites can associate with the same VPN instance. A VPN instance contains the VPN membership and routing rules of associated sites.

Address space overlapping

Each VPN independently manages its address space.

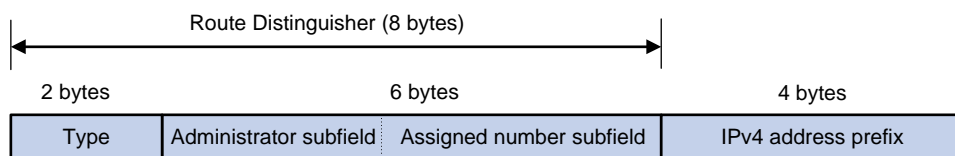
The address spaces of VPNs might overlap. For example, if both VPN 1 and VPN 2 use the addresses on subnet 10.110.10.0/24, address space overlapping occurs.

VPN-IPv4 address

BGP cannot process overlapping VPN address spaces. For example, if both VPN 1 and VPN 2 use the subnet 10.110.10.0/24 and each advertise a route destined for the subnet, BGP selects only one of them, resulting in the loss of the other route.

Multiprotocol BGP (MP-BGP) can solve this problem by advertising VPN-IPv4 prefixes.

Figure 31 VPN-IPv4 address structure



As shown in [Figure 31](#), a VPN-IPv4 address consists of 12 bytes. The first eight bytes represent the RD, followed by a four-byte IPv4 prefix. The RD and the IPv4 prefix form a unique VPN-IPv4 prefix.

An RD can be in one of the following formats:

- When the Type field is 0, the Administrator subfield occupies two bytes, the Assigned number subfield occupies four bytes, and the RD format is *16-bit AS number:32-bit user-defined number*. For example, 100:1.
- When the Type field is 1, the Administrator subfield occupies four bytes, the Assigned number subfield occupies two bytes, and the RD format is *32-bit IPv4 address:16-bit user-defined number*. For example, 172.1.1.1:1.
- When the Type field is 2, the Administrator subfield occupies four bytes, the Assigned number subfield occupies two bytes, and the RD format is *32-bit AS number:16-bit user-defined number*, where the minimum value of the AS number is 65536. For example, 65536:1.

To guarantee global uniqueness for an RD, do not set the Administrator subfield to any private AS number or private IP address.

Route target attribute

MPLS L3VPN uses route target community attributes to control the advertisement of VPN routing information. A VPN instance on a PE supports the following types of route target attributes:

- **Export target attribute**—A PE sets the export target attribute for VPN-IPv4 routes learned from directly connected sites before advertising them to other PEs.
- **Import target attribute**—A PE checks the export target attribute of VPN-IPv4 routes received from other PEs. If the export target attribute matches the import target attribute of a VPN instance, the PE adds the routes to the routing table of the VPN instance.

Route target attributes define which sites can receive VPN-IPv4 routes, and from which sites a PE can receive routes.

Like RDs, route target attributes can be one of the following formats:

- *16-bit AS number:32-bit user-defined number*. For example, 100:1.
- *32-bit IPv4 address:16-bit user-defined number*. For example, 172.1.1.1:1.
- *32-bit AS number:16-bit user-defined number*, where the minimum value of the AS number is 65536. For example, 65536:1.

MP-BGP

MP-BGP supports multiple address families, including IPv4 multicast, IPv6 unicast, IPv6 multicast, and VPN-IPv4 address families.

In MPLS L3VPN, MP-BGP advertises VPN-IPv4 routes for VPN sites between PEs.

MPLS L3VPN route advertisement

In a basic MPLS L3VPN, CEs and PEs are responsible for advertising VPN routing information. P routers maintain only the routes within the backbone. A PE maintains only routing information for directly connected VPNs, rather than for all VPNs.

VPN routing information is advertised from the local CE to the remote CE using the following process:

1. From the local CE to the ingress PE:

The CE advertises standard IPv4 routing information to the ingress PE over a static route, RIP route, OSPF route, IS-IS route, EBGp route, or IBGP route.

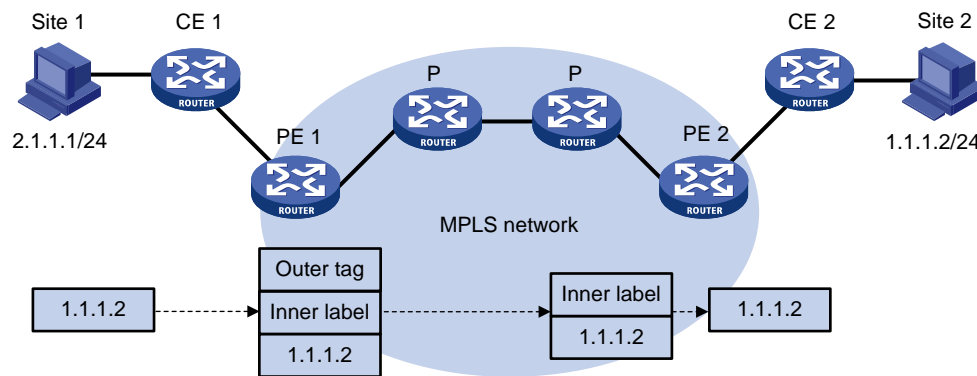
2. From the ingress PE to the egress PE:
The ingress PE adds RD and route target attributes to these standard IPv4 routes to create VPN-IPv4 routes, saves them to the routing table of the VPN instance created for the CE, and advertises the VPN-IPv4 routes to the egress PE through MP-BGP.
3. From the egress PE to the remote CE:
After receiving the VPN-IPv4 routes, the egress PE compares their export target attribute with the local import target attribute, and, if they match, adds the routes to the routing table of the VPN instance. Then the egress PE restores the VPN-IPv4 routes to the original IPv4 routes and advertises those routes to the connected CE over a static route, RIP route, OSPF route, IS-IS route, EBGP route, or IBGP route.

MPLS L3VPN packet forwarding

In a basic MPLS L3VPN (within a single AS), a PE adds the following information into VPN packets:

- **Outer tag**—Identifies the public tunnel from the local PE to the remote PE. The public tunnel can be an LSP, an MPLS TE tunnel, or a GRE tunnel. Based on the outer tag, a VPN packet can be forwarded along the public tunnel to the remote PE. For a GRE public tunnel, the outer tag is the GRE encapsulation. For an LSP or MPLS TE tunnel, the outer tag is an MPLS label.
- **Inner label**—Identifies the remote VPN site. The remote PE uses the inner label to forward packets to the target VPN site. MP-BGP advertises inner labels for VPN routes among PEs.

Figure 32 VPN packet forwarding



As shown in Figure 32, a VPN packet is forwarded from Site 1 to Site 2 using the following process:

1. Site 1 sends an IP packet with the destination address 1.1.1.2. CE 1 transmits the packet to PE 1.
2. PE 1 finds the matching VPN route based on the inbound interface and destination address of the packet, labels the packet with both the inner label and the outer tag, and forwards the packet to the public tunnel.
3. P devices forward the packet to PE 2 by the outer tag. If the outer tag is an MPLS label, the label is removed from the packet at the penultimate hop. If the outer tag is GRE encapsulation, PE 2 removes the GRE encapsulation.
4. PE 2 finds the matching VPN route according to the inner label and destination address of the packet, and then forwards the packet out of the interface to CE 2.
5. CE 2 transmits the packet to the destination through IP forwarding.

When two sites of a VPN are connected to the same PE, the PE directly forwards packets between the two sites through the VPN routing table without adding any tag or label.

For more information about GRE, see *Layer-3 IP Services Configuration Guide*.

MPLS L3VPN networking schemes

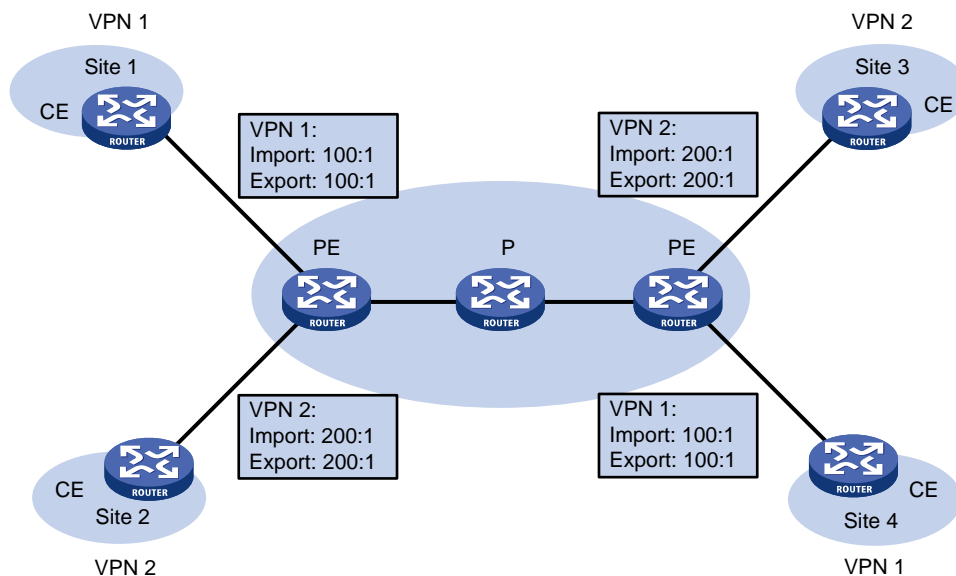
In MPLS L3VPNs, route target attributes are used to control the advertisement and reception of VPN routes between sites. They work independently and can be configured with multiple values to support flexible VPN access control and implement multiple types of VPN networking schemes.

Basic VPN networking scheme

In the simplest case, all users in a VPN form a closed user group. They can forward traffic to each other but cannot communicate with any user outside the VPN.

For the basic VPN networking scheme, you must assign a route target to each VPN for identifying the export target attribute and import target attribute of the VPN. Moreover, this route target cannot be used by any other VPNs.

Figure 33 Network diagram for basic VPN networking scheme



In [Figure 33](#), the route target for VPN 1 is 100:1, while that for VPN 2 is 200:1. The two VPN 1 sites can communicate with each other, and the two VPN 2 sites can communicate with each other. However, the VPN 1 sites cannot communicate with the VPN 2 sites.

Hub and spoke networking scheme

The hub and spoke networking scheme is suitable for a VPN where all users must communicate with each other through an access control device.

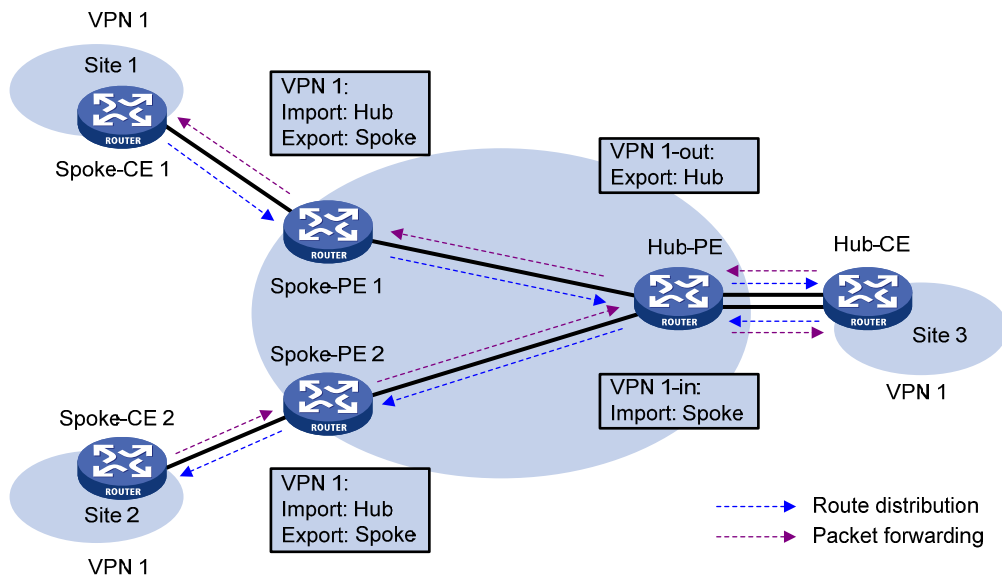
In a hub and spoke network as shown in [Figure 34](#), configure route targets as follows:

- On spoke PEs (PEs connected to spoke sites), set the export target to Spoke and the import target to Hub.
- On the hub PE (PE connected to the hub site), use two interfaces or subinterfaces that each belong to a different VPN instance to connect the hub CE. One VPN instance receives routes from spoke PEs and has the import target set to Spoke, and the other VPN instance advertises routes to spoke PEs and has the export target set to Hub.

These route targets rules produce the following results:

- The hub PE can receive all VPN-IPv4 routes from spoke PEs.
- All spoke PEs can receive VPN-IPv4 routes advertised by the hub PE.
- The hub PE advertises the routes learned from a spoke PE to the other spoke PEs so the spoke sites can communicate with each other through the hub site.
- The import target attribute of a spoke PE is different from the export target attribute of any other spoke PE. Therefore, any two spoke PEs cannot directly advertise VPN-IPv4 routes to each other or directly access each other.

Figure 34 Network diagram for hub and spoke network



A route in Site 1 is advertised to Site 2 using the following process:

1. Spoke-CE 1 advertises a route in Site 1 to Spoke-PE 1.
2. Spoke-PE 1 changes the route to a VPN-IPv4 route and advertises the VPN-IPv4 route to Hub-PE through MP-BGP.
3. Hub-PE adds the VPN-IPv4 route into the routing table of VPN 1-in, changes it to the original IPv4 route, and advertises the IPv4 route to Hub-CE.
4. Hub-CE advertises the IPv4 route back to Hub-PE.
5. Hub-PE adds the IPv4 route to the routing table of VPN 1-out, changes it to a VPN-IPv4 route, and advertises the VPN-IPv4 route to Spoke-PE 2 through MP-BGP.
6. Spoke-PE 2 changes the VPN-IPv4 route to the original IPv4 route, and advertises the IPv4 route to Site 2.

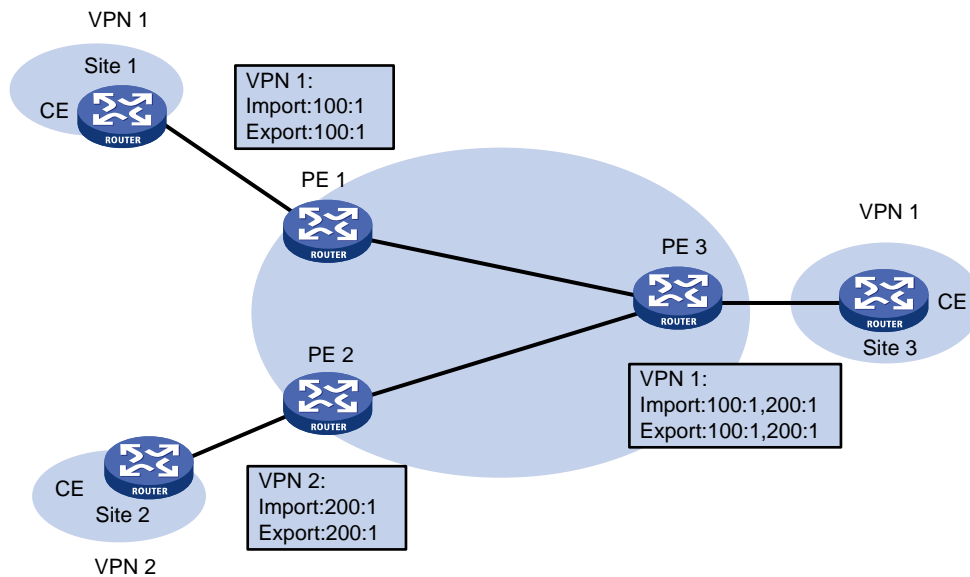
After spoke sites exchange routes through the hub site, they can communicate with each other through the hub site.

Extranet networking scheme

The extranet networking scheme allows specific resources in a VPN to be accessed by users not in the VPN.

In this networking scheme, if a VPN instance needs to access a shared site, the export target attribute and the import target attribute of the VPN instance must be contained in the import target attribute and the export target attribute of the VPN instance of the shared site, respectively.

Figure 35 Network diagram for extranet networking scheme



As shown in Figure 35, route targets configured on PEs produce the following results:

- PE 3 can receive VPN-IPv4 routes from PE 1 and PE 2.
- PE 1 and PE 2 can receive VPN-IPv4 routes advertised by PE 3.
- Site 1 and Site 3 of VPN 1 can communicate with each other, and Site 2 of VPN 2 and Site 3 of VPN 1 can communicate with each other.
- PE 3 advertises neither the VPN-IPv4 routes received from PE 1 to PE 2 nor the VPN-IPv4 routes received from PE 2 to PE 1 (routes learned from an IBGP neighbor are not advertised to any other IBGP neighbor). Therefore, Site 1 of VPN 1 and Site 2 of VPN 2 cannot communicate with each other.

Inter-AS VPN

In an inter-AS VPN networking scenario, multiple sites of a VPN are connected to multiple ISPs in different ASs, or to multiple ASs of an ISP.

RFC 2547bis presents the following inter-AS VPN solutions:

- **VRF-to-VRF**—ASBRs manage VPN routes between them through subinterfaces. This solution is also called "inter-AS option A."
- **EBGP redistribution of labeled VPN-IPv4 routes**—ASBRs advertise labeled VPN-IPv4 routes to each other through MP-EBGP. This solution is also called "inter-AS option B."
- **Multihop EBGP redistribution of labeled VPN-IPv4 routes**—PEs advertise labeled VPN-IPv4 routes to each other through MP-EBGP. This solution is also called "inter-AS option C."

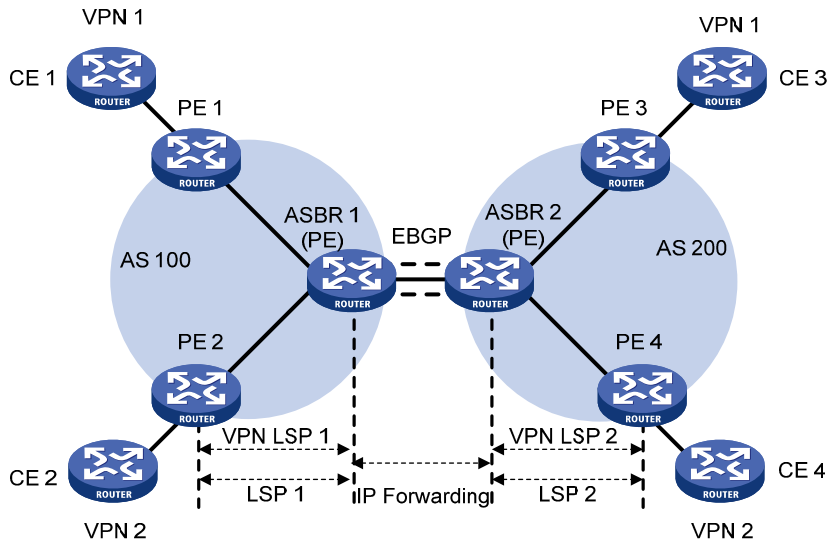
Inter-AS option A

In this solution, PEs of two ASs are directly connected, and each PE is also the ASBR of its AS.

The PEs acting as ASBRs are connected through multiple subinterfaces. Each of them treats the other as a CE and advertises IPv4 routes through conventional EBGP. Within an AS, packets are forwarded as VPN packets with two-level labels. Between ASBRs, conventional IP forwarding is used.

Ideally, each inter-AS VPN has a pair of subinterfaces to exchange VPN routing information.

Figure 36 Network diagram for inter-AS option A



Inter-AS option A is easy to carry out because no special configuration is required on the PEs acting as the ASBRs.

However, it has limited scalability because the PEs acting as the ASBRs must manage all the VPN routes and create VPN instances on a per-VPN basis. This leads to excessive VPN-IPv4 routes on the PEs. Creating a separate subinterface for each VPN also requires additional system resources.

Inter-AS option B

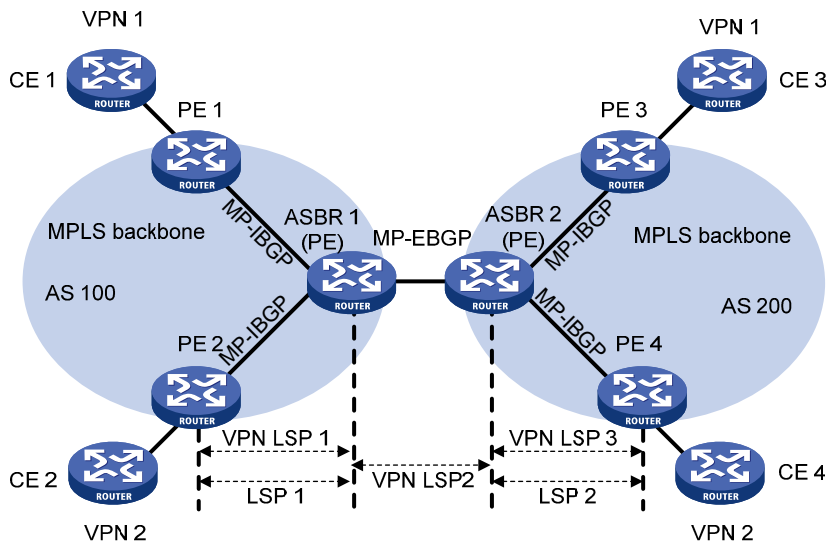
In this solution, two ASBRs use MP-EBGP to exchange labeled VPN-IPv4 routes that they obtain from the PEs in their respective ASs.

As shown in [Figure 37](#), the routes are advertised using the following process:

1. PEs in AS 100 advertise labeled VPN-IPv4 routes to the ASBR PE of AS 100 or the route reflector (RR) of the ASBR PE through MP-IBGP.
2. The ASBR PE advertises labeled VPN-IPv4 routes to the ASBR PE of AS 200 through MP-EBGP.
3. The ASBR PE of AS 200 advertises labeled VPN-IPv4 routes to PEs in AS 200 or to the RR of the PEs through MP-IBGP.

The ASBRs must perform special processing on the labeled VPN-IPv4 routes, which is also called ASBR extension method.

Figure 37 Network diagram for inter-AS option B



Inter-AS option B has better scalability than option A.

When adopting the MP-EBGP method, note the following:

- ASBRs do not perform route target filtering on VPN-IPv4 routes that they receive from each other. Therefore, the ISPs in different ASs must agree on the route exchange.
- VPN-IPv4 routes are exchanged only between VPN peers. A VPN site can exchange VPN-IPv4 routes neither with the public network nor with MP-EBGP peers with whom it has not reached agreement on the route exchange.

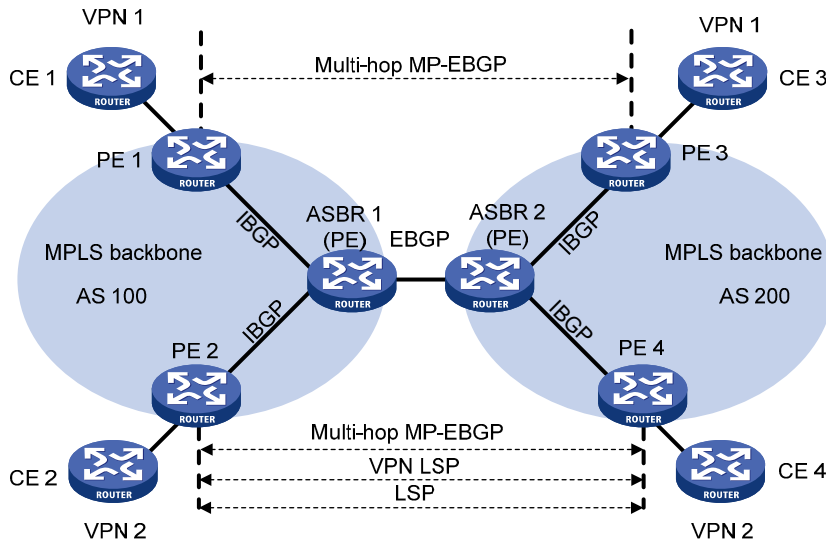
Inter-AS option C

The Inter-AS option A and option B solutions can meet the needs for inter-AS VPNs. However, they require that the ASBRs maintain and advertise VPN-IPv4 routes. When every AS needs to exchange a great amount of VPN routes, the ASBRs might become bottlenecks, which hinders network extension.

Inter-AS option C can solve the problem by making PEs directly exchange VPN-IPv4 routes without the participation of ASBRs:

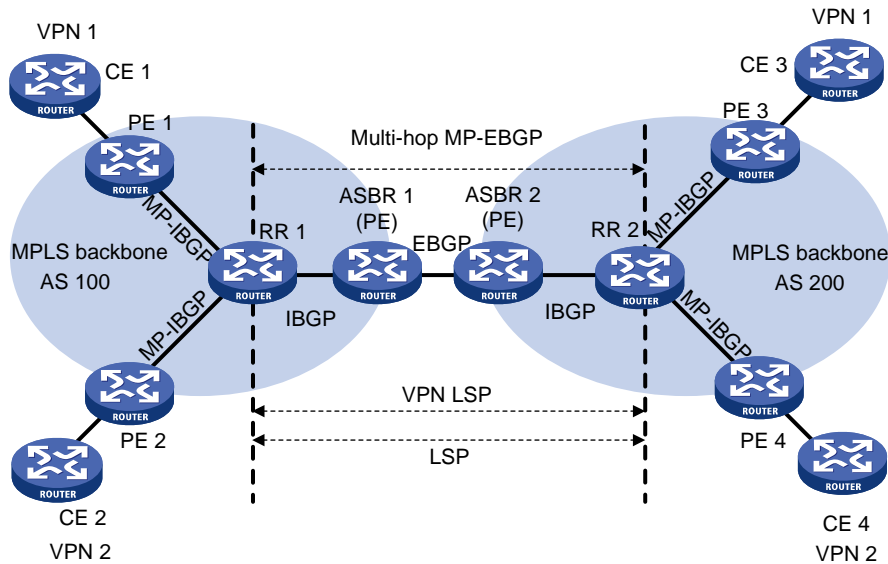
- Two ASBRs advertise labeled IPv4 routes to PEs in their respective ASs through IBGP.
- The ASBRs neither maintain VPN-IPv4 routes nor advertise VPN-IPv4 routes to each other.
- An ASBR maintains labeled IPv4 routes of the PEs in the AS and advertises them to the peers in the other ASs. The ASBR of another AS also advertises labeled IPv4 routes. Thus, an LSP is established between the ingress PE and egress PE.
- Between PEs of different ASs, multi-hop EBGP connections are established to exchange VPN-IPv4 routes.

Figure 38 Network diagram for inter-AS option C



To improve the scalability, you can specify an RR in each AS to maintain all VPN-IPv4 routes and to exchange VPN-IPv4 routes with PEs in the AS. The RRs in two ASs establish an inter-AS VPNv4 connection to advertise VPN-IPv4 routes, as shown in [Figure 39](#).

Figure 39 Network diagram for inter-AS option C using RRs



Carrier's carrier

If a customer of the MPLS L3VPN service provider is also a service provider, the MPLS L3VPN service provider is called the provider carrier or the Level 1 carrier, while the customer is called the customer carrier or the Level 2 carrier. This networking model is referred to as carrier's carrier. In this model, the Level 2 service provider serves as a CE of the Level 1 service provider.

For good scalability, the Level 1 carrier does not learn the routes of the customer network connected to a Level 2 carrier. It only learns the routes for delivering packets between different sites of the Level 2 carrier. Routes of the customer networks connected to a Level 2 carrier are exchanged through the BGP

session established between the routers of the Level 2 carrier. This can greatly reduce the number of routes maintained by the Level 1 carrier network.

Compared with the common MPLS L3VPN, the carrier's carrier is different because of the way in which a CE of a Level 1 carrier (a Level 2 carrier) accesses a PE of the Level 1 carrier:

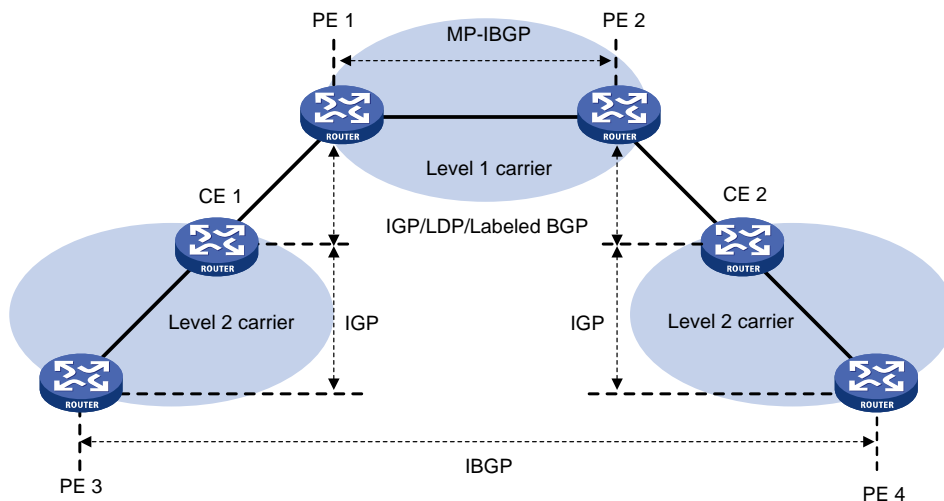
- If the PE and the CE are in a same AS, you must configure IGP and LDP between them.
- If the PE and the CE are not in the same AS, you must configure MP-EBGP to assign labels to routes exchanged between them.

In either case, you must enable MPLS on the CE of the Level 1 carrier. Moreover, the CE holds the VPN routes of the Level 2 carrier, but it does not advertise the routes to the PE of the Level 1 carrier. It only exchanges the routes with other PEs of the Level 2 carrier.

A Level 2 carrier can be an ordinary ISP or an MPLS L3VPN service provider.

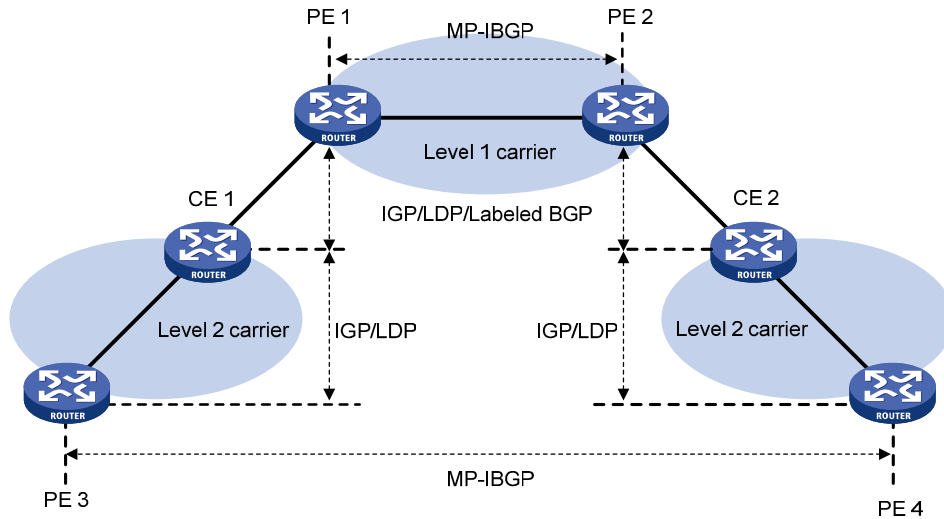
When the Level 2 carrier is an ordinary ISP, its PEs run IGP to communicate with the CEs, rather than MPLS. As shown in [Figure 40](#), PE 3 and PE 4 exchange VPN routes of the Level 2 carrier through an IBGP session.

Figure 40 Scenario where the Level 2 carrier is an ISP



When the Level 2 carrier is an MPLS L3VPN service provider, its PEs must run IGP and LDP to communicate with CEs. As shown in [Figure 41](#), PE 3 and PE 4 exchange VPN routes of the Level 2 carrier through an MP-IBGP session.

Figure 41 Scenario where the Level 2 carrier is an MPLS L3VPN service provider



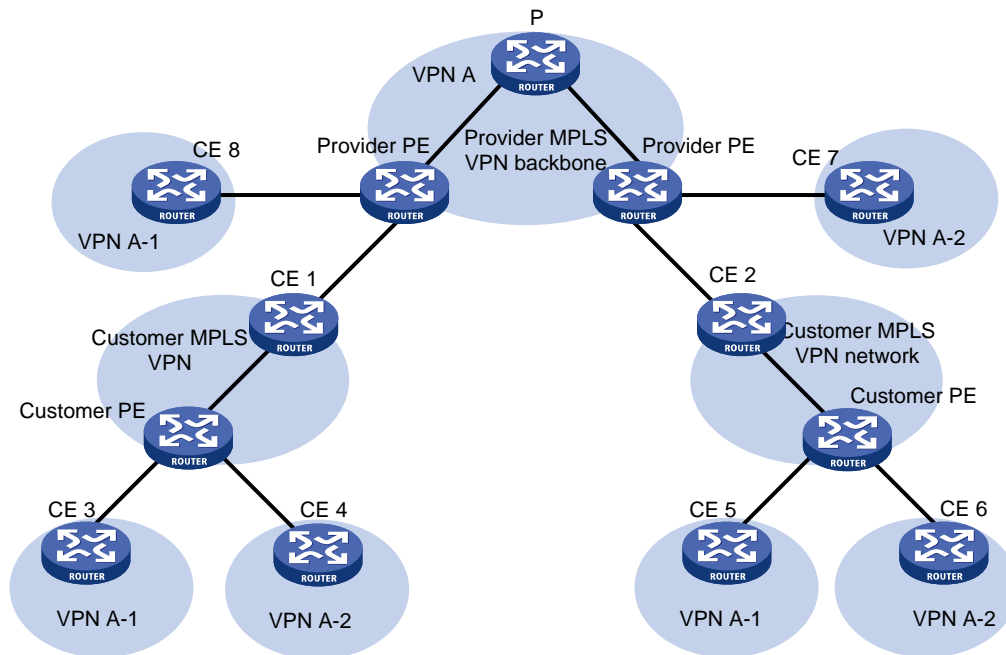
NOTE:

If equal cost routes exist between the Level 1 carrier and the Level 2 carrier, HP recommends that you establish equal cost LSPs between them.

Nested VPN

The nested VPN technology exchanges VPNv4 routes between PEs and CEs of the ISP MPLS L3VPN and allows a customer to manage its own internal VPNs. Figure 42 shows a nested VPN network. On the service provider's MPLS VPN network, there is a customer VPN named VPN A. The customer VPN contains two sub-VPNs, VPN A-1 and VPN A-2. The service provider PEs consider the customer's network as a common VPN user and do not join any sub-VPNs. The service provider CE devices (CE 1 and CE 2) exchange VPNv4 routes including sub-VPN routing information with the service provider PEs, which implements the propagation of the sub-VPN routing information throughout the customer network.

Figure 42 Network diagram for nested VPN



Propagation of routing information

In a nested VPN network, routing information is propagated using the following process:

1. A provider PE and its CEs exchange VPNv4 routes, which carry information about customer VPNs.
2. After receiving a VPNv4 route, a provider PE keeps the customer's internal VPN information, and appends the customer's MPLS VPN attributes on the service provider network. It replaces the RD of the VPNv4 route with the RD of the customer's MPLS VPN on the service provider network. It also adds the export route-target (ERT) attribute of the customer's MPLS VPN on the service provider network to the extended community attribute list of the route. The internal VPN information for the customer is maintained on the provider PE.
3. The provider PE advertises VPNv4 routes carrying the comprehensive VPN information to the other PEs of the service provider.
4. After another provider PE receives the VPNv4 routes, it matches the VPNv4 routes to the import targets of its local VPNs. Each local VPN accepts routes of its own and advertises them to provider CEs. If a provider CE (such as CE 7 and CE 8 in [Figure 42](#)) is connected to a provider PE through an IPv4 connection, the PE advertises IPv4 routes to the CE. If it is a VPNv4 connection (a customer MPLS VPN network), the PE advertises VPNv4 routes to the CE.
5. After receiving VPNv4 routes from the provider CE, a customer PE matches those routes to local import targets. Each customer VPN accepts only its own routes and advertises them to connected customer CEs (such as CE 3, CE 4, CE 5, and CE 6 in [Figure 42](#)).

Benefits

The nested VPN technology provides the following benefits:

- Support for VPN aggregation. It can aggregate a customer's internal VPNs into one VPN on the service provider's MPLS VPN network.
- Support for both symmetric networking and asymmetric networking. Sites of the same VPN can have the same number or different numbers of internal VPNs.
- Support for multiple-level nesting of internal VPNs.

Nested VPN is flexible and easy to implement. It reduces networking costs, provides diversified VPN networking methods for customers, and allows for multi-level hierarchical access control over internal VPNs.

HoVPN

In MPLS L3VPN solutions, PEs are the key devices, which provide the following functions:

- User access, requiring that the PEs must have a large number of interfaces.
- VPN route management and advertisement, and user packet processing, requiring that a PE must have a large-capacity memory and high forwarding capability.

Most network schemes use a typical hierarchical architecture. For example, the MAN architecture typically contains three layers: the core, distribution, and access. From the core layer to the access layer, the performance requirements on the devices decrease while the network expands.

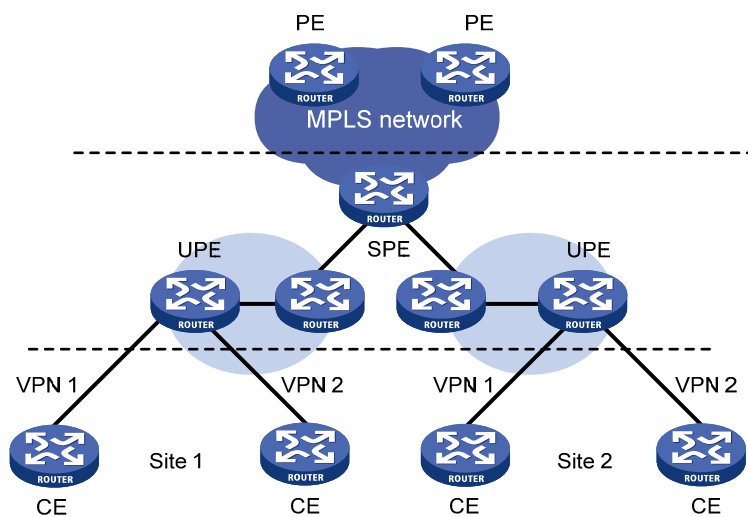
MPLS L3VPN, on the contrary, is a plane model where performance requirements are the same for all PEs. If a certain PE does not have enough performance or scalability, the performance or scalability of the whole network is influenced. Therefore, the plane model is not applicable to the large-scale VPN deployment.

To solve the scalability problem of the plane model, MPLS L3VPN must transition to the hierarchical model. Hierarchy of VPN (HoVPN) was proposed to meet the requirement. With HoVPN, the PE functions can be distributed among multiple PEs, which take different roles for the same functions and form a hierarchical architecture.

As in the typical hierarchical network model, HoVPN has different requirements on the devices at different layers of the hierarchy.

Implementation of HoVPN

Figure 43 Basic architecture of HoVPN



As shown in [Figure 43](#), devices directly connected to CEs are called underlayer PEs (UPEs) or user-end PEs, whereas devices that are connected to UPEs and are in the internal network are called superstratum PEs (SPE) or service provider-end PEs.

Multiple UPEs and SPEs comprise a hierarchical PE.

UPEs and SPEs play the following different roles:

- A UPE provides user access. It maintains the routes of directly connected VPN sites. It does not maintain the routes of the remote sites in the VPN, or it only maintains their summary routes. A UPE assigns inner labels to the routes of its directly connected sites, and advertises the labels along with VPN routes to the SPE through MP-BGP.
- An SPE manages and advertises VPN routes. It maintains all the routes of the VPNs connected through UPEs, including the routes of both the local and remote sites. An SPE advertises routes along with labels to UPEs, including the default routes of VPN instances or summary routes and the routes permitted by the routing policy. By using routing policies, you can control which sites in a VPN can communicate with each other.

Different roles mean different requirements:

- An SPE must have a large routing table capacity and high forwarding performance but needs fewer interface resources.
- A UPE must have higher access capability but needs a small routing table capacity and low forwarding performance.

HoVPN makes full use of both the high performance of SPEs and the high access capability of UPEs.

The concepts of SPE and UPE are relative. In the hierarchical PE architecture, a PE might be the SPE of its underlayer PEs and a UPE of its SPE at the same time.

The HoPE and common PEs can coexist in an MPLS network.

SPE-UPE

Either MP-IBGP or MP-EBGP can run between SPE and UPE.

For MP-IBGP to advertise routes between IBGP peers, the SPE acts as the RR and advertises routes from IBGP peer UPE to IBGP peer SPE. However, it does not act as the RR of the other PEs.

Recursion and extension of HoVPN

HoVPN supports HoPE recursion:

- A HoPE can act as a UPE to form a new HoPE with an SPE.
- A HoPE can act as an SPE to form a new HoPE with multiple UPEs.
- HoVPN supports multi-level recursion.

Figure 44 Recursion of HoPEs

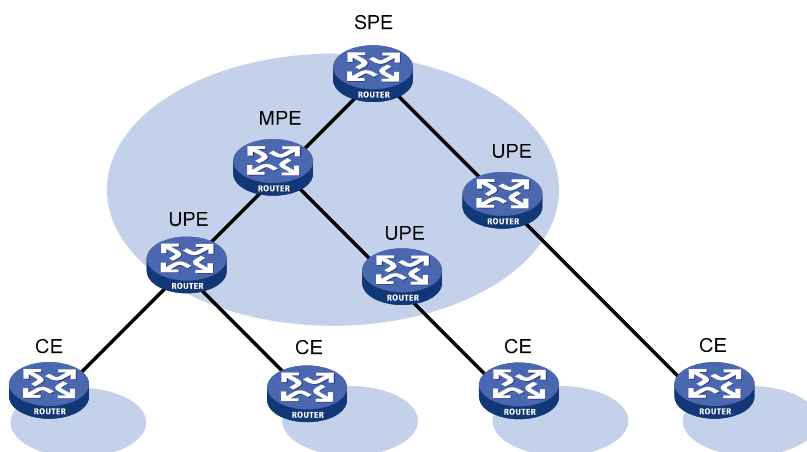


Figure 44 shows a three-level HoPE. The PE in the middle is called the "middle-level PE (MPE)." MP-BGP runs between SPE and MPE, and between MPE and UPE.

MP-BGP advertises all the VPN routes of UPEs to the SPEs, and advertises the default routes of the VPN instance of the SPEs or the VPN routes permitted by the routing policies to the UPEs.

The SPE maintains the VPN routes of all sites in the HoVPN. Each UPE maintains only VPN routes of its directly connected sites. An MPE has fewer routes than the SPE but has more routes than a UPE.

OSPF VPN extension

This section describes the OSPF VPN extension. For more information about OSPF, see *Layer 3—IP Routing Configuration Guide*.

OSPF for VPNs on a PE

OSPF is a commonly used IGP protocol. Running OSPF between a PE and a CE can simplify CE configuration and management because the CEs only need to support OSPF. In addition, if the customers require MPLS L3VPN services through a conventional OSPF backbone, using OSPF between a PE and a CE can simplify the transition.

For OSPF to run between CE and PE, the PE must support multiple OSPF processes. Each OSPF process corresponds to a VPN instance and maintains its own interfaces and routing table.

The following describes OSPF configurations between a PE and a CE:

- OSPF area configuration between a PE and a CE:

The OSPF area between a PE and a CE can be either a non-backbone area or a backbone area.

In the OSPF VPN extension application, the MPLS VPN backbone is considered the backbone area (area 0). The area 0 of each VPN site must be connected to the MPLS VPN backbone because OSPF requires that the backbone area be contiguous.

If a VPN site contains an OSPF area 0, the PE must be connected to the backbone area of the VPN site through area 0. You can configure a virtual link to connect the CE to the PE.

- BGP/OSPF interaction:

PEs advertise VPN routes to each other through BGP and to CEs through OSPF.

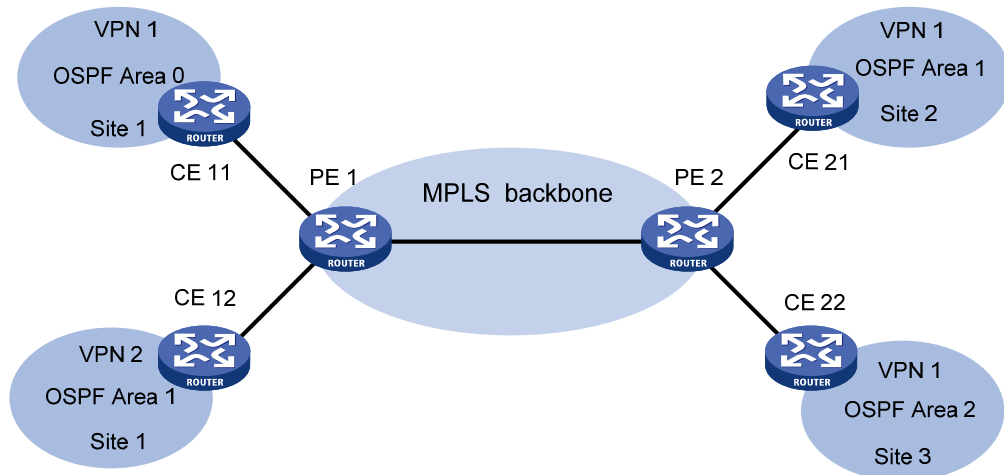
Conventional OSPF considers that two sites are in different ASs even if they belong to the same VPN. Therefore, the routes that one site learns are advertised to the other as external routes. This results in OSPF traffic and network management problems.

Extended OSPF supports multiple instances to address OSPF traffic and network management problems. When configured correctly, OSPF sites are considered directly connected, and PEs exchange OSPF routing information as they do on a dedicated line. This simplifies network management and makes OSPF applications more effective.

As shown in [Figure 45](#), PE 1 and PE 2 are connected through the MPLS backbone. CE 11, CE 21, and CE 22 belong to VPN 1. Assume that CE 11, CE 21, and CE 22 belong to the same OSPF domain. PEs advertise VPN 1 routes by using the following process:

- a. PE 1 redistributes OSPF routes of CE 11 into BGP.
- b. PE 1 advertises the VPN routes to PE 2 through BGP.
- c. PE 2 redistributes the BGP VPN routes into OSPF and advertises them to CE 21 and CE 22.

Figure 45 Application of OSPF in VPN



With the standard BGP/OSPF interaction, PE 2 advertises the BGP VPN routes to CE 21 and CE 22 in Type 5 LSAs (ASE LSAs). However, CE 11, CE 21, and CE 22 belong to the same OSPF domain, and route advertisements between them should use Type 3 LSAs (inter-area routes).

With the extended BGP/OSPF interaction, PEs advertise routes from one site to another site in Type 3 LSAs. The process requires that extended BGP community attributes include information for identifying the OSPF attributes.

Each OSPF domain must have a domain ID. HP recommends that you configure the same domain ID or adopt the default ID for all OSPF processes of the same VPN, so the system can know that VPN routes with the same domain ID are from the same VPN.

- Routing loop detection:

If a CE and a PE are connected through the OSPF backbone area, when a PE advertises BGP VPN routes learned from MPLS/BGP to the VPN site through LSAs, the LSAs might be received by another PE, resulting in a routing loop.

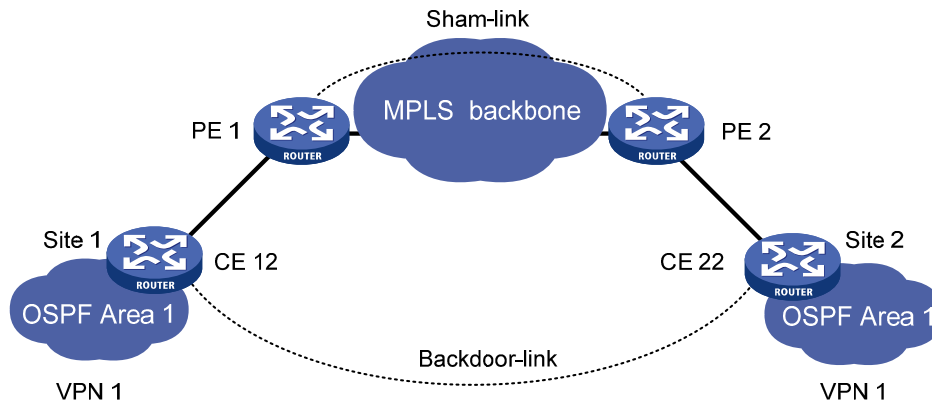
To avoid routing loops, when creating Type 3 LSAs, the PE always sets the flag bit DN for BGP VPN routes learned from MPLS/BGP, regardless of whether the PE and the CE are connected through the OSPF backbone. When performing route calculation, the OSPF process of the PE ignores the Type 3 LSAs whose DN bit is set.

If the PE needs to advertise routes from other OSPF domains to a CE, it must indicate that it is the ASBR, and advertise the routes in Type 5 LSAs.

OSPF sham link

On a PE, BGP routes received from the peer PE are redistributed into OSPF, and OSPF advertises these routes in Type 3 summary LSAs (inter-area routes) to the CE. As shown in Figure 46, both site 1 and site 2 belong to VPN 1 and OSPF area 1. Both an intra-area route (called a backdoor link) and an inter-area route exist between the two sites. The inter-area route is not preferred by OSPF because its priority is lower than the intra-area route priority.

Figure 46 Network diagram for sham link



To use the inter-area route, you can establish a sham link between the two PEs to change the inter-area route to an intra-area route. The sham link is advertised in a Type 1 LSA as an intra-area point-to-point link. You can also select the sham link or the backdoor link by adjusting their costs.

The sham link is considered a link between the two VPN instances. Each VPN instance has an endpoint address of the sham link, which is a loopback interface address with a 32-bit mask in the VPN address space. Different sham links of the same OSPF process can share an endpoint address, but sham links of different OSPF processes cannot share an endpoint address.

BGP advertises the endpoint addresses of sham links as VPN-IPv4 addresses. Sham link routes cannot be redistributed into BGP as VPN-IPv4 routes.

A sham link can be configured in any area and can only be manually configured. The local VPN instance must have a route to the destination of the sham link.

BGP AS number substitution

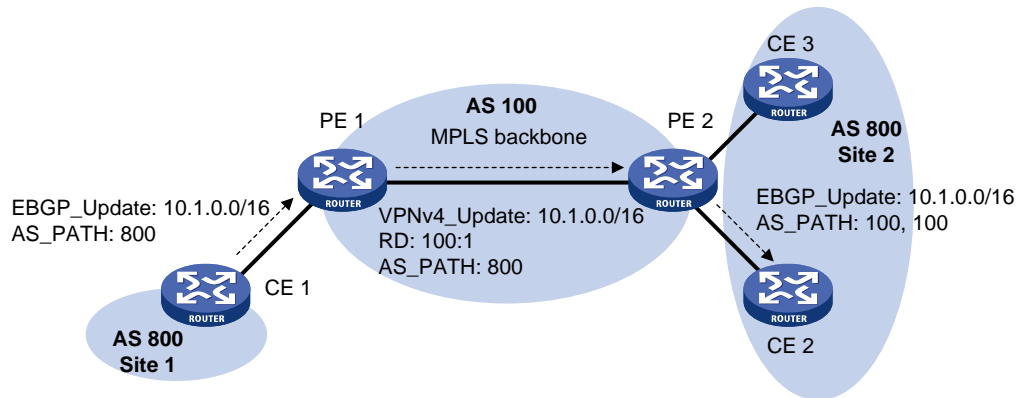
BGP detects routing loops by examining AS numbers. If EBGP runs between PE and CE, you must assign different AS numbers to geographically different sites to ensure correct transmission of routing information.

The BGP AS number substitution function allows physically dispersed CEs to use the same AS number. The function is a BGP outbound policy and affects routes to be advertised.

With the BGP AS number substitution function, when a PE advertises a route to a CE, if an AS number identical to that of the CE exists in the AS_PATH of the route, the PE replaces it with its own AS number.

After you enable the BGP AS number substitution function, the PE performs BGP AS number substitution for all routes and re-advertises them to connected CEs in the peer group.

Figure 47 Application of BGP AS number substitution



In Figure 47, both Site 1 and Site 2 use the AS number 800. AS number substitution is enabled on PE 2 for CE 2. Before advertising updates received from CE 1 to CE 2, PE 2 substitutes its own AS number 100 for the AS number 800. In this way, CE 2 can correctly receive the routing information from CE 1.

However, the AS number substitution function also introduces a routing loop in Site 2 because route updates originated from CE 3 can be advertised back to Site 2 through PE 2 and CE 2. To remove the routing loop, you can configure a routing policy on PE 2 to add the SoO attribute to route updates received from CE 2 and CE 3 so that PE 2 does not advertise route updates from CE 3 to CE 1.

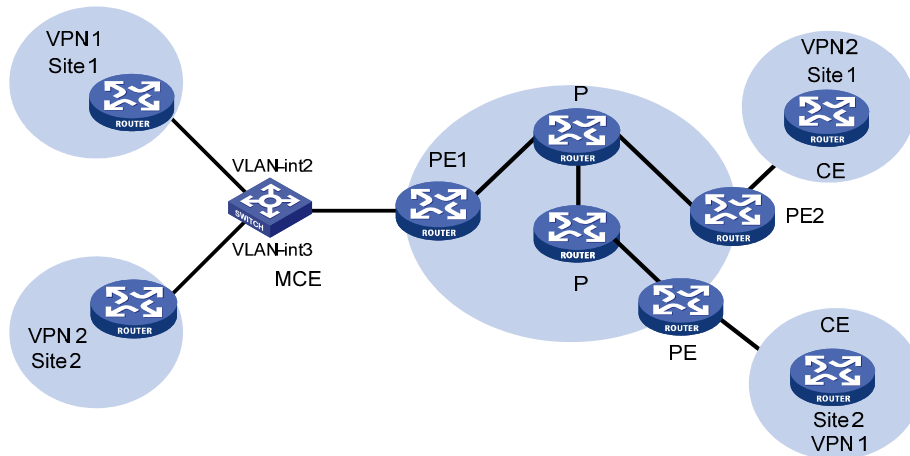
Multi-VPN-instance CE

BGP/MPLS VPN transmits private network data through MPLS tunnels over the public network. However, the traditional MPLS L3VPN architecture requires that each VPN instance use an exclusive CE to connect to a PE, as shown in Figure 30.

A private network is usually divided into multiple VPNs to isolate services. To meet these requirements, you can configure a CE for each VPN, which increases device expense and maintenance costs. Or, you can configure multiple VPNs to use the same CE and the same routing table, which sacrifices data security.

You can use the Multi-VPN-Instance CE (MCE) function in multi-VPN networks. MCE allows you to bind each VPN to a VLAN interface. The MCE creates and maintains a separate routing table for each VPN. This separates the forwarding paths of packets of different VPNs and, in conjunction with the PE, can correctly advertise the routes of each VPN to the peer PE, ensuring the normal transmission of VPN packets over the public network.

Figure 48 Network diagram for the MCE function



As shown in [Figure 48](#), the MCE device creates a routing table for each VPN. VLAN interface 2 binds to VPN 1 and VLAN-interface 3 binds to VPN 2. When receiving a route, the MCE device determines the source of the routing information according to the number of the receiving interface, and then adds it to the corresponding routing table. The MCE connects to PE 1 through a trunk link that permits packets tagged with VLAN 2 or VLAN 3. PE 1 determines the VPN that a received packet belongs to according to the VLAN tag of the packet, and sends the packet through the corresponding tunnel.

You can configure static routes, RIP, OSPF, IS-IS, EBGP, or IBGP between an MCE and a VPN site and between an MCE and a PE.

NOTE:

To implement dynamic IP assignment for DHCP clients in private networks, you can configure DHCP server or DHCP relay agent on the MCE. The IP address spaces for different private networks cannot overlap.

MPLS L3VPN configuration task list

Tasks at a glance

[Configuring basic MPLS L3VPN](#)

[Configuring inter-AS VPN](#)

[Configuring nested VPN](#)

[Configuring HoVPN](#)

[Configuring an OSPF sham link](#)

[Configuring routing on an MCE](#)

[Specifying the VPN label processing mode on the egress PE](#)

[Configuring BGP AS number substitution](#)

[Enabling SNMP notifications for MPLS L3VPN](#)

Configuring basic MPLS L3VPN

Tasks at a glance

Configuring VPN instances:

1. (Required.) [Creating a VPN instance](#)
2. (Required.) [Associating a VPN instance with an interface](#)
3. (Optional.) [Configuring route related attributes for a VPN instance](#)

(Required.) [Configuring routing between a PE and a CE](#)

(Required.) [Configuring routing between PEs](#)

(Optional.) [Configuring BGP VPNv4 route control](#)

Configuration prerequisites

Before you configure basic MPLS L3VPN, complete the following tasks:

- Configure an IGP for the MPLS backbone (on the PEs and Ps) to achieve IP connectivity.
- Configure basic MPLS for the MPLS backbone.
- Configure MPLS LDP for the MPLS backbone so that LDP LSPs can be established.
- Configure IP addresses for CE interfaces connected to PEs.

Configuring VPN instances

VPN instances isolate VPN routes from public network routes and routes among VPNs. This feature allows VPN instances to be used in network scenarios besides MPLS L3VPNs.

All VPN instance configurations are performed on PEs or MCEs.

Creating a VPN instance

A VPN instance is a collection of the VPN membership and routing rules of its associated site. A VPN instance might not correspond to one VPN.

To create and configure a VPN instance:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a VPN instance and enter VPN instance view.	ip vpn-instance <i>vpn-instance-name</i>	By default, no VPN instance is created. You can configure a maximum of 1023 VPN instances on a PE.
3. Configure an RD for the VPN instance.	route-distinguisher <i>route-distinguisher</i>	By default, no RD is specified for a VPN instance.
4. (Optional.) Configure a description for the VPN instance.	description <i>text</i>	By default, no description is configured for a VPN instance.

Step	Command	Remarks
5. (Optional.) Configure a VPN ID for the VPN instance.	vpn-id <i>vpn-id</i>	By default, no VPN ID is configured for a VPN instance.

Associating a VPN instance with an interface

After creating and configuring a VPN instance, associate the VPN instance with the interface connected to the CE.

To associate a VPN instance with an interface:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Associate a VPN instance with the interface.	ip binding vpn-instance <i>vpn-instance-name</i>	By default, no VPN instance is associated with an interface. The ip binding vpn-instance command deletes the IP address of the current interface. You must re-configure an IP address for the interface after configuring the command.

Configuring route related attributes for a VPN instance

VPN routes are controlled and advertised on a PE using the following process:

- When a VPN route learned from a site gets redistributed into BGP, BGP associates it with a route target extended community attribute list, which is usually the export target attribute of the VPN instance associated with the site.
- The VPN instance determines which routes it can accept and redistribute according to the **import-extcommunity** in the route target.
- The VPN instance determines how to change the route target attributes for routes to be advertised according to the **export-extcommunity** in the route target.

To configure route related attributes for a VPN instance:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter VPN instance view or IPv4 VPN view	<ul style="list-style-type: none"> • Enter VPN instance view: ip vpn-instance <i>vpn-instance-name</i> • Enter IPv4 VPN view: a. ip vpn-instance <i>vpn-instance-name</i> b. ipv4-family 	Configurations made in VPN instance view apply to both IPv4 VPN and IPv6 VPN. IPv4 VPN prefers the configurations in IPv4 VPN view over the configurations in VPN instance view.
3. Configure route targets.	vpn-target <i>vpn-target</i> <1-8> [both export-extcommunity import-extcommunity]	By default, no route targets are configured.

Step	Command	Remarks
4. Set the maximum number of routes allowed.	routing-table limit <i>number</i> { <i>warn-threshold</i> simply-alert }	The default setting depends on the device model. For more information, see the command in <i>MPLS Command Reference</i> . Setting the maximum number of routes for a VPN instance can prevent the PE from learning too many routes.
5. Apply an import routing policy.	import route-policy <i>route-policy</i>	By default, all routes matching the import target attribute are accepted. The specified routing policy must have been created. For information about routing policies, see <i>Layer 3—IP Routing Configuration Guide</i> .
6. Apply an export routing policy.	export route-policy <i>route-policy</i>	By default, routes to be advertised are not filtered. The specified routing policy must have been created. For information about routing policies, see <i>Layer 3—IP Routing Configuration Guide</i> .
7. Apply a tunnel policy to the VPN instance.	tnl-policy <i>tunnel-policy-name</i>	By default, only one tunnel is selected (no load balancing) in this order: LSP tunnel, GRE tunnel, and CR-LSP tunnel. The specified tunnel policy must have been created. For information about tunnel policies, see <i>Configuring tunnel policies</i> .

Configuring routing between a PE and a CE

You can configure static routing, RIP, OSPF, IS-IS, EBGp, or IBGP between a PE and a CE.

Configuring static routing between a PE and a CE

Step	Command	Remarks
1. Enter system view.	system-view	N/A

Step	Command	Remarks
2. Configure a static route for a VPN instance.	<pre> ip route-static vpn-instance s-vpn-instance-name dest-address { mask mask-length } { next-hop-address [public] [track track-entry-number] interface-type interface-number [next-hop-address] vpn-instance d-vpn-instance-name next-hop-address [track track-entry-number] } [permanent] [preference preference-value] [tag tag-value] [description description-text] </pre>	<p>By default, no static route is configured for a VPN instance.</p> <p>Perform this configuration on the PE. On the CE, configure a common static route.</p> <p>For more information about static routing, see <i>Layer 3—IP Routing Configuration Guide</i>.</p>

Configuring RIP between a PE and a CE

A RIP process belongs to the public network or a single VPN instance. If you create a RIP process without binding it to a VPN instance, the process belongs to the public network.

To configure RIP between a PE and a CE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a RIP process for a VPN instance and enter RIP view.	rip [process-id] vpn-instance vpn-instance-name	Perform this configuration on the PE. On the CE, create a common RIP process.
3. Enable RIP on the interface attached to the specified network.	network network-address	By default, RIP is disabled on an interface.

Configuring OSPF between a PE and a CE

An OSPF process that is bound to a VPN instance does not use the public network router ID configured in system view. Therefore, you must specify a router ID when starting a process or configure an IP address for at least one interface of the VPN instance.

An OSPF process belongs to the public network or a single VPN instance. If you create an OSPF process without binding it to a VPN instance, the process belongs to the public network.

To configure OSPF between a PE and a CE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A

Step	Command	Remarks
2. Create an OSPF process for a VPN instance and enter the OSPF view.	ospf [<i>process-id</i> router-id <i>router-id</i> vpn-instance <i>vpn-instance-name</i>] *	<p>Perform this configuration on the PE. On the CE, create a common OSPF process.</p> <p>The maximum number of OSPF processes that a VPN instance can run depends on the device's memory.</p> <p>Deleting a VPN instance also deletes all related OSPF processes.</p>
3. (Optional.) Configure an OSPF domain ID.	domain-id <i>domain-id</i> [secondary]	<p>The default domain ID is 0.</p> <p>Perform this configuration on the PE. On the CE, configure common OSPF.</p> <p>The domain ID is carried in the routes of the OSPF process. When redistributing routes from the OSPF process, BGP adds the domain ID as an extended community attribute into BGP VPN routes.</p> <p>An OSPF process can be configured with only one domain ID. Domain IDs of different OSPF processes are independent of each other.</p> <p>All OSPF processes of a VPN must be configured with the same domain ID, while OSPF processes on PEs in different VPNs can be configured with domain IDs as desired.</p>
4. Configure the type codes of OSPF extended community attributes.	ext-community-type { domain-id <i>type-code1</i> router-id <i>type-code2</i> route-type <i>type-code3</i> }	<p>The defaults are as follows:</p> <ul style="list-style-type: none"> • 0x0005 for Domain ID. • 0x0107 for Router ID. • 0x0306 for Route Type. <p>Perform this configuration on the PE.</p>
5. Create an OSPF area and enter area view.	area <i>area-id</i>	By default, no OSPF area is created.
6. Enable OSPF on the interface attached to the specified network in the area.	network <i>ip-address wildcard-mask</i>	By default, an interface neither belongs to any area nor runs OSPF.

Configuring IS-IS between a PE and a CE

An IS-IS process belongs to the public network or a single VPN instance. If you create an IS-IS process without binding it to a VPN instance, the process belongs to the public network.

To configure IS-IS between a PE and a CE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create an IS-IS process for a VPN instance and enter IS-IS view.	isis [<i>process-id</i>] vpn-instance <i>vpn-instance-name</i>	Perform this configuration on the PE. On the CE, configure common IS-IS.
3. Configure a network entity title for the IS-IS process.	network-entity <i>net</i>	By default, no NET is configured.
4. Return to system view.	quit	N/A
5. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
6. Enable the IS-IS process on the interface.	isis enable [<i>process-id</i>]	By default, no IS-IS process is enabled on the interface.

Configuring EBGP between a PE and a CE

1. Configure the PE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable BGP and enter BGP view.	bgp <i>as-number</i>	N/A
3. Enter BGP VPN view.	ip vpn-instance <i>vpn-instance-name</i>	Configuration commands in BGP VPN view are the same as those in BGP view. For details, see <i>Layer 3—IP Routing Configuration Guide</i> .
4. Configure the CE as the VPN EBGP peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	By default, no BGP peer is configured. For more information about BGP peers and peer groups, see <i>Layer 3—IP Routing Configuration Guide</i> .
5. Create and enter BGP VPN IPv4 unicast family view.	address-family ipv4 [unicast]	N/A
6. Enable IPv4 unicast route exchange with the specified peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange IPv4 unicast routes with any peer.
7. Redistribute the routes of the local CE.	import-route <i>protocol</i> [[{ <i>process-id</i> all-processes } [med <i>med-value</i> route-policy <i>route-policy-name</i>] *]	A PE must redistribute the routes of the local CE into its VPN routing table so it can advertise them to the peer PE.

Step	Command	Remarks
8. (Optional.) Allow the local AS number to appear in the AS_PATH attribute of a received route, and set the maximum number of repetitions.	peer { <i>group-name</i> <i>ip-address</i> } allow-as-loop [<i>number</i>]	By default, BGP discards incoming route updates that contain the local AS number. BGP detects routing loops by examining AS numbers. In a hub-spoke network where EBGP is running between a PE and a CE, the routing information the PE advertises to a CE carries the AS number of the PE. Therefore, the route updates that the PE receives from the CE also include the AS number of the PE. This causes the PE to be unable to receive the route updates. In this case, you must configure this command to allow routing loops.

2. Configure the CE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Configure the PE as a BGP peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	By default, no BGP peer is created.
4. Create and enter BGP IPv4 unicast family view.	address-family ipv4 [unicast]	N/A
5. Enable IPv4 unicast route exchange with the specified peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange IPv4 unicast routes with any peer.
6. (Optional.) Configure route redistribution.	import-route <i>protocol</i> [[{ <i>process-id</i> all-processes } [med <i>med-value</i> route-policy <i>route-policy-name</i>] *]	A CE must redistribute its routes to the PE so the PE can advertise them to the peer CE.

Configuring IBGP between a PE and a CE

Use IBGP between PE and CE only in a basic MPLS L3VPN network. In networks such as Hub&Spoke, Extranet, inter-AS VPN, carrier's carrier, nested VPN, and HoVPN, you cannot use IBGP between PE and CE.

1. Configure the PE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A

Step	Command	Remarks
3. Enter BGP VPN view.	ip vpn-instance <i>vpn-instance-name</i>	Configuration commands in BGP VPN view are the same as those in BGP view. For details, see <i>Layer 3—IP Routing Configuration Guide</i> .
4. Configure the CE as the VPN IBGP peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	By default, no BGP peer is created.
5. Create and enter BGP VPN IPv4 unicast family view.	address-family ipv4 [unicast]	N/A
6. Enable IPv4 unicast route exchange with the specified peer.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange IPv4 unicast routes with any peer.
7. Configure the CE as a client of the RR.	peer { <i>group-name</i> <i>ip-address</i> } reflect-client	By default, no RR or RR client is configured, and the PE does not advertise routes learned from the IBGP peer CE to other IBGP peers, including VPNv4 IBGP peers. The PE advertises routes learned from the CE to other IBGP peers only when you configure the IBGP peer CE as a client of the RR. Configuring an RR does not change the next hop of a route. To change the next hop of a route, configure an inbound policy on the receiving side.
8. (Optional.) Enable route reflection between clients.	reflect between-clients	Route reflection between clients is enabled by default.
9. (Optional.) Configure the cluster ID for the RR.	reflector cluster-id { <i>cluster-id</i> <i>ip-address</i> }	By default, the RR uses its own router ID as the cluster ID. If multiple RRs exist in a cluster, use this command to configure the same cluster ID for all RRs in the cluster to avoid routing loops.

2. Configure the CE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Configure the PE as an IBGP peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	By default, no BGP peer is created.
4. Create and enter BGP IPv4 unicast family view.	address-family ipv4 [unicast]	N/A

Step	Command	Remarks
5. Enable IPv4 unicast route exchange with the specified peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange IPv4 unicast routes with any peer.
6. (Optional.) Configure route redistribution.	import-route <i>protocol</i> [{ <i>process-id</i> all-processes } [med <i>med-value</i> route-policy <i>route-policy-name</i>] *]	A CE must redistribute its routes to the PE so the PE can advertise them to the peer CE.

Configuring routing between PEs

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Configure the remote PE as a BGP peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	By default, no BGP peer is created.
4. Specify the source interface for route updates.	peer { <i>group-name</i> <i>ip-address</i> } connect-interface <i>interface-type</i> <i>interface-number</i>	By default, BGP uses the egress interface of the optimal route destined for the peer as the source interface.
5. Enter BGP-VPNv4 address family view.	address-family vpn4	N/A
6. Enable BGP-VPNv4 route exchange with the specified peer.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange BGP-VPNv4 routes with any peer.

Configuring BGP VPNv4 route control

BGP VPNv4 route control is configured similarly with BGP route control, except that it is configured in BGP-VPNv4 address family view. For detailed information about BGP route control, see *Layer 3—IP Routing Configuration Guide*.

To configure BGP VPNv4 route control:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Enter BGP-VPNv4 address family view.	address-family vpn4	N/A
4. Configure filtering of advertised routes.	filter-policy { <i>acl-number</i> prefix-list <i>prefix-list-name</i> } export [<i>protocol process-id</i>]	Optional. By default, BGP does not filter advertised routes.
5. Configure filtering of received routes.	filter-policy { <i>acl-number</i> prefix-list <i>prefix-list-name</i> } import	Optional. By default, BGP does not filter received routes.

Step	Command	Remarks
6. Advertise community attributes to a peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } advertise-community	Optional. By default, no community attributes are advertised to any peer or peer group.
7. Allow the local AS number to appear in the AS_PATH attribute of routes received from the peer, and set the maximum number of repetitions.	peer { <i>group-name</i> <i>ip-address</i> } allow-as-loop [<i>number</i>]	By default, BGP discards route updates that contain the local AS number.
8. Filter routes received from or advertised to a peer or peer group based on an AS_PATH list.	peer { <i>group-name</i> <i>ip-address</i> } as-path-acl <i>aspath-filter-number</i> { import export }	Optional. By default, no AS filtering list is applied to a peer or peer group.
9. Advertise a default VPN route to a peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } default-route-advertise vpn-instance <i>vpn-instance-name</i>	Optional. By default, no default VPN route is advertised to a peer or peer group.
10. Apply an ACL to filter routes received from or advertised to a peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } filter-policy <i>acl-number</i> { export import }	Optional. By default, no ACL-based filtering is configured.
11. Save all route updates from a peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } keep-all-routes	By default, BGP does not save route updates from any peer.
12. Specify the router as the next hop of routes sent to a peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } next-hop-local	By default, the router sets itself as the next hop for routes sent to an EBGp peer or peer group, but it does not change the next hop for routes sent to an IBGP peer or peer group.
13. Configure BGP to not change the next hop of routes sent to an EBGp peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } next-hop-invariable	Optional. By default, the router sets itself as the next hop for routes sent to an EBGp peer or peer group. In an inter-AS option C network where an RR is used to advertise VPNv4 routes, configure this command on the RR so the RR does not change the next hop of routes sent to EBGp peers and clients.
14. Specify a preferred value for routes received from a peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } preferred-value <i>value</i>	Optional. By default, the preferred value is 0.
15. Apply a prefix list to filter routes received from or advertised to a peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } prefix-list <i>prefix-list-name</i> { export import }	By default, no prefix list based filtering is configured.

Step	Command	Remarks
16. Configure BGP updates advertised to an EBGP peer or peer group to carry only public AS numbers.	peer { <i>group-name</i> <i>ip-address</i> } public-as-only	Optional. By default, BGP route updates advertised to an EBGP peer or peer group can carry both public and private AS numbers.
17. Configure the router as a route reflector and specify a peer or peer group as its client.	peer { <i>group-name</i> <i>ip-address</i> } reflect-client	By default, no RR is configured.
18. Specify the maximum number of routes BGP can receive from a peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } route-limit <i>prefix-number</i> [{ alert-only reconnect <i>reconnect-time</i> } <i>percentage-value</i>] *	By default, the number of routes that BGP can receive from a peer or peer group is not limited.
19. Apply a routing policy to a peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } route-policy <i>route-policy-name</i> { export import }	By default, no routing policy is applied to a peer or peer group.
20. Enable route target-based filtering of received VPNv4 routes.	policy vpn-target	By default, this feature is enabled.
21. Enable route reflection between clients.	reflect between-clients	By default, route reflection between clients is enabled on the RR.
22. Configure a cluster ID for the route reflector.	reflector cluster-id { <i>cluster-id</i> <i>ip-address</i> }	By default, the RR uses its own router ID as the cluster ID.
23. Configure filtering of reflected routes.	rr-filter <i>extended-community-list-number</i>	By default, the RR does not filter reflected routes.

Configuring inter-AS VPN

If the MPLS backbone spans multiple ASs, you must configure inter-AS VPN.

Before you configure an inter-AS VPN, complete the following tasks:

- Configure an IGP for the MPLS backbones in each AS.
- Configure basic MPLS for the MPLS backbone of each AS.
- Configure MPLS LDP for the MPLS backbone of each AS so that LDP LSPs can be established.
- Configure basic MPLS L3VPN for each AS.

When configuring basic MPLS L3VPN for each AS, specific configurations might be required on PEs or ASBR PEs. This depends on the inter-AS VPN solution selected.

Configuring inter-AS option A

Inter-AS option A applies to scenarios with a few VPNs.

To configure inter-AS option A, create VPN instances on PEs and ASBR PEs. The VPN instances on PEs are used to allow CEs to access the network, and the VPN instances on ASBR PEs are used to access the peer ASBR PEs. An ASBR PE considers the peer ASBR PE as an CE.

The route targets configured on the PEs must match those configured on the ASBR-PEs in the same AS to make sure VPN routes sent by the PEs (or ASBR-PEs) can be received by the ASBR-PEs (or PEs). Route targets configured on the PEs in different ASs do not have such requirements.

For more information, see "[Configuring basic MPLS L3VPN.](#)"

Configuring inter-AS option B

Inter-AS option B requires that ASBR PEs maintain all VPNv4 routing information and advertise the information to peer ASBR PEs. The ASBR PEs must receive all VPNv4 routing information without performing route target-based filtering.

The route targets for the VPN instances on the PEs in different ASs must match for the same VPN.

An ASBR-PE always sets itself as the next hop of VPNv4 routes advertised to an MP-IBGP peer regardless of the **peer next-hop-local** command.

ASBR-PEs use BGP to assign labels and create BGP LSPs. There is no need to configure MPLS LDP between ASBR-PEs.

To configure inter-AS option B on an ASBR PE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view of the interface connecting to the remote ASBR-PE.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Configure the IP address of the interface.	ip address <i>ip-address</i> { <i>mask</i> <i>mask-length</i> }	N/A
4. Return to system view.	quit	N/A
5. Enter BGP view.	bgp <i>as-number</i>	N/A
6. Enter BGP-VPNv4 address family view.	address-family vpnv4	N/A
7. Disable route target based filtering of VPNv4 routes.	undo policy vpn-target	By default, the PE filters received VPNv4 routes by route targets. The routes surviving the filtering are added to the routing table, and the others are discarded.

Configuring inter-AS option C

To configure inter-AS option C, perform configurations on PEs and ASBR PEs, and configure routing policies on the ASBR PEs.

Configuring a PE

Establish an ordinary IBGP peer relationship between a PE and an ASBR PE in an AS, and an MP-EBGP peer relationship between PEs of different ASs.

The PEs and ASBR PEs in an AS must be able to exchange labeled IPv4 routes.

To configure a PE for inter-AS option C:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Configure the ASBR PE in the same AS as an IBGP peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	By default, no BGP peer is created.
4. Configure the PE of another AS as an EBGP peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	By default, no BGP peer is created.
5. Enter BGP IPv4 unicast address family view.	address-family ipv4 [unicast]	N/A
6. Enable the PE to exchange IPv4 unicast routes with the peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange IPv4 unicast routes with any peer.
7. Enable the PE to exchange labeled IPv4 routes with the ASBR PE in the same AS.	peer { <i>group-name</i> <i>ip-address</i> } label-route-capability	By default, BGP does not advertise labeled routes to any IPv4 peer or peer group.
8. Return to BGP view.	quit	N/A
9. Enter BGP-VPNv4 address family view.	address-family vpnv4	N/A
10. Enable the PE to exchange VPNv4 routes with the peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange VPNv4 routes with any peer.
11. (Optional.) Configure the PE to not change the next hop of routes advertised to the EBGP peer.	peer { <i>group-name</i> <i>ip-address</i> } next-hop-invariable	Configure this command on the RR so the RR does not change the next hop of advertised VPNv4 routes.

Configuring an ASBR PE

In the inter-AS option C solution, an inter-AS LSP is required, and the public network routes advertised between the relevant PEs and ASBRs must carry MPLS label information.

An ASBR-PE establishes common IBGP peer relationships with PEs in the same AS, and a common EBGP peer relationship with the peer ASBR PE. All of them can exchange labeled IPv4 routes.

Public network routes carrying MPLS labels are advertised through MP-BGP. According to RFC 3107 "Carrying Label Information in BGP-4," the label mapping information for a particular route is piggybacked in the same BGP update message that is used to distribute the route. This capability is implemented through BGP extended attributes and requires that BGP peers can handle labeled IPv4 routes.

To configure an ASBR PE for inter-AS option C:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Configure the PE in the same AS as an IBGP peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	By default, no BGP peer is created.
4. Configure the peer ASBR PE as an EBGP peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	By default, no BGP peer is created.

Step	Command	Remarks
5. Enter BGP IPv4 unicast address family view.	address-family ipv4 [unicast]	N/A
6. Enable exchange of IPv4 unicast routes with the peer or peer group.	peer { group-name ip-address } enable	By default, BGP does not exchange IPv4 unicast routes with any peer.
7. Enable exchange of labeled IPv4 routes with the PE in the local AS and the peer ASBR PE.	peer { group-name ip-address } label-route-capability	By default, BGP does not advertise labeled routes to any IPv4 peer or peer group.
8. Configure the ASBR PE to set itself as the next hop of routes advertised to the PE in the local AS.	peer { group-name ip-address } next-hop-local	By default, BGP does not use its address as the next hop of routes advertised to an IBGP peer or peer group.

Configuring a routing policy on an ASBR PE

A routing policy on an ASBR PE does the following:

- Assigns MPLS labels to routes received from the PEs in the local AS before advertising them to the peer ASBR PE.
- Assigns new MPLS labels to labeled IPv4 routes advertised to PEs in the local AS.

Which IPv4 routes are assigned with MPLS labels depends on the routing policy. Only routes that meet the criteria are assigned with labels. All other routes are still common IPv4 routes.

To configure a routing policy for inter-AS option C on an ASBR PE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a routing policy and enter routing policy view.	route-policy policy-name permit node seq-number	By default, no routing policy is created.
3. Match IPv4 routes carrying labels.	if-match mpls-label	By default, no match criterion is configured.
4. Set labels for IPv4 routes.	apply mpls-label	By default, no apply clause is configured.

Configuring nested VPN

For a network with many VPNs, nested VPN is a good solution to implement layered management of VPNs and to conceal the deployment of internal VPNs.

To build a nested VPN network, perform the following configurations:

- **Configurations between customer PE and customer CE**—Configure VPN instances on the customer PE and configure route exchange between customer PE and customer CE.
- **Configurations between customer PE and provider CE**—Configure BGP VPNv4 route exchange between them.
- **Configurations between provider CE and provider PE**—Configure VPN instances and enable nested VPN on the provider PE and configure BGP VPNv4 route exchange between the provider CE

and provider PE. To make sure the provider CE can receive all VPNv4 routes, configure the **undo policy vpn-target** command on the provider CE to not filter VPNv4 routes by RTs.

- **Configurations between provider PEs**—Configure BGP VPNv4 route exchange between them.

Nested VPN allows a customer PE to directly exchange VPNv4 routes with a provider PE, without needing to deploy a provider CE. In this case, the customer PE also acts as the provider CE. Therefore, you must configure provider CE settings on it.

Configurations on the customer CE, customer PE, and provider CE are similar to basic MPLS L3VPN configurations. This task describes the configurations on the provider PE.

When you configure nested VPN, follow these guidelines:

- The address spaces of sub-VPNs of a VPN cannot overlap.
- Do not assign nested VPN peers addresses that public network peers use.
- Nested VPN does not support multi-hop EBGP. A provider PE and a provider CE must use the addresses of the directly connected interfaces to establish a neighbor relationship.

To configure nested VPN:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Enter BGP-VPN VPNv4 address family view.	address-family vpnv4	N/A
4. Enable nested VPN.	nesting-vpn	By default, nested VPN is disabled.
5. Return to BGP view.	quit	N/A
6. Enter BGP-VPN view.	ip vpn-instance <i>vpn-instance-name</i>	N/A
7. Specify the peer CE or the peer group of the peer CE.	peer { <i>group-name</i> <i>peer-address</i> } as-number <i>as-number</i>	By default, no peer is specified.
8. Enter BGP-VPN VPNv4 address family view.	address-family vpnv4	N/A
9. Enable BGP VPNv4 route exchange with the peer CE or the peer group of the peer CE.	peer { <i>group-name</i> <i>peer-address</i> } enable	By default, BGP does not exchange VPNv4 routes with any peer.

Configuring HoVPN

HoVPN is suited to build hierarchical VPNs, reducing performance requirements for PEs.

Before you configure HoVPN, complete basic MPLS L3VPN settings on UPE and SPE.

Do not configure the **peer default-route-advertise vpn-instance** and **peer upe route-policy** commands at the same time.

Do not connect an SPE to a CE directly. If an SPE must be directly connected to a CE, the VPN instance on the SPE and that on the UPE must be configured with different RDs.

To configure HoVPN:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Specify a BGP peer or peer group.	peer { <i>group-name</i> <i>peer-address</i> } as-number <i>as-number</i>	By default, no BGP peer is specified.
4. Enter BGP-VPN VPNv4 address family view.	address-family vpnvp4	N/A
5. Enable BGP-VPNv4 route exchange with the peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange VPNv4 routes with any peer.
6. Specify the BGP peer or peer group as a UPE.	peer { <i>group-name</i> <i>ip-address</i> } upe	By default, no peer is a UPE.
7. Advertise routes to the UPE.	<ul style="list-style-type: none"> Advertise a default VPN route to the UPE: peer { <i>group-name</i> <i>ip-address</i> } default-route-advertise vpn-instance <i>vpn-instance-name</i> Advertise routes permitted by a routing policy to the UPE: peer { <i>group-name</i> <i>ip-address</i> } upe route-policy <i>route-policy-name</i> export 	<p>Use either command.</p> <p>By default, no route is advertised to the UPE.</p> <p>Do not configure both commands.</p> <p>The peer default-route-advertise vpn-instance command advertises a default route using the local address as the next hop to the UPE, regardless of whether the default route is present in the local routing table. However, if the specified peer is not a UPE, the command does not advertise a default route.</p>

Configuring an OSPF sham link

When a backdoor link exists between the two sites of a VPN, you can create a sham link between PEs to forward VPN traffic through the sham link on the backbone rather than the backdoor link. A sham link is considered an OSPF intra-area route.

The source and destination addresses of the sham link must be loopback interface addresses with 32-bit masks. The loopback interfaces must be bound to VPN instances, and their addresses are advertised through BGP.

Before you configure an OSPF sham link, complete the following tasks:

- Configure basic MPLS L3VPN (OSPF is used between PE and CE).
- Configure OSPF in the LAN where customer CEs reside.

Configuring a loopback interface

Step	Command	Remarks
1. Enter system view.	system-view	N/A

Step	Command	Remarks
2.	Create a loopback interface and enter loopback interface view. interface loopback <i>interface-number</i>	N/A
3.	Bind the loopback interface to a VPN instance. ip binding vpn-instance <i>vpn-instance-name</i>	By default, the interface is associated with no VPN instance.
4.	Configure the address of the loopback interface. ip address <i>ip-address</i> { <i>mask</i> <i>mask-length</i> }	N/A

Redistributing the loopback interface route

Step	Command	Remarks
1.	Enter system view. system-view	N/A
2.	Enter BGP view. bgp <i>as-number</i>	N/A
3.	Enter BGP-VPN view. ip vpn-instance <i>vpn-instance-name</i>	N/A
4.	Enter BGP-VPN IPv4 unicast address family view. address-family ipv4 [unicast]	N/A
5.	Redistribute direct routes into BGP (including the loopback interface route). import-route direct	By default, no direct routes are redistributed into BGP.

Creating a sham link

Step	Command	Remarks
1.	Enter system view. system-view	N/A
2.	Enter OSPF view. ospf [<i>process-id</i> router-id <i>router-id</i> vpn-instance <i>vpn-instance-name</i>] *	HP recommends that you specify a router ID.
3.	Configure the external route tag for imported VPN routes. route-tag <i>tag-value</i>	N/A
4.	Enter OSPF area view. area <i>area-id</i>	N/A
5.	Configure a sham link. sham-link <i>source-ip-address</i> <i>destination-ip-address</i> [cost <i>cost</i> dead <i>dead-interval</i> hello <i>hello-interval</i> { hmac-md5 md5 } <i>key-id</i> { cipher <i>cipher-string</i> plain <i>plain-string</i> } simple { cipher <i>cipher-string</i> plain <i>plain-string</i> } } retransmit <i>retrans-interval</i> trans-delay <i>delay</i>] *	By default, no sham link is configured.

Configuring routing on an MCE

MCE implements service isolation through route isolation. MCE routing configuration includes the following:

- MCE-VPN site routing configuration
- MCE-PE routing configuration

On the PE, disable routing loop detection to avoid route loss during route calculation, and disable route redistribution between routing protocols to save system resources.

Before you configure routing on an MCE, complete the following tasks:

- Configure VPN instances, and bind the VPN instances with the interfaces connected to the VPN sites and the PE.
- Configure the link layer and network layer protocols on related interfaces to ensure IP connectivity.

Configuring routing between an MCE and a VPN site

You can configure static routing, RIP, OSPF, IS-IS, EBGp or IBGP between an MCE and a VPN site.

Configuring static routing between an MCE and a VPN site

An MCE can reach a VPN site through a static route. Static routing on a traditional CE is globally effective and does not support address overlapping among VPNs. An MCE supports binding a static route to a VPN instance, so that the static routes of different VPN instances can be isolated from each other.

To configure a static route to a VPN site:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure a static route for a VPN instance.	ip route-static vpn-instance <i>s-vpn-instance-name</i> <i>dest-address</i> { <i>mask</i> <i>mask-length</i> } { <i>next-hop-address</i> [public] [track <i>track-entry-number</i>] <i>interface-type</i> <i>interface-number</i> [<i>next-hop-address</i>] vpn-instance <i>d-vpn-instance-name</i> <i>next-hop-address</i> [track <i>track-entry-number</i>] } [permanent] [preference <i>preference-value</i>] [tag <i>tag-value</i>] [description <i>description-text</i>]	By default, no static route is configured. Perform this configuration on the MCE. On the VPN site, configure a common static route.
3. (Optional.) Configure the default preference for static routes.	ip route-static default-preference <i>default-preference-value</i>	The default preference is 60.

Configuring RIP between an MCE and a VPN site

A RIP process belongs to the public network or a single VPN instance. If you create a RIP process without binding it to a VPN instance, the process belongs to the public network. Binding RIP processes to VPN

instances can isolate routes of different VPNs. For more information about RIP, see *Layer 3—IP Routing Configuration Guide*.

To configure RIP between an MCE and a VPN site:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a RIP process for a VPN instance and enter RIP view.	rip [<i>process-id</i>] vpn-instance <i>vpn-instance-name</i>	Perform this configuration on the MCE. On a VPN site, create a common RIP process.
3. Enable RIP on the interface attached to the specified network.	network <i>network-address</i>	By default, RIP is disabled on an interface.
4. Redistribute remote site routes advertised by the PE into RIP.	import-route <i>protocol</i> [<i>process-id</i>] [allow-ibgp] [cost <i>cost</i> route-policy <i>route-policy-name</i> tag <i>tag</i>] *	By default, no route is redistributed into RIP.
5. (Optional.) Configure the default cost value for the redistributed routes.	default cost <i>value</i>	The default cost is 0.

Configuring OSPF between an MCE and a VPN site

An OSPF process belongs to the public network or a single VPN instance. If you create an OSPF process without binding it to a VPN instance, the process belongs to the public network.

Binding OSPF processes to VPN instances can isolate routes of different VPNs. For more information about OSPF, see *Layer 3—IP Routing Configuration Guide*.

To configure OSPF between an MCE and a VPN site:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create an OSPF process for a VPN instance and enter OSPF view.	ospf [<i>process-id</i> router-id <i>router-id</i> vpn-instance <i>vpn-instance-name</i>] *	Perform this configuration on the MCE. On a VPN site, create a common OSPF process. An OSPF process bound to a VPN instance does not use the public network router ID configured in system view. Therefore, configure a router ID for the OSPF process. An OSPF process can belong to only one VPN instance, but one VPN instance can use multiple OSPF processes to advertise VPN routes.

Step	Command	Remarks
3. (Optional.) Configure the OSPF domain ID.	domain-id <i>domain-id</i> [secondary]	The default domain ID is 0. Perform this configuration on the MCE. All OSPF processes of the same VPN instance must be configured with the same OSPF domain ID to ensure correct route advertisement.
4. Redistribute remote site routes advertised by the PE into OSPF.	import-route <i>protocol</i> [<i>process-id</i> all-processes allow-ibgp] [cost <i>cost</i> route-policy <i>route-policy-name</i> tag <i>tag</i> type <i>type</i>] *	By default, no routes are redistributed into OSPF.
5. Create an OSPF area and enter OSPF area view.	area <i>area-id</i>	By default, no OSPF area is created.
6. Enable OSPF on the interface attached to the specified network in the area.	network <i>ip-address</i> <i>wildcard-mask</i>	By default, an interface neither belongs to any area nor runs OSPF.

Configuring IS-IS between an MCE and a VPN site

An IS-IS process belongs to the public network or a single VPN instance. If you create an IS-IS process without binding it to a VPN instance, the process belongs to the public network.

Binding IS-IS processes to VPN instances can isolate routes of different VPNs. For more information about IS-IS, see *Layer 3—IP Routing Configuration Guide*.

To configure IS-IS between an MCE and a VPN site:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create an IS-IS process for a VPN instance and enter IS-IS view.	isis [<i>process-id</i>] vpn-instance <i>vpn-instance-name</i>	Perform this configuration on the MCE. On a VPN site, configure a common IS-IS process.
3. Configure a network entity title.	network-entity <i>net</i>	By default, no NET is configured.
4. Redistribute remote site routes advertised by the PE into IS-IS.	import-route <i>protocol</i> [<i>process-id</i> all-processes allow-ibgp] [cost <i>cost</i> cost-type { external internal } [level-1 level-1-2 level-2] route-policy <i>route-policy-name</i> tag <i>tag</i>] *	By default, IS-IS does not redistribute routes from any other routing protocol. If you do not specify the route level in the command, the command redistributes routes to the level-2 routing table by default.
5. Return to system view.	quit	N/A
6. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
7. Enable the IS-IS process on the interface.	isis enable [<i>process-id</i>]	IS-IS is disabled by default.

Configuring EBGW between an MCE and a VPN site

To run EBGW between an MCE and a VPN site, you must configure a BGP peer for each VPN instance on the MCE, and redistribute the IGP routes of each VPN instance on the VPN site.

You can configure filtering policies to filter received routes and advertised routes.

1. Configure the MCE:

Routes redistributed from OSPF to BGP have their OSPF attributes removed. To enable BGP to distinguish routes redistributed from different OSPF domains, you must enable the redistributed routes to carry the OSPF domain ID by configuring the **domain-id** command in OSPF view. The domain ID is added to BGP VPN routes as an extended community attribute.

To configure the MCE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Enter BGP-VPN view.	ip vpn-instance <i>vpn-instance-name</i>	N/A
4. Configure an EBGW peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	By default, no BGP peer is configured.
5. Enter BGP-VPN IPv4 unicast address family view.	address-family ipv4 [unicast]	N/A
6. Enable BGP to exchange IPv4 unicast routes with the peer.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange IPv4 unicast routes with any peer.
7. Allow the local AS number to appear in the AS_PATH attribute of routes received from the peer, and set the maximum number of repetitions.	peer { <i>group-name</i> <i>ip-address</i> } allow-as-loop [<i>number</i>]	By default, BGP discards incoming route updates that contain the local AS number. BGP detects routing loops by examining AS numbers. The routing information the MCE advertises to a site carries the local AS number. Therefore, the route updates that the MCE receives from the site also include the local AS number. This causes the MCE to be unable to receive the route updates. In this case, you must configure this command to allow routing loops.
8. Redistribute remote site routes advertised by the PE into BGP.	import-route <i>protocol</i> [{ <i>process-id</i> all-processes } [med <i>med-value</i> route-policy <i>route-policy-name</i>] *]	By default, no routes are redistributed into BGP.

Step	Command	Remarks
9. (Optional.) Configure filtering of advertised routes.	filter-policy { <i>acl-number</i> prefix-list <i>prefix-list-name</i> } export [<i>protocol</i> <i>process-id</i>]	By default, BGP does not filter advertised routes.
10. (Optional.) Configure filtering of received routes.	filter-policy { <i>acl-number</i> prefix-list <i>prefix-list-name</i> } import	By default, BGP does not filter received routes.

2. Configure a VPN site:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Configure the MCE as an EBGP peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	N/A
4. Enter BGP-VPN IPv4 unicast address family view.	address-family ipv4 [unicast]	N/A
5. Enable BGP to exchange IPv4 unicast routes with the peer.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange IPv4 unicast routes with any peer.
6. Redistribute the IGP routes of the VPN into BGP.	import-route <i>protocol</i> [[<i>process-id</i> all-processes] [med <i>med-value</i> route-policy <i>route-policy-name</i>] *]	By default, no routes are redistributed into BGP. A VPN site must advertise the VPN network addresses it can reach to the connected MCE.

Configuring IBGP between MCE and VPN site

To run IBGP between an MCE and a VPN site, you must configure a BGP peer for each VPN instance on the MCE, and redistribute the IGP routes of each VPN instance on the VPN site.

1. Configure the MCE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Enter BGP-VPN view.	ip vpn-instance <i>vpn-instance-name</i>	N/A
4. Configure an IBGP peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	N/A
5. Enter BGP-VPN IPv4 unicast address family view.	address-family ipv4 [unicast]	N/A
6. Enable BGP to exchange IPv4 unicast routes with the peer.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange IPv4 unicast routes with any peer.

Step	Command	Remarks
7. (Optional.) Configure the system to be the RR, and specify the peer as the client of the RR.	peer { <i>group-name</i> <i>ip-address</i> } reflect-client	By default, no RR or RR client is configured. After you configure a VPN site as an IBGP peer, the MCE does not advertise the BGP routes learned from the VPN site to other IBGP peers, including VPNv4 peers. The MCE advertises routes learned from a VPN site only when you configure the VPN site as a client of the RR (the MCE).
8. Redistribute remote site routes advertised by the PE into BGP.	import-route <i>protocol</i> [<i>process-id</i> all-processes] [med <i>med-value</i> route-policy <i>route-policy-name</i>] *	By default, no routes are redistributed into BGP.
9. (Optional.) Configure filtering of advertised routes.	filter-policy { <i>acl-number</i> prefix-list <i>prefix-list-name</i> } export [<i>protocol</i> <i>process-id</i>]	By default, BGP does not filter advertised routes.
10. (Optional.) Configure filtering of received routes.	filter-policy { <i>acl-number</i> prefix-list <i>prefix-list-name</i> } import	By default, BGP does not filter received routes.

2. Configure a VPN site:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Configure the MCE as an IBGP peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	N/A
4. Enter BGP-VPN IPv4 unicast address family view.	address-family ipv4 [unicast]	N/A
5. Enable BGP to exchange IPv4 unicast routes with the peer.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange IPv4 unicast routes with any peer.
6. Redistribute the IGP routes of the VPN into BGP.	import-route <i>protocol</i> [{ <i>process-id</i> all-processes } [med <i>med-value</i> route-policy <i>route-policy-name</i>] *]	By default, no routes are redistributed into BGP. A VPN site must advertise VPN network addresses to the connected MCE.

Configuring routing between an MCE and a PE

MCE-PE routing configuration includes these tasks:

- Binding the MCE-PE interfaces to VPN instances.
- Performing route configurations.
- Redistributing VPN routes into the routing protocol running between the MCE and the PE.

Perform the following configurations on the MCE. For information about how to configure the PE, see "Configuring routing between a PE and a CE."

Configuring static routing between an MCE and a PE

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure a static route for a VPN instance.	ip route-static vpn-instance <i>s-vpn-instance-name</i> <i>dest-address</i> { <i>mask</i> <i>mask-length</i> } { <i>next-hop-address</i> [public] [track <i>track-entry-number</i>] <i>interface-type</i> <i>interface-number</i> [<i>next-hop-address</i>] } vpn-instance <i>d-vpn-instance-name</i> <i>next-hop-address</i> [track <i>track-entry-number</i>] } [permanent] [preference <i>preference-value</i>] [tag <i>tag-value</i>] [description <i>description-text</i>]	By default, no static route is configured.
3. (Optional.) Configure the default preference for static routes.	ip route-static default-preference <i>default-preference-value</i>	The default preference is 60.

Configuring RIP between an MCE and a PE

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a RIP process for a VPN instance and enter RIP view.	rip [<i>process-id</i>] vpn-instance <i>vpn-instance-name</i>	N/A
3. Enable RIP on the interface attached to the specified network.	network <i>network-address</i>	By default, RIP is disabled on an interface.
4. Redistribute the VPN routes.	import-route <i>protocol</i> [<i>process-id</i> all-processes allow-ibgp] [cost <i>cost</i> route-policy <i>route-policy-name</i> tag <i>tag</i>] *	By default, no routes are redistributed into RIP.
5. (Optional.) Configure the default cost for redistributed routes.	default cost <i>value</i>	The default cost is 0.

Configuring OSPF between an MCE and a PE

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create an OSPF process for a VPN instance and enter OSPF view.	ospf [<i>process-id</i> router-id <i>router-id</i> vpn-instance <i>vpn-instance-name</i>] *	N/A

Step	Command	Remarks
3. Disable routing loop detection.	vpn-instance-capability simple	By default, routing loop detection is enabled. You must disable routing loop detection for a VPN OSPF process on the MCE. Otherwise, the MCE cannot receive OSPF routes from the PE.
4. (Optional.) Configure the OSPF domain ID.	domain-id <i>domain-id</i> [secondary]	The default domain ID is 0.
5. Redistribute the VPN routes.	import-route <i>protocol</i> [<i>process-id</i> all-processes allow-ibgp] [cost <i>cost</i> route-policy <i>route-policy-name</i> tag <i>tag</i> type <i>type</i>] *	By default, no routes are redistributed into OSPF.
6. (Optional.) Configure filtering of advertised routes.	filter-policy { <i>acl-number</i> prefix-list <i>prefix-list-name</i> } export [<i>protocol</i> [<i>process-id</i>]]	By default, redistributed routes are not filtered.
7. (Optional.) Configure the default parameters for redistributed routes (cost, route number, tag, and type).	default { cost <i>cost</i> tag <i>tag</i> type <i>type</i> } *	The default cost is 1, the default tag is 1, and default type of redistributed routes is Type-2.
8. Create an OSPF area and enter OSPF area view.	area <i>area-id</i>	By default, no OSPF area is created.
9. Enable OSPF on the interface attached to the specified network in the area.	network <i>ip-address</i> <i>wildcard-mask</i>	By default, an interface neither belongs to any area nor runs OSPF.

Configuring IS-IS between an MCE and a PE

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create an IS-IS process for a VPN instance and enter IS-IS view.	isis [<i>process-id</i>] vpn-instance <i>vpn-instance-name</i>	N/A
3. Configure a network entity title.	network-entity <i>net</i>	By default, no NET is configured.
4. Redistribute VPN routes.	import-route <i>protocol</i> [<i>process-id</i> all-processes allow-ibgp] [cost <i>cost</i> cost-type { external internal }] [level-1 level-1-2 level-2] route-policy <i>route-policy-name</i> tag <i>tag</i>] *	By default, IS-IS does not redistribute routes from any other routing protocol. If you do not specify the route level in the command, the command redistributes routes to the level-2 routing table by default.
5. (Optional.) Configure filtering of advertised routes.	filter-policy { <i>acl-number</i> prefix-list <i>prefix-list-name</i> route-policy <i>route-policy-name</i> } export [<i>protocol</i> [<i>process-id</i>]]	By default, IS-IS does not filter advertised routes.
6. Return to system view.	quit	N/A

Step	Command	Remarks
7. Enter interface view.	interface <i>interface-type interface-number</i>	N/A
8. Enable the IS-IS process on the interface.	isis enable [<i>process-id</i>]	By default, no IS-IS process is enabled.

Configuring EBGP between an MCE and a PE

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Enter BGP-VPN view.	ip vpn-instance <i>vpn-instance-name</i>	N/A
4. Configure the PE as an EBGp peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	N/A
5. Enter BGP-VPN IPv4 unicast address family view.	address-family ipv4 [unicast]	N/A
6. Enable BGP to exchange IPv4 unicast routes with the peer.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange IPv4 unicast routes with any peer.
7. Redistribute the VPN routes of the VPN site.	import-route <i>protocol</i> [<i>process-id</i> all-processes] [med <i>med-value</i> route-policy <i>route-policy-name</i>] *	By default, no routes are redistributed into BGP.
8. (Optional.) Configure filtering of advertised routes.	filter-policy { <i>acl-number</i> prefix-list <i>prefix-list-name</i> } export [<i>protocol</i> <i>process-id</i>]	By default, BGP does not filter advertised routes.
9. (Optional.) Configure filtering of received routes.	filter-policy { <i>acl-number</i> prefix-list <i>prefix-list-name</i> } import	Optional. By default, BGP does not filter received routes.

Configuring IBGP between an MCE and a PE

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Enter BGP-VPN view.	ip vpn-instance <i>vpn-instance-name</i>	N/A
4. Configure the PE as an IBGP peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	N/A
5. Enter BGP-VPN IPv4 unicast address family view.	address-family ipv4 [unicast]	N/A
6. Enable BGP to exchange IPv4 unicast routes with the peer.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange IPv4 unicast routes with any peer.

Step	Command	Remarks
7. Redistribute the VPN routes of the VPN site.	import-route <i>protocol</i> [<i>process-id</i> all-processes] [med <i>med-value</i> route-policy <i>route-policy-name</i>] *	By default, no routes are redistributed into BGP.
8. (Optional.) Configure filtering of advertised routes.	filter-policy { <i>acl-number</i> prefix-list <i>prefix-list-name</i> } export [<i>protocol</i> <i>process-id</i>]	By default, BGP does not filter advertised routes.
9. (Optional.) Configure filtering of received routes.	filter-policy { <i>acl-number</i> prefix-list <i>prefix-list-name</i> } import	Optional. By default, BGP does not filter received routes.

Specifying the VPN label processing mode on the egress PE

An egress PE can process VPN labels in either POPGO or POP mode:

- **POPGO forwarding**—Pops the label and forwards the packet out of the egress interface corresponding to the label.
- **POP forwarding**—Pops the label and forwards the packet through the FIB table.

To specify the VPN label processing mode on an egress PE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Specify the VPN label processing mode as POPGO forwarding.	vpn popgo	The default is POP forwarding.

Configuring BGP AS number substitution

When CEs at different sites have the same AS number, configure the BGP AS number substitution function to avoid route loss. If the AS_PATH attribute of a route contains the AS number of the specified CE, the PE replaces the AS number with its own AS number before advertising the route to that CE.

Before you configure BGP AS number substitution, complete basic MPLS L3VPN configuration.

To configure BGP AS number substitution:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Enter BGP-VPN view.	ip vpn-instance <i>vpn-instance-name</i>	N/A
4. Configure a BGP peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	N/A

Step	Command	Remarks
5. Enable the BGP AS number substitution function.	peer { <i>ip-address</i> <i>group-name</i> } substitute-as	By default, BGP AS number substitution is disabled. For more information about this command, see <i>Layer 3—IP Routing Command Reference</i> .

Enabling SNMP notifications for MPLS L3VPN

This feature enables generating SNMP notifications for MPLS L3VPN when important events occur (for example, when the maximum number of routes in a VPN instance is exceeded), as defined in RFC 4382. The generated SNMP notifications are sent to the SNMP module. The SNMP module determines how to output the notifications according to the configured output rules.

For more information about SNMP notifications, see *Network Management and Monitoring Configuration Guide*.

To enable SNMP notifications for MPLS L3VPN:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable SNMP notifications for MPLS L3VPN.	snmp-agent trap enable l3vpn	By default, SNMP notifications for MPLS L3VPN are enabled.

Displaying and maintaining MPLS L3VPN

You can soft-reset or reset BGP sessions to apply new BGP configurations. A soft reset operation updates BGP routing information without tearing down BGP connections. A reset operation updates BGP routing information by tearing down, and then reestablishing BGP connections. Soft reset requires that BGP peers have route refresh capability.

Use the following commands to soft reset or reset BGP sessions:

Task	Command	Remarks
Soft reset BGP sessions for VPNv4 address family.	refresh bgp { <i>ip-address</i> all external group <i>group-name</i> internal } { export import } vpn4 [vpn-instance <i>vpn-instance-name</i>]	Available in user view.
Reset BGP sessions for VPNv4 address family.	reset bgp { <i>as-number</i> <i>ip-address</i> all external internal group <i>group-name</i> } vpn4 [vpn-instance <i>vpn-instance-name</i>]	Available in user view.

Use the following commands to display MPLS L3VPN:

Task	Command	Remarks
Display the routing table for a VPN instance. For more information about this command, see <i>Layer 3—IP Routing Command Reference</i> .	display ip routing-table vpn-instance <i>vpn-instance-name</i> [statistics verbose]	Available in any view.
Display information about a specific or all VPN instances.	display ip vpn-instance [instance-name <i>vpn-instance-name</i>]	Available in any view.
Display the FIB of a VPN instance.	display fib vpn-instance <i>vpn-instance-name</i>	Available in any view.
Display FIB entries that match the specified destination IP address in the specified VPN instance.	display fib vpn-instance <i>vpn-instance-name</i> <i>ip-address</i> [<i>mask</i> <i>mask-length</i>]	Available in any view.
Display BGP VPNv4 peer group information.	display bgp group vpnv4 [vpn-instance <i>vpn-instance-name</i>] [<i>group-name</i>]	Available in any view.
Display BGP VPNv4 peer information.	display bgp peer vpnv4 [vpn-instance <i>vpn-instance-name</i>] [<i>group-name</i> log-info <i>ip-address</i> { log-info verbose } verbose]	Available in any view.
Display BGP VPNv4 routes.	display bgp routing-table vpnv4 [route-distinguisher <i>route-distinguisher</i>] [<i>network-address</i> [{ <i>mask</i> <i>mask-length</i> }] [longest-match]]]	Available in any view.
Display BGP VPNv4 route advertisement information.	display bgp routing-table vpnv4 [route-distinguisher <i>route-distinguisher</i>] <i>network-address</i> [<i>mask</i> <i>mask-length</i>] advertise-info	Available in any view.
Display BGP VPNv4 routes matching the specified AS PATH list.	display bgp routing-table vpnv4 [route-distinguisher <i>route-distinguisher</i>] as-path-acl <i>as-path-acl-number</i>	Available in any view.
Display BGP VPNv4 routes matching the specified BGP community list.	display bgp routing-table vpnv4 [route-distinguisher <i>route-distinguisher</i>] community-list { { <i>basic-community-list-number</i> <i>comm-list-name</i> } [whole-match] <i>adv-community-list-number</i> }	Available in any view.
Display BGP VPNv4 routes advertised to or received from the specified BGP peer.	display bgp routing-table vpnv4 [vpn-instance <i>vpn-instance-name</i>] peer <i>ip-address</i> { advertised-routes received-routes } [<i>network-address</i> [<i>mask</i> <i>mask-length</i>]] statistics]	Available in any view.
Display incoming labels for BGP IPv4 unicast routes.	display bgp routing-table ipv4 [unicast] [vpn-instance <i>vpn-instance-name</i>] inlabel	Available in any view.
Display outgoing labels for BGP IPv4 unicast routes.	display bgp routing-table ipv4 [unicast] [vpn-instance <i>vpn-instance-name</i>] outlabel	Available in any view.
Display incoming labels for BGP VPNv4 routes.	display bgp routing-table vpnv4 inlabel	Available in any view.
Display outgoing labels for BGP VPNv4 routes.	display bgp routing-table vpnv4 outlabel	Available in any view.

Task	Command	Remarks
Display BGP VPNv4 route statistics.	display bgp routing-table vpnv4 statistics	Available in any view.
Display BGP VPNv4 address family update group information.	display bgp update-group vpnv4 [vpn-instance <i>vpn-instance-name</i>] [<i>ip-address</i>]	Available in any view.
Display OSPF sham link information. (MSR2000/MSR3000)	display ospf [<i>process-id</i>] sham-link [area <i>area-id</i>]	Available in any view.
Display OSPF sham link information. (MSR4000)	display ospf [<i>process-id</i>] sham-link [area <i>area-id</i>] [standby slot <i>slot-number</i>]	Available in any view.

MPLS L3VPN configuration examples

Configuring basic MPLS L3VPN

Network requirements

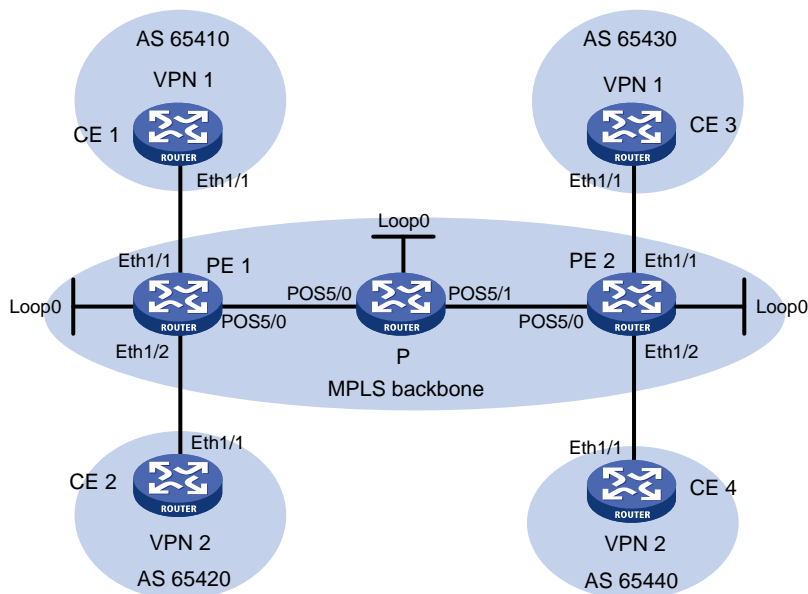
CE 1 and CE 3 belong to VPN 1. CE 2 and CE 4 belong to VPN 2.

VPN 1 uses route target attribute 111:1. VPN 2 uses route target attribute 222:2. Users of different VPNs cannot access each other.

A PE and its connected CE use EBGP to exchange VPN routing information.

PEs use OSPF to communicate with each other and use MP-IBGP to exchange VPN routing information.

Figure 49 Network diagram



Device	Interface	IP address	Device	Interface	IP address
CE 1	Eth1/1	10.1.1.1/24	P	Loop0	2.2.2.9/32
PE 1	Loop0	1.1.1.9/32	PE 2	POS5/0	172.1.1.2/24

	Eth1/1	10.1.1.2/24		POS5/1	172.2.1.1/24
	Eth1/2	10.2.1.2/24	PE 2	Loop0	3.3.3.9/32
	POS5/0	172.1.1.1/24		Eth1/1	10.3.1.2/24
CE 2	Eth1/1	10.2.1.1/24		Eth1/2	10.4.1.2/24
CE 3	Eth1/1	10.3.1.1/24		POS5/0	172.2.1.2/24
CE 4	Eth1/1	10.4.1.1/24			

Configuration procedure

1. Configure OSPF on the MPLS backbone to ensure IP connectivity within the backbone:

Configure PE 1.

```
<PE1> system-view
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 1.1.1.9 32
[PE1-LoopBack0] quit
[PE1] interface pos 5/0
[PE1-POS5/0] ip address 172.1.1.1 24
[PE1-POS5/0] quit
[PE1] ospf
[PE1-ospf-1] area 0
[PE1-ospf-1-area-0.0.0.0] network 172.1.1.0 0.0.0.255
[PE1-ospf-1-area-0.0.0.0] network 1.1.1.9 0.0.0.0
[PE1-ospf-1-area-0.0.0.0] quit
[PE1-ospf-1] quit
```

Configure the P device.

```
<P> system-view
[P] interface loopback 0
[P-LoopBack0] ip address 2.2.2.9 32
[P-LoopBack0] quit
[P] interface pos 5/0
[P-POS5/0] ip address 172.1.1.2 24
[P-POS5/0] quit
[P] interface pos 5/1
[P-POS5/1] ip address 172.2.1.1 24
[P-POS5/1] quit
[P] ospf
[P-ospf-1] area 0
[P-ospf-1-area-0.0.0.0] network 172.1.1.0 0.0.0.255
[P-ospf-1-area-0.0.0.0] network 172.2.1.0 0.0.0.255
[P-ospf-1-area-0.0.0.0] network 2.2.2.9 0.0.0.0
[P-ospf-1-area-0.0.0.0] quit
[P-ospf-1] quit
```

Configure PE 2.

```
<PE2> system-view
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 3.3.3.9 32
[PE2-LoopBack0] quit
[PE2] interface pos 5/0
```

```

[PE2-POS5/0] ip address 172.2.1.2 24
[PE2-POS5/0] quit
[PE2] ospf
[PE2-ospf-1] area 0
[PE2-ospf-1-area-0.0.0.0] network 172.2.1.0 0.0.0.255
[PE2-ospf-1-area-0.0.0.0] network 3.3.3.9 0.0.0.0
[PE2-ospf-1-area-0.0.0.0] quit
[PE2-ospf-1] quit

```

After the configurations, OSPF adjacencies are established between PE 1, P, and PE 2. Execute the **display ospf peer** command. The output shows that the adjacency status is Full. Execute the **display ip routing-table** command. The output shows that the PEs have learned the routes to the loopback interfaces of each other. Take PE 1 as an example:

```
[PE1] display ip routing-table protocol ospf
```

```
Summary Count : 5
```

```
OSPF Routing table Status : <Active>
```

```
Summary Count : 3
```

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
2.2.2.9/32	OSPF	10	1	172.1.1.2	POS5/0
3.3.3.9/32	OSPF	10	2	172.1.1.2	POS5/0
172.2.1.0/24	OSPF	10	2	172.1.1.2	POS5/0

```
OSPF Routing table Status : <Inactive>
```

```
Summary Count : 2
```

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
1.1.1.9/32	OSPF	10	0	1.1.1.9	Loop0
172.1.1.0/24	OSPF	10	1	172.1.1.1	POS5/0

```
[PE1] display ospf peer verbose
```

```
OSPF Process 1 with Router ID 1.1.1.9
```

```
Neighbors
```

```
Area 0.0.0.0 interface 172.1.1.1(POS5/0)'s neighbors
```

```
Router ID: 2.2.2.9          Address: 172.1.1.2          GR State: Normal
```

```
State: Full Mode: Nbr is Master Priority: 1
```

```
DR: 172.1.1.2 BDR: 172.1.1.1 MTU: 0
```

```
Options is 0x02 (-|-|-|-|-|E|-)
```

```
Dead timer due in 39 sec
```

```
Neighbor is up for 00:00:29
```

```
Authentication Sequence: [ 0 ]
```

```
Neighbor state change count: 6
```

2. Configure basic MPLS and MPLS LDP on the MPLS backbone to establish LDP LSPs:

Configure PE 1.

```
[PE1] mpls lsr-id 1.1.1.9
```

```
[PE1] mpls ldp
```

```
[PE1-ldp] quit
[PE1] interface pos 5/0
[PE1-POS5/0] mpls enable
[PE1-POS5/0] mpls ldp enable
[PE1-POS5/0] quit
```

Configure the P device.

```
[P] mpls lsr-id 2.2.2.9
[P] mpls ldp
[P-ldp] quit
[P] interface pos 5/0
[P-POS5/0] mpls enable
[P-POS5/0] mpls ldp enable
[P-POS5/0] quit
[P] interface pos 5/1
[P-POS5/1] mpls enable
[P-POS5/1] mpls ldp enable
[P-POS5/1] quit
```

Configure PE 2.

```
[PE2] mpls lsr-id 3.3.3.9
[PE2] mpls ldp
[PE2-ldp] quit
[PE2] interface pos 5/0
[PE2-POS5/0] mpls enable
[PE2-POS5/0] mpls ldp enable
[PE2-POS5/0] quit
```

After the configurations, LDP sessions are established between PE 1, P, and PE 2. Execute the **display mpls ldp peer** command. The output shows that the session status is Operational. Execute the **display mpls ldp lsp** command. The output shows the LSPs established by LDP. Take PE 1 as an example:

```
[PE1] display mpls ldp peer
Total number of peers: 1
Peer LDP ID      State          LAM  Role    GR   MD5  KA Sent/Rcvd
2.2.2.9:0        Operational    DU   Passive Off  Off  5/5
[PE1] display mpls ldp lsp
                Status codes: * - stale, L - liberal
                Statistics:
                FECs: 3      Ingress LSPs: 2      Transit LSPs: 2      Egress LSPs: 1

FEC              In/Out Label    Nexthop          OutInterface
1.1.1.9/32       3/-
                  -/1151(L)
2.2.2.9/32       -/3             172.1.1.2        POS5/0
                  1151/3          172.1.1.2        POS5/0
3.3.3.9/32       -/1150          172.1.1.2        POS5/0
                  1150/1150       172.1.1.2        POS5/0
```

3. Configure VPN instances on PEs to allow CE access:

Configure PE 1.

```
[PE1] ip vpn-instance vpn1
```

```

[PE1-vpn-instance-vpn1] route-distinguisher 100:1
[PE1-vpn-instance-vpn1] vpn-target 111:1
[PE1-vpn-instance-vpn1] quit
[PE1] ip vpn-instance vpn2
[PE1-vpn-instance-vpn2] route-distinguisher 100:2
[PE1-vpn-instance-vpn2] vpn-target 222:2
[PE1-vpn-instance-vpn2] quit
[PE1] interface ethernet 1/1
[PE1-Ethernet1/1] ip binding vpn-instance vpn1
[PE1-Ethernet1/1] ip address 10.1.1.2 24
[PE1-Ethernet1/1] quit
[PE1] interface ethernet 1/2
[PE1-Ethernet1/2] ip binding vpn-instance vpn2
[PE1-Ethernet1/2] ip address 10.2.1.2 24
[PE1-Ethernet1/2] quit

```

Configure PE 2.

```

[PE2] ip vpn-instance vpn1
[PE2-vpn-instance-vpn1] route-distinguisher 200:1
[PE2-vpn-instance-vpn1] vpn-target 111:1
[PE2-vpn-instance-vpn1] quit
[PE2] ip vpn-instance vpn2
[PE2-vpn-instance-vpn2] route-distinguisher 200:2
[PE2-vpn-instance-vpn2] vpn-target 222:2
[PE2-vpn-instance-vpn2] quit
[PE2] interface ethernet 1/1
[PE2-Ethernet1/1] ip binding vpn-instance vpn1
[PE2-Ethernet1/1] ip address 10.3.1.2 24
[PE2-Ethernet1/1] quit
[PE2] interface ethernet 1/2
[PE2-Ethernet1/2] ip binding vpn-instance vpn2
[PE2-Ethernet1/2] ip address 10.4.1.2 24
[PE2-Ethernet1/2] quit

```

Configure IP addresses for the CEs according to [Figure 49](#). (Details not shown.)

After completing the configurations, execute the **display ip vpn-instance** command on the PEs to display the configuration of the VPN instance. Use the **ping** command to test connectivity between the PEs and their attached CEs. The PEs can ping their attached CEs. Take PE 1 and CE 1 as an example:

```

[PE1] display ip vpn-instance
  Total VPN-Instances configured : 2
  VPN-Instance Name           RD           Create time
  vpn1                         100:1       2012/02/13 12:49:08
  vpn2                         100:2       2012/02/13 12:49:20
[PE1] ping -vpn-instance vpn1 10.1.1.1
Ping 10.1.1.1 (10.1.1.1): 56 data bytes, press escape sequence to break
56 bytes from 10.1.1.1: icmp_seq=0 ttl=255 time=1.000 ms
56 bytes from 10.1.1.1: icmp_seq=1 ttl=255 time=2.000 ms
56 bytes from 10.1.1.1: icmp_seq=2 ttl=255 time=0.000 ms
56 bytes from 10.1.1.1: icmp_seq=3 ttl=255 time=1.000 ms

```

```
56 bytes from 10.1.1.1: icmp_seq=4 ttl=255 time=0.000 ms
```

```
--- Ping statistics for 10.1.1.1 ---
```

```
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
```

```
round-trip min/avg/max/stddev = 0.000/0.800/2.000/0.748 ms
```

4. Establish EBGP peer relationships between PEs and CEs, and redistribute VPN routes into BGP:

Configure CE 1.

```
<CE1> system-view
[CE1] bgp 65410
[CE1-bgp] peer 10.1.1.2 as-number 100
[CE1-bgp] address-family ipv4 unicast
[CE1-bgp-ipv4] peer 10.1.1.2 enable
[CE1-bgp-ipv4] import-route direct
[CE1-bgp-ipv4] quit
[CE1-bgp] quit
```

Configure the other three CEs in the same way that CE 1 is configured. (Details not shown.)

Configure PE 1.

```
[PE1] bgp 100
[PE1-bgp] ip vpn-instance vpn1
[PE1-bgp-vpn1] peer 10.1.1.1 as-number 65410
[PE1-bgp-vpn1] address-family ipv4 unicast
[PE1-bgp-ipv4-vpn1] peer 10.1.1.1 enable
[PE1-bgp-ipv4-vpn1] import-route direct
[PE1-bgp-ipv4-vpn1] quit
[PE1-bgp-vpn1] quit
[PE1-bgp] ip vpn-instance vpn2
[PE1-bgp-vpn2] peer 10.2.1.1 as-number 65420
[PE1-bgp-vpn2] address-family ipv4 unicast
[PE1-bgp-ipv4-vpn2] peer 10.2.1.1 enable
[PE1-bgp-ipv4-vpn2] import-route direct
[PE1-bgp-ipv4-vpn2] quit
[PE1-bgp-vpn1] quit
[PE1-bgp] quit
```

Configure PE 2 in the same way that PE 1 is configured. (Details not shown.)

After completing the configuration, execute the **display bgp peer ipv4 vpn-instance** command on the PEs. The output shows that BGP peer relationship has been established between the PEs and CEs and has reached the Established state. Take PE 1 as an example:

```
[PE1] display bgp peer ipv4 vpn-instance vpn1
```

```
BGP local router ID: 1.1.1.9
```

```
Local AS number: 100
```

```
Total number of peers: 1
```

```
Peers in established state: 1
```

Peer	AS	MsgRcvd	MsgSent	OutQ	PrefRcv	Up/Down	State
10.1.1.1	65410	4	4	0	2	00:00:22	Established

5. Create an MP-IBGP peer relationship between PEs:

Configure PE 1.

```
[PE1] bgp 100
[PE1-bgp] peer 3.3.3.9 as-number 100
[PE1-bgp] peer 3.3.3.9 connect-interface loopback 0
[PE1-bgp] address-family vpnv4
[PE1-bgp-vpnv4] peer 3.3.3.9 enable
[PE1-bgp-vpnv4] quit
[PE1-bgp] quit
```

Configure PE 2.

```
[PE2] bgp 100
[PE2-bgp] peer 1.1.1.9 as-number 100
[PE2-bgp] peer 1.1.1.9 connect-interface loopback 0
[PE2-bgp] address-family vpnv4
[PE2-bgp-vpnv4] peer 1.1.1.9 enable
[PE2-bgp-vpnv4] quit
[PE2-bgp] quit
```

After completing the configuration, execute the **display bgp peer vpnv4** command. The output shows that a BGP peer relationship has been established between the PEs and has reached the Established state.

```
[PE1] display bgp peer vpnv4
```

```
BGP local router ID: 1.1.1.9
Local AS number: 100
Total number of peers: 1                Peers in established state: 1

Peer                AS  MsgRcvd  MsgSent  OutQ  PrefRcv  Up/Down  State
-----
3.3.3.9            100      3         6       0       0 00:00:32 Established
```

Verifying the configuration

Execute the **display ip routing-table vpn-instance** command on the PEs. The output shows the routes to the CEs. Take PE 1 as an example:

```
[PE1] display ip routing-table vpn-instance vpn1
```

```
Destinations : 13          Routes : 13

Destination/Mask    Proto  Pre  Cost           NextHop           Interface
-----
0.0.0.0/32          Direct  0    0             127.0.0.1         InLoop0
10.1.1.0/24          Direct  0    0             10.1.1.2          Eth1/1
10.1.1.0/32          Direct  0    0             10.1.1.2          Eth1/1
10.1.1.2/32          Direct  0    0             127.0.0.1         InLoop0
10.1.1.255/32        Direct  0    0             10.1.1.2          Eth1/1
10.3.1.0/24          BGP    255  0             3.3.3.9           POS5/0
127.0.0.0/8          Direct  0    0             127.0.0.1         InLoop0
127.0.0.0/32          Direct  0    0             127.0.0.1         InLoop0
127.0.0.1/32          Direct  0    0             127.0.0.1         InLoop0
127.255.255.255/32   Direct  0    0             127.0.0.1         InLoop0
224.0.0.0/4          Direct  0    0             0.0.0.0           NULL0
```

```

224.0.0.0/24      Direct 0    0          0.0.0.0      NULL0
255.255.255.255/32 Direct 0    0          127.0.0.1    InLoop0

```

CEs of the same VPN can ping each other, whereas those of different VPNs cannot. For example, CE 1 can ping CE 3 (10.3.1.1), but it cannot ping CE 4 (10.4.1.1).

Configuring MPLS L3VPN over a GRE tunnel

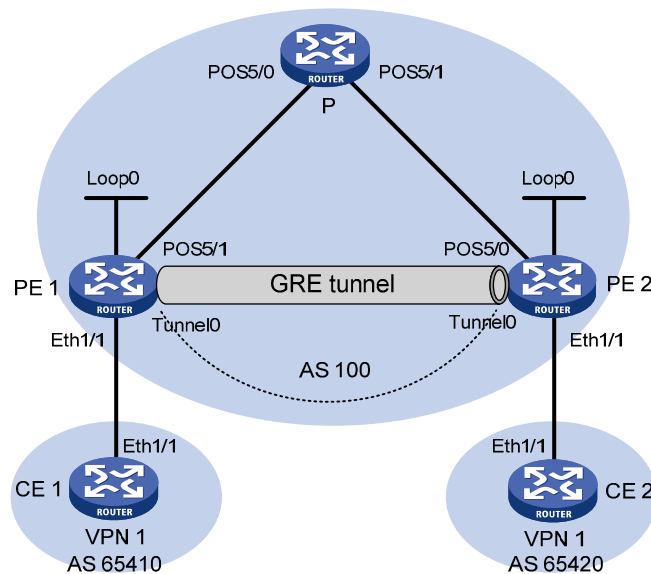
Network requirements

CE 1 and CE 2 belong to VPN 1. The PEs support MPLS. The P router does not support MPLS and provides only IP functions.

On the backbone, use a GRE tunnel to encapsulate and forward VPN packets to implement MPLS L3VPN.

Configure tunnel policies on the PEs, and specify the tunnel type for VPN traffic as GRE.

Figure 50 Network diagram



Device	Interface	IP address	Device	Interface	IP address
CE 1	Eth1/1	10.1.1.1/24	P	POS5/0	172.1.1.2/24
PE 1	Loop0	1.1.1.9/32	PE 2	Loop0	2.2.2.9/32
	Eth1/1	10.1.1.2/24		Eth1/1	10.2.1.2/24
	POS5/1	172.1.1.1/24		POS5/0	172.2.1.2/24
	Tunnel0	20.1.1.1/24		Tunnel0	20.1.1.2/24
CE 2	Eth1/1	10.2.1.1/24			

Configuration procedure

1. Configure an IGP on the MPLS backbone to ensure IP connectivity within the backbone:

This example uses OSPF. (Details not shown.)

After the configurations, OSPF adjacencies are established between PE 1, P, and PE 2. Execute the **display ospf peer** command. The output shows that the adjacency status is Full. Execute the **display ip routing-table** command. The output shows that the PEs have learned the loopback route of each other.

2. Configure basic MPLS on the PEs:

Configure PE 1.

```
<PE1> system-view
[PE1] mpls lsr-id 1.1.1.9
```

Configure PE 2.

```
<PE2> system-view
[PE2] mpls lsr-id 2.2.2.9
```

3. Configure VPN instances on PEs to allow CE access, and apply tunnel policies to the VPN instances, using a GRE tunnel for VPN packet forwarding:

Configure PE 1.

```
[PE1] tunnel-policy gre1
[PE1-tunnel-policy-gre1] select-seq gre load-balance-number 1
[PE1-tunnel-policy-gre1] quit
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 100:1
[PE1-vpn-instance-vpn1] vpn-target 100:1 both
[PE1-vpn-instance-vpn1] tnl-policy gre1
[PE1-vpn-instance-vpn1] quit
[PE1] interface ethernet 1/1
[PE1-Ethernet1/1] ip binding vpn-instance vpn1
[PE1-Ethernet1/1] ip address 10.1.1.2 24
[PE1-Ethernet1/1] quit
```

Configure PE 2.

```
[PE2] tunnel-policy gre1
[PE2-tunnel-policy-gre1] select-seq gre load-balance-number 1
[PE2-tunnel-policy-gre1] quit
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance-vpn1] route-distinguisher 100:2
[PE2-vpn-instance-vpn1] vpn-target 100:1 both
[PE2-vpn-instance-vpn1] tnl-policy gre1
[PE2-vpn-instance-vpn1] quit
[PE2] interface ethernet 1/1
[PE2-Ethernet1/1] ip binding vpn-instance vpn1
[PE2-Ethernet1/1] ip address 10.2.1.2 24
[PE2-Ethernet1/1] quit
```

Configure CE 1.

```
<CE1> system-view
[CE1] interface ethernet 1/1
[CE1-Ethernet1/1] ip address 10.1.1.1 24
[CE1-Ethernet1/1] quit
```

Configure CE 2.

```
<CE2> system-view
[CE2] interface ethernet 1/1
[CE2-Ethernet1/1] ip address 10.2.1.1 24
[CE2-Ethernet1/1] quit
```

After completing the configurations, execute the **display ip vpn-instance** command on the PEs to display the configuration of the VPN instance. Use the **ping** command to test connectivity between the PEs and their attached CEs. The PEs can ping their attached CEs. Take PE 1 as an example:

```
[PE1] display ip vpn-instance
  Total VPN-Instances configured : 1
  VPN-Instance Name              RD              Create time
  vpn1                           100:1          2012/02/13 15:59:50

[PE1] ping -vpn-instance vpn1 10.1.1.1
Ping 10.1.1.1 (10.1.1.1): 56 data bytes, press escape sequence to break
56 bytes from 10.1.1.1: icmp_seq=0 ttl=255 time=1.000 ms
56 bytes from 10.1.1.1: icmp_seq=1 ttl=255 time=0.000 ms
56 bytes from 10.1.1.1: icmp_seq=2 ttl=255 time=0.000 ms
56 bytes from 10.1.1.1: icmp_seq=3 ttl=255 time=0.000 ms
56 bytes from 10.1.1.1: icmp_seq=4 ttl=255 time=0.000 ms

--- Ping statistics for 10.1.1.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.000/0.200/1.000/0.400 ms
```

4. Establish EBGP peer relationships between PEs and CEs, and redistribute VPN routes into BGP:

Configure CE 1.

```
[CE1] bgp 65410
[CE1-bgp] peer 10.1.1.2 as-number 100
[CE1-bgp] address-family ipv4 unicast
[CE1-bgp-ipv4] peer 10.1.1.2 enable
[CE1-bgp-ipv4] import-route direct
[CE1-bgp-ipv4] quit
[CE1-bgp] quit
```

Configure PE 1.

```
[PE1] bgp 100
[PE1-bgp] ip vpn-instance vpn1
[PE1-bgp-vpn1] peer 10.1.1.1 as-number 65410
[PE1-bgp-vpn1] address-family ipv4 unicast
[PE1-bgp-ipv4-vpn1] peer 10.1.1.1 enable
[PE1-bgp-ipv4-vpn1] peer 10.1.1.1 next-hop-local
[PE1-bgp-ipv4-vpn1] import-route direct
[PE1-bgp-ipv4-vpn1] quit
[PE1-bgp-vpn1] quit
[PE1-bgp] quit
```

Configure CE 2 and PE 2 in the same way that CE 1 and PE 1 are configured. (Details not shown.)

Execute the **display bgp peer ipv4 vpn-instance** command on the PEs. The output shows that a BGP peer relationship has been established between a PE and a CE and has reached the Established state.

Take PE 1 as an example:

```
[PE1] display bgp peer ipv4 vpn-instance vpn1
```

```
BGP local router ID: 1.1.1.9
```

```

Local AS number: 100
Total number of peers: 1                Peers in established state: 1

Peer                AS  MsgRcvd  MsgSent  OutQ  PrefRcv  Up/Down  State
10.1.1.1            65410    4        4       0      2 00:00:13 Established

```

5. Configure an MP-IBGP peer relationship between PEs:

Configure PE 1.

```

[PE1] bgp 100
[PE1-bgp] peer 2.2.2.9 as-number 100
[PE1-bgp] peer 2.2.2.9 connect-interface loopback 0
[PE1-bgp] address-family vpnv4
[PE1-bgp-vpnv4] peer 2.2.2.9 enable
[PE1-bgp-vpnv4] quit
[PE1-bgp] quit

```

Configure PE 2 in the same way that PE 1 is configured. (Details not shown.)

Execute the **display bgp peer vpnv4** command on the PEs. The output shows that a BGP peer relationship has been established between the PEs and has reached the Established state.

```

[PE1] display bgp peer vpnv4

BGP local router ID: 1.1.1.9
Local AS number: 100
Total number of peers: 1                Peers in established state: 1

Peer                AS  MsgRcvd  MsgSent  OutQ  PrefRcv  Up/Down  State
2.2.2.9            100     5        7       0      2 00:00:43 Established

```

6. Configure a GRE tunnel:

Configure PE 1.

```

[PE1] interface tunnel 0 mode gre
[PE1-Tunnel0] source loopback 0
[PE1-Tunnel0] destination 2.2.2.9
[PE1-Tunnel0] ip address 20.1.1.1 24
[PE1-Tunnel0] mpls enable
[PE1-Tunnel0] quit

```

Configure PE 2.

```

[PE2] interface tunnel 0 mode gre
[PE2-Tunnel0] source loopback 0
[PE2-Tunnel0] destination 1.1.1.9
[PE2-Tunnel0] ip address 20.1.1.2 24
[PE2-Tunnel0] mpls enable
[PE2-Tunnel0] quit

```

Verifying the configuration

After the configurations, the CEs can learn the interface routes from each other. Take CE 1 as an example:

```

[CE1] display ip routing-table

```

Destinations : 13 Routes : 13

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
0.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
10.1.1.0/24	Direct	0	0	10.1.1.1	Eth1/1
10.1.1.0/32	Direct	0	0	10.1.1.1	Eth1/1
10.1.1.1/32	Direct	0	0	127.0.0.1	InLoop0
10.1.1.255/32	Direct	0	0	10.1.1.1	Eth1/1
10.2.1.0/24	BGP	255	0	10.1.1.2	Eth1/1
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0
127.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0
224.0.0.0/4	Direct	0	0	0.0.0.0	NULL0
224.0.0.0/24	Direct	0	0	0.0.0.0	NULL0
255.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0

CE 1 and CE 2 can ping each other.

Configuring MPLS L3VPN inter-AS option A

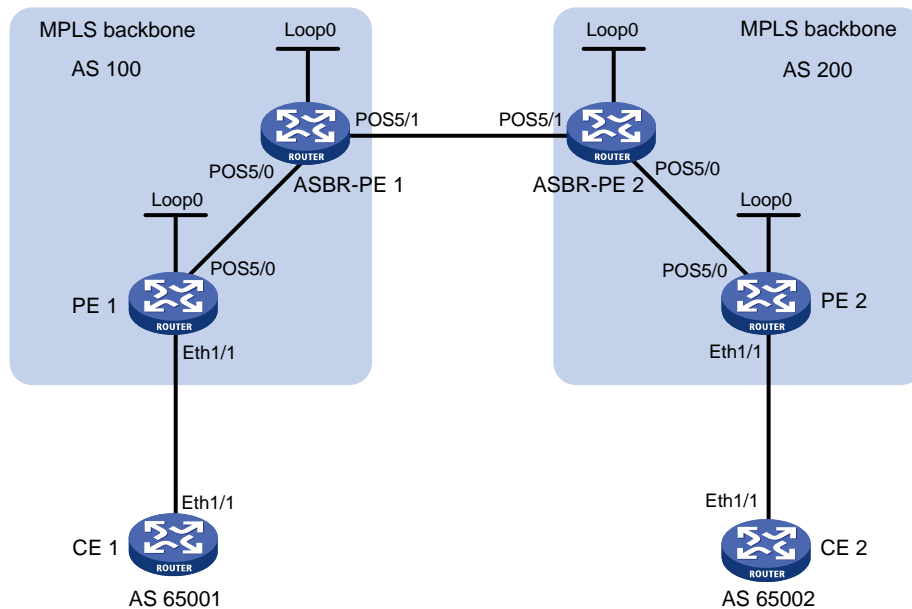
Network requirements

CE 1 and CE 2 belong to the same VPN. CE 1 accesses the network through PE 1 in AS 100, and CE 2 accesses the network through PE 2 in AS 200.

Configure inter-AS option A MPLS L3VPN, and use the VRF-to-VRF method to manage VPN routes.

Run OSPF on the MPLS backbone of each AS.

Figure 51 Network diagram



Device	Interface	IP address	Device	Interface	IP address
CE 1	Eth1/1	10.1.1.1/24	CE 2	Eth1/1	10.2.1.1/24

PE 1	Loop0	1.1.1.9/32	PE 2	Loop0	4.4.4.9/32
	Eth1/1	10.1.1.2/24		Eth1/1	10.2.1.2/24
	POS5/0	172.1.1.2/24		POS5/0	162.1.1.2/24
ASBR-PE1	Loop0	2.2.2.9/32	ASBR-PE2	Loop0	3.3.3.9/32
	POS5/0	172.1.1.1/24		POS5/0	162.1.1.1/24
	POS5/1	192.1.1.1/24		POS5/1	192.1.1.2/24

Configuration procedure

1. Configure IGP on the MPLS backbone:

This example uses OSPF. (Details not shown.)

After the configurations, each ASBR PE and the PE in the same AS can establish an OSPF adjacency. Execute the **display ospf peer** command. The output shows that the adjacency has reached the Full state, and that PEs can learn the routes to the loopback interfaces of each other. Each ASBR PE and the PE in the same AS can ping each other.

2. Configure basic MPLS and MPLS LDP on the MPLS backbone to establish LDP LSPs:

Configure basic MPLS on PE 1, and enable MPLS LDP on the interface connected to ASBR PE 1.

```
<PE1> system-view
[PE1] mpls lsr-id 1.1.1.9
[PE1] mpls ldp
[PE1-ldp] quit
[PE1] interface pos 5/0
[PE1-POS5/0] mpls enable
[PE1-POS5/0] mpls ldp enable
[PE1-POS5/0] quit
```

Configure basic MPLS on ASBR PE 1, and enable MPLS LDP on the interface connected to PE 1.

```
<ASBR-PE1> system-view
[ASBR-PE1] mpls lsr-id 2.2.2.9
[ASBR-PE1] mpls ldp
[ASBR-PE1-ldp] quit
[ASBR-PE1] interface pos 5/0
[ASBR-PE1-POS5/0] mpls enable
[ASBR-PE1-POS5/0] mpls ldp enable
[ASBR-PE1-POS5/0] quit
```

Configure basic MPLS on ASBR PE 2, and enable MPLS LDP on the interface connected to PE 2.

```
<ASBR-PE2> system-view
[ASBR-PE2] mpls lsr-id 3.3.3.9
[ASBR-PE2] mpls ldp
[ASBR-PE2-ldp] quit
[ASBR-PE2] interface pos 5/0
[ASBR-PE2-POS5/0] mpls enable
[ASBR-PE2-POS5/0] mpls ldp enable
[ASBR-PE2-POS5/0] quit
```

Configure basic MPLS on PE 2, and enable MPLS LDP on the interface connected to ASBR PE 2.

```
<PE2> system-view
[PE2] mpls lsr-id 4.4.4.9
[PE2] mpls ldp
```

```
[PE2-ldp] quit
[PE2] interface pos 5/0
[PE2-POS5/0] mpls enable
[PE2-POS5/0] mpls ldp enable
[PE2-POS5/0] quit
```

After the configurations, each PE and the ASBR PE in the same AS can establish an LDP neighbor relationship. Execute the **display mpls ldp peer** command on the devices. The output shows that the LDP session status is Operational.

3. Configure VPN instances on PEs:

For the same VPN, the route targets for the VPN instance on the PE must match those for the VPN instance on the ASBR-PE in the same AS. This is not required for PEs in different ASs.

Configure CE 1.

```
<CE1> system-view
[CE1] interface ethernet 1/1
[CE1-Ethernet1/1] ip address 10.1.1.1 24
[CE1-Ethernet1/1] quit
```

Configure PE 1.

```
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 100:1
[PE1-vpn-instance-vpn1] vpn-target 100:1 both
[PE1-vpn-instance-vpn1] quit
[PE1] interface ethernet 1/1
[PE1-Ethernet1/1] ip binding vpn-instance vpn1
[PE1-Ethernet1/1] ip address 10.1.1.2 24
[PE1-Ethernet1/1] quit
```

Configure CE 2.

```
<CE2> system-view
[CE2] interface ethernet 1/1
[CE2-Ethernet1/1] ip address 10.2.1.1 24
[CE2-Ethernet1/1] quit
```

Configure PE 2.

```
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance-vpn1] route-distinguisher 200:2
[PE2-vpn-instance-vpn1] vpn-target 100:1 both
[PE2-vpn-instance-vpn1] quit
[PE2] interface ethernet 1/1
[PE2-Ethernet1/1] ip binding vpn-instance vpn1
[PE2-Ethernet1/1] ip address 10.2.1.2 24
[PE2-Ethernet1/1] quit
```

Configure ASBR PE 1, creating a VPN instance and binding the instance to the interface connected with ASBR PE 2. (ASBR PE 1 considers ASBR PE 2 its CE.)

```
[ASBR-PE1] ip vpn-instance vpn1
[ASBR-PE1-vpn-vpn1] route-distinguisher 100:1
[ASBR-PE1-vpn-vpn1] vpn-target 100:1 both
[ASBR-PE1-vpn-vpn1] quit
[ASBR-PE1] interface pos 5/1
[ASBR-PE1-POS5/1] ip binding vpn-instance vpn1
```

```
[ASBR-PE1-POS5/1] ip address 192.1.1.1 24
[ASBR-PE1-POS5/1] quit
```

Configure ASBR PE 2, creating a VPN instance and binding the instance to the interface connected with ASBR PE 1. (ASBR PE 2 considers ASBR PE 1 its CE.)

```
[ASBR-PE2] ip vpn-instance vpn1
[ASBR-PE2-vpn-vpn1] route-distinguisher 200:1
[ASBR-PE2-vpn-vpn1] vpn-target 100:1 both
[ASBR-PE2-vpn-vpn1] quit
[ASBR-PE2] interface pos 5/1
[ASBR-PE2-POS5/1] ip binding vpn-instance vpn1
[ASBR-PE2-POS5/1] ip address 192.1.1.2 24
[ASBR-PE2-POS5/1] quit
```

After completing the configurations, display the VPN instance configurations by executing the **display ip vpn-instance** command. The PEs can ping their attached CEs, and the ASBR PEs can ping each other.

4. Establish EBGP peer relationships between PEs and CEs, and redistribute VPN routes into BGP:

Configure CE 1.

```
[CE1] bgp 65001
[CE1-bgp] peer 10.1.1.2 as-number 100
[CE1-bgp] address-family ipv4 unicast
[CE1-bgp-ipv4] peer 10.1.1.2 enable
[CE1-bgp-ipv4] import-route direct
[CE1-bgp-ipv4] quit
[CE1-bgp] quit
```

Configure PE 1.

```
[PE1] bgp 100
[PE1-bgp] ip vpn-instance vpn1
[PE1-bgp-vpn1] peer 10.1.1.1 as-number 65001
[PE1-bgp-vpn1] address-family ipv4 unicast
[PE1-bgp-ipv4-vpn1] peer 10.1.1.1 enable
[PE1-bgp-ipv4-vpn1] import-route direct
[PE1-bgp-ipv4-vpn1] quit
[PE1-bgp-vpn1] quit
[PE1-bgp] quit
```

Configure CE 2.

```
[CE2] bgp 65002
[CE2-bgp] peer 10.2.1.2 as-number 200
[CE2-bgp] address-family ipv4 unicast
[CE2-bgp-ipv4] peer 10.2.1.2 enable
[CE2-bgp-ipv4] import-route direct
[CE2-bgp-ipv4] quit
[CE2-bgp] quit
```

Configure PE 2.

```
[PE2] bgp 200
[PE2-bgp] ip vpn-instance vpn1
[PE2-bgp-vpn1] peer 10.2.1.1 as-number 65002
[PE2-bgp-vpn1] address-family ipv4 unicast
```

```

[PE2-bgp-ipv4-vpn1] peer 10.2.1.1 enable
[PE2-bgp-ipv4-vpn1] import-route direct
[PE2-bgp-ipv4-vpn1] quit
[PE2-bgp-vpn1] quit
[PE2-bgp] quit

```

5. Establish an MP-IBGP peer relationship between each PE and the ASBR-PE in the same AS, and an EBGP peer relationship between the ASBR PEs:

Configure PE 1.

```

[PE1] bgp 100
[PE1-bgp] peer 2.2.2.9 as-number 100
[PE1-bgp] peer 2.2.2.9 connect-interface loopback 0
[PE1-bgp] address-family vpnv4
[PE1-bgp-vpnv4] peer 2.2.2.9 enable
[PE1-bgp-vpnv4] peer 2.2.2.9 next-hop-local
[PE1-bgp-vpnv4] quit
[PE1-bgp] quit

```

Configure ASBR-PE 1.

```

[ASBR-PE1] bgp 100
[ASBR-PE1-bgp] ip vpn-instance vpn1
[ASBR-PE1-bgp-vpn1] peer 192.1.1.2 as-number 200
[ASBR-PE1-bgp-vpn1] address-family ipv4 unicast
[ASBR-PE1-bgp-ipv4-vpn1] peer 192.1.1.2 enable
[ASBR-PE1-bgp-ipv4-vpn1] quit
[ASBR-PE1-bgp-vpn1] quit
[ASBR-PE1-bgp] peer 1.1.1.9 as-number 100
[ASBR-PE1-bgp] peer 1.1.1.9 connect-interface loopback 0
[ASBR-PE1-bgp] address-family vpnv4
[ASBR-PE1-bgp-vpnv4] peer 1.1.1.9 enable
[ASBR-PE1-bgp-vpnv4] peer 1.1.1.9 next-hop-local
[ASBR-PE1-bgp-vpnv4] quit
[ASBR-PE1-bgp] quit

```

Configure ASBR-PE 2.

```

[ASBR-PE2] bgp 200
[ASBR-PE2-bgp] ip vpn-instance vpn1
[ASBR-PE2-bgp-vpn1] peer 192.1.1.1 as-number 100
[ASBR-PE2-bgp-vpn1] address-family ipv4 unicast
[ASBR-PE2-bgp-ipv4-vpn1] peer 192.1.1.1 enable
[ASBR-PE2-bgp-ipv4-vpn1] quit
[ASBR-PE2-bgp-vpn1] quit
[ASBR-PE2-bgp] peer 4.4.4.9 as-number 200
[ASBR-PE2-bgp] peer 4.4.4.9 connect-interface loopback 0
[ASBR-PE2-bgp] address-family vpnv4
[ASBR-PE2-bgp-vpnv4] peer 4.4.4.9 enable
[ASBR-PE2-bgp-vpnv4] peer 4.4.4.9 next-hop-local
[ASBR-PE2-bgp-vpnv4] quit
[ASBR-PE2-bgp] quit

```

Configure PE 2.


```

[PE2] bgp 200
[PE2-bgp] peer 3.3.3.9 as-number 200
[PE2-bgp] peer 3.3.3.9 connect-interface loopback 0
[PE2-bgp] address-family vpnv4
[PE2-bgp-vpnv4] peer 3.3.3.9 enable
[PE2-bgp-vpnv4] peer 3.3.3.9 next-hop-local
[PE2-bgp-vpnv4] quit
[PE2-bgp] quit

```

Verifying the configuration

After the configurations, the CEs can learn the interface routes from each other and ping each other.

Configuring MPLS L3VPN inter-AS option B

Network requirements

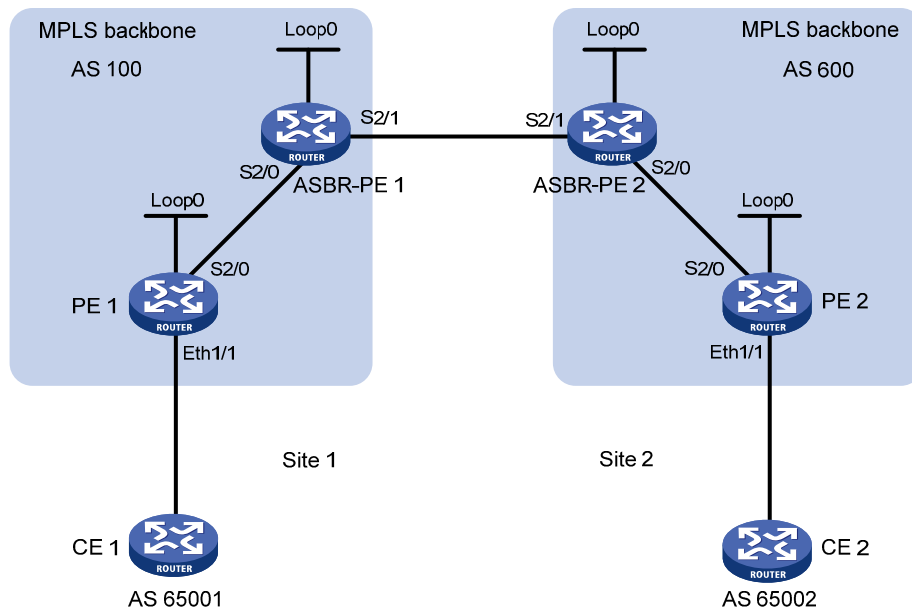
Site 1 and Site 2 belong to the same VPN. CE 1 of Site 1 accesses the network through PE 1 in AS 100, and CE 2 of Site 2 accesses the network through PE 2 in AS 600.

PEs in the same AS run IS-IS.

PE 1 and ASBR-PE 1 exchange VPNv4 routes through MP-IBGP. PE 2 and ASBR-PE 2 exchange VPNv4 routes through MP-IBGP. ASBR-PE 1 and ASBR-PE 2 exchange VPNv4 routes through MP-EBGP.

ASBRs do not perform route target filtering of received VPN-IPv4 routes.

Figure 52 Network diagram



Device	Interface	IP address	Device	Interface	IP address
PE 1	Loop0	2.2.2.9/32	PE 2	Loop0	5.5.5.9/32
	Eth1/1	30.0.0.1/8		Eth1/1	20.0.0.1/8
	S2/0	1.1.1.2/8		S2/0	9.1.1.2/8
ASBR-PE 1	Loop0	3.3.3.9/32	ASBR-PE 2	Loop0	4.4.4.9/32
	S2/0	1.1.1.1/8		S2/0	9.1.1.1/8
	S2/1	11.0.0.2/8		S2/1	11.0.0.1/8

Configuration procedure

1. Configure PE 1:

Configure IS-IS on PE 1.

```
<PE1> system-view
[PE1] isis 1
[PE1-isis-1] network-entity 10.111.111.111.111.00
[PE1-isis-1] quit
```

Configure LSR ID, and enable MPLS and LDP.

```
[PE1] mpls lsr-id 2.2.2.9
[PE1] mpls ldp
[PE1-ldp] quit
```

Configure interface Serial 2/0, and enable IS-IS, MPLS, and LDP on the interface.

```
[PE1] interface serial 2/0
[PE1-Serial2/0] ip address 1.1.1.2 255.0.0.0
[PE1-Serial2/0] isis enable 1
[PE1-Serial2/0] mpls enable
[PE1-Serial2/0] mpls ldp enable
[PE1-Serial2/0] quit
```

Configure interface Loopback 0, and enable IS-IS on it.

```
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 2.2.2.9 32
[PE1-LoopBack0] isis enable 1
[PE1-LoopBack0] quit
```

Create VPN instance **vpn1**, and configure the RD and route target attributes.

```
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 11:11
[PE1-vpn-instance-vpn1] vpn-target 1:1 2:2 3:3 import-extcommunity
[PE1-vpn-instance-vpn1] vpn-target 3:3 export-extcommunity
[PE1-vpn-instance-vpn1] quit
```

Bind the interface connected to CE 1 to the created VPN instance.

```
[PE1] interface ethernet 1/1
[PE1-Ethernet1/1] ip binding vpn-instance vpn1
[PE1-Ethernet1/1] ip address 30.0.0.1 8
[PE1-Ethernet1/1] quit
```

Enable BGP on PE 1.

```
[PE1] bgp 100
```

Configure IBGP peer 3.3.3.9 as a VPNv4 peer.

```
[PE1-bgp] peer 3.3.3.9 as-number 100
[PE1-bgp] peer 3.3.3.9 connect-interface loopback 0
[PE1-bgp] address-family vpnv4
[PE1-bgp-vpnv4] peer 3.3.3.9 enable
[PE1-bgp-vpnv4] quit
```

Redistribute direct routes to the VPN routing table of vpn1.

```
[PE1-bgp] ip vpn-instance vpn1
[PE1-bgp-vpn1] address-family ipv4 unicast
```

```
[PE1-bgp-ipv4-vpn1] import-route direct
[PE1-bgp-ipv4-vpn1] quit
[PE1-bgp-vpn1] quit
[PE1-bgp] quit
```

2. Configure ASBR-PE 1:

Enable IS-IS on ASBR-PE 1.

```
<ASBR-PE1> system-view
[ASBR-PE1] isis 1
[ASBR-PE1-isis-1] network-entity 10.222.222.222.00
[ASBR-PE1-isis-1] quit
```

Configure LSR ID, and enable MPLS and LDP.

```
[ASBR-PE1] mpls lsr-id 3.3.3.9
[ASBR-PE1] mpls ldp
[ASBR-PE1-ldp] quit
```

Configure interface Serial 2/0, and enable IS-IS, MPLS, and LDP on the interface.

```
[ASBR-PE1] interface serial 2/0
[ASBR-PE1-Serial2/0] ip address 1.1.1.1 255.0.0.0
[ASBR-PE1-Serial2/0] isis enable 1
[ASBR-PE1-Serial2/0] mpls enable
[ASBR-PE1-Serial2/0] mpls ldp enable
[ASBR-PE1-Serial2/0] quit
```

Configure interface Serial 2/1, and enable MPLS.

```
[ASBR-PE1] interface serial 2/1
[ASBR-PE1-Serial2/1] ip address 11.0.0.2 255.0.0.0
[ASBR-PE1-Serial2/1] mpls enable
[ASBR-PE1-Serial2/1] quit
```

Configure interface Loopback 0, and enable IS-IS on it.

```
[ASBR-PE1] interface loopback 0
[ASBR-PE1-LoopBack0] ip address 3.3.3.9 32
[ASBR-PE1-LoopBack0] isis enable 1
[ASBR-PE1-LoopBack0] quit
```

Enable BGP on ASBR-PE 1.

```
[ASBR-PE1] bgp 100
[ASBR-PE1-bgp] peer 2.2.2.9 as-number 100
[ASBR-PE1-bgp] peer 2.2.2.9 connect-interface loopback 0
[ASBR-PE1-bgp] peer 11.0.0.1 as-number 600
[ASBR-PE1-bgp] peer 11.0.0.1 connect-interface serial 2/1
```

Disable route target based filtering of received VPNv4 routes.

```
[ASBR-PE1-bgp] address-family vpnv4
[ASBR-PE1-bgp-vpnv4] undo policy vpn-target
```

Configure both IBGP peer 2.2.2.9 and EBGP peer 11.0.0.1 as VPNv4 peers.

```
[ASBR-PE1-bgp-vpnv4] peer 11.0.0.1 enable
[ASBR-PE1-bgp-vpnv4] peer 2.2.2.9 enable
[ASBR-PE1-bgp-vpnv4] quit
```

3. Configure ASBR-PE 2:

Enable IS-IS on ASBR-PE 2.

```

<ASBR-PE2> system-view
[ASBR-PE2] isis 1
[ASBR-PE2-isis-1] network-entity 10.222.222.222.222.00
[ASBR-PE2-isis-1] quit
# Configure LSR ID, and enable MPLS and LDP.
[ASBR-PE2] mpls lsr-id 4.4.4.9
[ASBR-PE2] mpls ldp
[ASBR-PE2-ldp] quit
# Configure interface Serial 2/0, and enable IS-IS, MPLS, and LDP on the interface.
[ASBR-PE2] interface serial 2/0
[ASBR-PE2-Serial2/0] ip address 9.1.1.1 255.0.0.0
[ASBR-PE2-Serial2/0] isis enable 1
[ASBR-PE2-Serial2/0] mpls enable
[ASBR-PE2-Serial2/0] mpls ldp enable
[ASBR-PE2-Serial2/0] quit
# Configure interface Serial 2/1, and enable MPLS.
[ASBR-PE2] interface serial 2/1
[ASBR-PE2-Serial2/1] ip address 11.0.0.1 255.0.0.0
[ASBR-PE2-Serial2/1] mpls enable
[ASBR-PE2-Serial2/1] quit
# Configure interface Loopback 0, and enable IS-IS on it.
[ASBR-PE2] interface loopback 0
[ASBR-PE2-LoopBack0] ip address 4.4.4.9 32
[ASBR-PE2-LoopBack0] isis enable 1
[ASBR-PE2-LoopBack0] quit
# Enable BGP on ASBR-PE 2.
[ASBR-PE2] bgp 600
[ASBR-PE2-bgp] peer 11.0.0.2 as-number 100
[ASBR-PE2-bgp] peer 11.0.0.2 connect-interface serial 2/1
[ASBR-PE2-bgp] peer 5.5.5.9 as-number 600
[ASBR-PE2-bgp] peer 5.5.5.9 connect-interface loopback 0
# Disable route target based filtering of received VPNv4 routes.
[ASBR-PE2-bgp] address-family vpnv4
[ASBR-PE2-bgp-vpnv4] undo policy vpn-target
# Configure both IBGP peer 5.5.5.9 and EBGP peer 11.0.0.2 as VPNv4 peers.
[ASBR-PE2-bgp-vpnv4] peer 11.0.0.2 enable
[ASBR-PE2-bgp-vpnv4] peer 5.5.5.9 enable
[ASBR-PE2-bgp-vpnv4] quit
[ASBR-PE2-bgp] quit

```

4. Configure PE 2:

```

# Enable IS-IS on PE 2.
<PE2> system-view
[PE2] isis 1
[PE2-isis-1] network-entity 10.111.111.111.111.00
[PE2-isis-1] quit
# Configure the LSR ID, and enable MPLS and LDP.

```

```

[PE2] mpls lsr-id 5.5.5.9
[PE2] mpls ldp
[PE2-ldp] quit
# Configure interface Serial 2/0, and enable IS-IS, MPLS, and LDP on the interface.
[PE2] interface serial 2/0
[PE2-Serial2/0] ip address 9.1.1.2 255.0.0.0
[PE2-Serial2/0] isis enable 1
[PE2-Serial2/0] mpls enable
[PE2-Serial2/0] mpls ldp enable
[PE2-Serial2/0] quit
# Configure interface Loopback 0, and enable IS-IS on it.
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 5.5.5.9 32
[PE2-LoopBack0] isis enable 1
[PE2-LoopBack0] quit
# Create VPN instance vpn1, and configure the RD and route target attributes.
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance-vpn1] route-distinguisher 12:12
[PE2-vpn-instance-vpn1] vpn-target 1:1 2:2 3:3 import-extcommunity
[PE2-vpn-instance-vpn1] vpn-target 3:3 export-extcommunity
[PE2-vpn-instance-vpn1] quit
# Bind the interface connected with CE 1 to the created VPN instance.
[PE2] interface ethernet 1/1
[PE2-Ethernet1/1] ip binding vpn-instance vpn1
[PE2-Ethernet1/1] ip address 20.0.0.1 8
[PE2-Ethernet1/1] quit
# Enable BGP on PE 2.
[PE2] bgp 600
# Configure IBGP peer 4.4.4.9 as a VPNv4 peer.
[PE2-bgp] peer 4.4.4.9 as-number 600
[PE2-bgp] peer 4.4.4.9 connect-interface loopback 0
[PE2-bgp] address-family vpnv4
[PE2-bgp-vpnv4] peer 4.4.4.9 enable
[PE2-bgp-vpnv4] quit
# Redistribute direct routes to the VPN routing table of vpn1.
[PE2-bgp] ip vpn-instance vpn1
[PE2-bgp-vpn1] address-family ipv4 unicast
[PE2-bgp-ipv4-vpn1] import-route direct
[PE2-bgp-ipv4-vpn1] quit
[PE2-bgp-vpn1] quit
[PE2-bgp] quit

```

Verifying the configuration

Ping PE 1 from PE 2, and ping PE 2 from PE 1. They can ping each other successfully. Take PE1 as an example:

```

[PE1] ping -a 30.0.0.1 -vpn-instance vpn1 20.0.0.1
Ping 20.0.0.1 (20.0.0.1) from 30.0.0.1: 56 data bytes, press escape sequence to break

```

```

56 bytes from 20.0.0.1: icmp_seq=0 ttl=255 time=0.000 ms
56 bytes from 20.0.0.1: icmp_seq=1 ttl=255 time=0.000 ms
56 bytes from 20.0.0.1: icmp_seq=2 ttl=255 time=0.000 ms
56 bytes from 20.0.0.1: icmp_seq=3 ttl=255 time=0.000 ms
56 bytes from 20.0.0.1: icmp_seq=4 ttl=255 time=0.000 ms

```

```

--- Ping statistics for 20.0.0.1 ---

```

```

5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.000/0.000/0.000/0.000 ms

```

Configuring MPLS L3VPN inter-AS option C

Network requirements

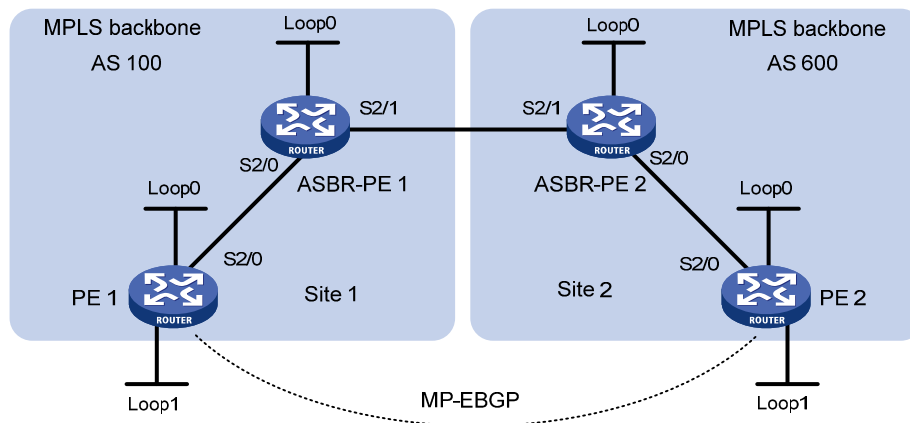
Site 1 and Site 2 belong to the same VPN. Site 1 accesses the network through PE 1 in AS 100, and Site 2 accesses the network through PE 2 in AS 600. PEs in the same AS run IS-IS.

PE 1 and ASBR-PE 1 exchange labeled IPv4 routes through IBGP. PE 2 and ASBR-PE 2 exchange labeled IPv4 routes through IBGP. PE 1 and PE 2 are MP-EBGP peers and exchange VPNv4 routes.

ASBR-PE 1 and ASBR-PE 2 use routing policies and label the routes received from each other.

ASBR-PE 1 and ASBR-PE 2 use EBGP to exchange labeled IPv4 routes.

Figure 53 Network diagram



Device	Interface	IP address	Device	Interface	IP address
PE 1	Loop0	2.2.2.9/32	PE 2	Loop0	5.5.5.9/32
	Loop1	30.0.0.1/32		Loop1	20.0.0.1/32
	S2/0	1.1.1.2/8		S2/0	9.1.1.2/8
ASBR-PE 1	Loop0	3.3.3.9/32	ASBR-PE 2	Loop0	4.4.4.9/32
	S2/0	1.1.1.1/8		S2/0	9.1.1.1/8
	S2/1	11.0.0.2/8		S2/1	11.0.0.1/8

Configuration procedure

1. Configure PE 1:

```
# Configure IS-IS on PE 1.
```

```
<PE1> system-view
```

```
[PE1] isis 1
```

```

[PE1-isis-1] network-entity 10.111.111.111.111.00
[PE1-isis-1] quit
# Configure LSR ID, and enable MPLS and LDP.
[PE1] mpls lsr-id 2.2.2.9
[PE1] mpls ldp
[PE1-ldp] quit
# Configure interface Serial 2/0, and enable IS-IS, MPLS, and LDP on the interface.
[PE1] interface serial 2/0
[PE1-Serial2/0] ip address 1.1.1.2 255.0.0.0
[PE1-Serial2/0] isis enable 1
[PE1-Serial2/0] mpls enable
[PE1-Serial2/0] mpls ldp enable
[PE1-Serial2/0] quit
# Configure interface Loopback 0, and start IS-IS on it.
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 2.2.2.9 32
[PE1-LoopBack0] isis enable 1
[PE1-LoopBack0] quit
# Create VPN instance vpn1, and configure the RD and route target attributes.
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 11:11
[PE1-vpn-instance-vpn1] vpn-target 1:1 2:2 3:3 import-extcommunity
[PE1-vpn-instance-vpn1] vpn-target 3:3 export-extcommunity
[PE1-vpn-instance-vpn1] quit
# Configure interface Loopback 1, and bind the interface to VPN instance vpn1.
[PE1] interface loopback 1
[PE1-LoopBack1] ip binding vpn-instance vpn1
[PE1-LoopBack1] ip address 30.0.0.1 32
[PE1-LoopBack1] quit
# Start BGP on PE 1.
[PE1] bgp 100
# Enable the capability to advertise labeled routes to IBGP peer 3.3.3.9 and to receive labeled routes from the peer.
[PE1-bgp] peer 3.3.3.9 as-number 100
[PE1-bgp] peer 3.3.3.9 connect-interface loopback 0
[PE1-bgp] address-family ipv4 unicast
[PE1-bgp-ipv4] peer 3.3.3.9 enable
[PE1-bgp-ipv4] peer 3.3.3.9 label-route-capability
[PE1-bgp-ipv4] quit
# Configure the maximum hop count from PE 1 to EBGp peer 5.5.5.9 as 10.
[PE1-bgp] peer 5.5.5.9 as-number 600
[PE1-bgp] peer 5.5.5.9 connect-interface loopback 0
[PE1-bgp] peer 5.5.5.9 ebgp-max-hop 10
# Configure peer 5.5.5.9 as a VPNv4 peer.
[PE1-bgp] address-family vpnv4
[PE1-bgp-vpnv4] peer 5.5.5.9 enable

```

```
[PE1-bgp-vpnv4] quit
# Redistribute direct routes to the routing table of vpn1.
[PE1-bgp] ip vpn-instance vpn1
[PE1-bgp-vpn1] address-family ipv4 unicast
[PE1-bgp-ipv4-vpn1] import-route direct
[PE1-bgp-ipv4-vpn1] quit
[PE1-bgp-vpn1] quit
[PE1-bgp] quit
```

2. Configure ASBR-PE 1:

```
# Start IS-IS on ASBR-PE 1.
<ASBR-PE1> system-view
[ASBR-PE1] isis 1
[ASBR-PE1-isis-1] network-entity 10.222.222.222.000
[ASBR-PE1-isis-1] quit

# Configure the LSR ID, and enable MPLS and LDP.
[ASBR-PE1] mpls lsr-id 3.3.3.9
[ASBR-PE1] mpls ldp
[ASBR-PE1-ldp] quit

# Configure interface Serial 2/0, and enable IS-IS, MPLS, and LDP on the interface.
[ASBR-PE1] interface serial 2/0
[ASBR-PE1-Serial2/0] ip address 1.1.1.1 255.0.0.0
[ASBR-PE1-Serial2/0] isis enable 1
[ASBR-PE1-Serial2/0] mpls enable
[ASBR-PE1-Serial2/0] mpls ldp enable
[ASBR-PE1-Serial2/0] quit

# Configure interface Serial 2/1, and enable MPLS on it.
[ASBR-PE1] interface serial 2/1
[ASBR-PE1-Serial2/1] ip address 11.0.0.2 255.0.0.0
[ASBR-PE1-Serial2/1] mpls enable
[ASBR-PE1-Serial2/1] quit

# Configure interface Loopback 0, and start IS-IS on it.
[ASBR-PE1] interface loopback 0
[ASBR-PE1-LoopBack0] ip address 3.3.3.9 32
[ASBR-PE1-LoopBack0] isis enable 1
[ASBR-PE1-LoopBack0] quit

# Create routing policies.
[ASBR-PE1] route-policy policy1 permit node 1
[ASBR-PE1-route-policy-policy1-1] apply mpls-label
[ASBR-PE1-route-policy-policy1-1] quit
[ASBR-PE1] route-policy policy2 permit node 1
[ASBR-PE1-route-policy-policy2-1] if-match mpls-label
[ASBR-PE1-route-policy-policy2-1] apply mpls-label
[ASBR-PE1-route-policy-policy2-1] quit

# Start BGP on ASBR-PE 1, and apply the routing policy policy2 to routes advertised to IBGP peer 2.2.2.9.
[ASBR-PE1] bgp 100
```



```

[ASBR-PE1-bgp] peer 2.2.2.9 as-number 100
[ASBR-PE1-bgp] peer 2.2.2.9 connect-interface loopback 0
[ASBR-PE1-bgp] address-family ipv4 unicast
[ASBR-PE1-bgp-ipv4] peer 2.2.2.9 enable
[ASBR-PE1-bgp-ipv4] peer 2.2.2.9 route-policy policy2 export
# Enable the capability to advertise labeled routes to IBGP peer 2.2.2.9 and to receive labeled
routes from the peer.
[ASBR-PE1-bgp-ipv4] peer 2.2.2.9 label-route-capability
# Redistribute routes from IS-IS process 1 to BGP.
[ASBR-PE1-bgp-ipv4] import-route isis 1
[ASBR-PE1-bgp-ipv4] quit
# Apply the routing policy policy1 to routes advertised to EBGP peer 11.0.0.1.
[ASBR-PE1-bgp] peer 11.0.0.1 as-number 600
[ASBR-PE1-bgp] address-family ipv4 unicast
[ASBR-PE1-bgp-ipv4] peer 11.0.0.1 enable
[ASBR-PE1-bgp-ipv4] peer 11.0.0.1 route-policy policy1 export
# Enable the capability to advertise labeled routes to EBGP peer 11.0.0.1 and to receive labeled
routes from the peer.
[ASBR-PE1-bgp-ipv4] peer 11.0.0.1 label-route-capability
[ASBR-PE1-bgp-ipv4] quit
[ASBR-PE1-bgp] quit

```

3. Configure ASBR-PE 2:

```

# Enable IS-IS on ASBR-PE 2.
<ASBR-PE2> system-view
[ASBR-PE2] isis 1
[ASBR-PE2-isis-1] network-entity 10.222.222.222.222.00
[ASBR-PE2-isis-1] quit
# Configure the LSR ID, and enable MPLS and LDP.
[ASBR-PE2] mpls lsr-id 4.4.4.9
[ASBR-PE2] mpls ldp
[ASBR-PE2-ldp] quit
# Configure interface Serial 2/0, and enable IS-IS, MPLS, and LDP on the interface.
[ASBR-PE2] interface serial 2/0
[ASBR-PE2-Serial2/0] ip address 9.1.1.1 255.0.0.0
[ASBR-PE2-Serial2/0] isis enable 1
[ASBR-PE2-Serial2/0] mpls enable
[ASBR-PE2-Serial2/0] mpls ldp enable
[ASBR-PE2-Serial2/0] quit
# Configure interface Loopback 0, and enable IS-IS on it.
[ASBR-PE2] interface loopback 0
[ASBR-PE2-LoopBack0] ip address 4.4.4.9 32
[ASBR-PE2-LoopBack0] isis enable 1
[ASBR-PE2-LoopBack0] quit
# Configure interface Serial 2/1, and enable MPLS on the interface.
[ASBR-PE2] interface serial 2/1
[ASBR-PE2-Serial2/1] ip address 11.0.0.1 255.0.0.0

```

```

[ASBR-PE2-Serial2/1] mpls enable
[ASBR-PE2-Serial2/1] quit
# Create routing policies.
[ASBR-PE2] route-policy policy1 permit node 1
[ASBR-PE2-route-policy-policy1-1] apply mpls-label
[ASBR-PE2-route-policy-policy1-1] quit
[ASBR-PE2] route-policy policy2 permit node 1
[ASBR-PE2-route-policy-policy2-1] if-match mpls-label
[ASBR-PE2-route-policy-policy2-1] apply mpls-label
[ASBR-PE2-route-policy-policy2-1] quit
# Enable BGP on ASBR-PE 2, and enable the capability to advertise labeled routes to IBGP peer
5.5.5.9 and to receive labeled routes from the peer.
[ASBR-PE2] bgp 600
[ASBR-PE2-bgp] peer 5.5.5.9 as-number 600
[ASBR-PE2-bgp] peer 5.5.5.9 connect-interface loopback 0
[ASBR-PE2-bgp] address-family ipv4 unicast
[ASBR-PE2-bgp-ipv4] peer 5.5.5.9 enable
[ASBR-PE2-bgp-ipv4] peer 5.5.5.9 label-route-capability
# Apply the routing policy policy2 to routes advertised to IBGP peer 5.5.5.9.
[ASBR-PE2-bgp-ipv4] peer 5.5.5.9 route-policy policy2 export
# Redistribute routes from IS-IS process 1.
[ASBR-PE2-bgp-ipv4] import-route isis 1
[ASBR-PE2-bgp-ipv4] quit
# Apply the routing policy policy1 to routes advertised to EBGp peer 11.0.0.2.
[ASBR-PE2-bgp] peer 11.0.0.2 as-number 100
[ASBR-PE2-bgp] address-family ipv4 unicast
[ASBR-PE2-bgp-ipv4] peer 11.0.0.2 enable
[ASBR-PE2-bgp-ipv4] peer 11.0.0.2 route-policy policy1 export
# Enable the capability to advertise labeled routes to EBGp peer 11.0.0.2 and to receive labeled
routes from the peer.
[ASBR-PE2-bgp-ipv4] peer 11.0.0.2 label-route-capability
[ASBR-PE2-bgp-ipv4] quit
[ASBR-PE2-bgp] quit

```

4. Configure PE 2:

```

# Enable IS-IS on PE 2.
<PE2> system-view
[PE2] isis 1
[PE2-isis-1] network-entity 10.111.111.111.00
[PE2-isis-1] quit
# Configure the LSR ID, and enable MPLS and LDP.
[PE2] mpls lsr-id 5.5.5.9
[PE2] mpls ldp
[PE2-ldp] quit
# Configure interface Serial 2/0, and enable IS-IS, MPLS, and LDP on the interface.
[PE2] interface serial 2/0
[PE2-Serial2/0] ip address 9.1.1.2 255.0.0.0

```

```

[PE2-Serial2/0] isis enable 1
[PE2-Serial2/0] mpls enable
[PE2-Serial2/0] mpls ldp enable
[PE2-Serial2/0] quit
# Configure the interface Loopback 0, and enable IS-IS on it.
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 5.5.5.9 32
[PE2-LoopBack0] isis enable 1
[PE2-LoopBack0] quit
# Create VPN instance vpn1, and configure the RD and route target attributes.
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance-vpn1] route-distinguisher 11:11
[PE2-vpn-instance-vpn1] vpn-target 1:1 2:2 3:3 import-extcommunity
[PE2-vpn-instance-vpn1] vpn-target 3:3 export-extcommunity
[PE2-vpn-instance-vpn1] quit
# Configure interface Loopback 1 and bind the interface to VPN instance vpn1.
[PE2] interface loopback 1
[PE2-LoopBack1] ip binding vpn-instance vpn1
[PE2-LoopBack1] ip address 20.0.0.1 32
[PE2-LoopBack1] quit
# Enable BGP on PE 2.
[PE2] bgp 600
# Enable the capability to advertise labeled routes to IBGP peer 4.4.4.9 and to receive labeled
routes from the peer.
[PE2-bgp] peer 4.4.4.9 as-number 600
[PE2-bgp] peer 4.4.4.9 connect-interface loopback 0
[PE2-bgp] address-family ipv4 unicast
[PE2-bgp-ipv4] peer 4.4.4.9 enable
[PE2-bgp-ipv4] peer 4.4.4.9 label-route-capability
[PE2-bgp-ipv4] quit
# Configure the maximum hop count from PE 2 to EBGP peer 2.2.2.9 as 10.
[PE2-bgp] peer 2.2.2.9 as-number 100
[PE2-bgp] peer 2.2.2.9 connect-interface loopback 0
[PE2-bgp] peer 2.2.2.9 ebgp-max-hop 10
# Configure peer 2.2.2.9 as a VPNv4 peer.
[PE2-bgp] address-family vpnv4
[PE2-bgp-vpnv4] peer 2.2.2.9 enable
[PE2-bgp-vpnv4] quit
# Redistribute direct routes to the routing table of vpn1.
[PE2-bgp] ip vpn-instance vpn1
[PE2-bgp-vpn1] address-family ipv4 unicast
[PE2-bgp-ipv4-vpn1] import-route direct
[PE2-bgp-ipv4-vpn1] quit
[PE2-bgp-vpn1] quit
[PE2-bgp] quit

```

Verifying the configuration

After the configurations, PE 1 and PE 2 can ping each other. Ping PE 2 from PE 1:

```
[PE1] ping -a 30.0.0.1 -vpn-instance vpn1 20.0.0.1
Ping 20.0.0.1 (20.0.0.1) from 30.0.0.1: 56 data bytes, press escape sequence to break
56 bytes from 20.0.0.1: icmp_seq=0 ttl=253 time=2.000 ms
56 bytes from 20.0.0.1: icmp_seq=1 ttl=253 time=1.000 ms
56 bytes from 20.0.0.1: icmp_seq=2 ttl=253 time=1.000 ms
56 bytes from 20.0.0.1: icmp_seq=3 ttl=253 time=1.000 ms
56 bytes from 20.0.0.1: icmp_seq=4 ttl=253 time=1.000 ms

--- Ping statistics for 20.0.0.1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/stddev = 1.000/1.200/2.000/0.400 ms
```

Configuring MPLS L3VPN carrier's carrier

Network requirements

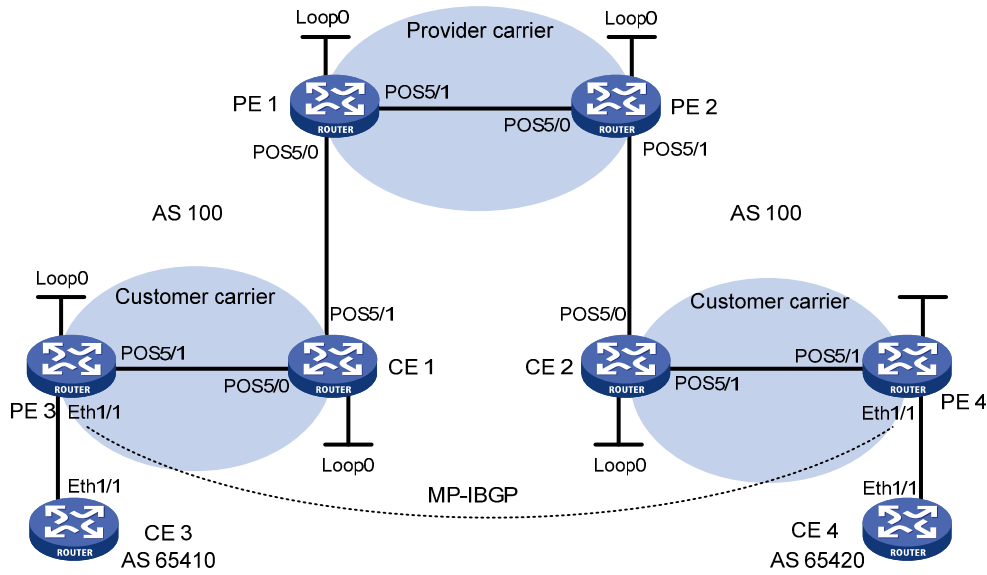
Configure carrier's carrier for the scenario shown in [Figure 54](#). In this scenario:

- PE 1 and PE 2 are the provider carrier's PE routers. They provide VPN services for the customer carrier.
- CE 1 and CE 2 are the customer carrier's routers. They are connected to the provider carrier's backbone as CE routers.
- PE 3 and PE 4 are the customer carrier's PE routers. They provide MPLS L3VPN services for the end customers.
- CE 3 and CE 4 are customers of the customer carrier.

The key to carrier's carrier deployment is to configure exchange of two kinds of routes:

- Exchange of the customer carrier's internal routes on the provider carrier's backbone.
- Exchange of the end customers' VPN routes between PE 3 and PE 4, the PEs of the customer carrier. In this process, an MP-IBGP peer relationship must be established between PE 3 and PE 4.

Figure 54 Network diagram



Device	Interface	IP address	Device	Interface	IP address
CE 3	Eth1/1	100.1.1.1/24	CE 4	Eth1/1	120.1.1.1/24
PE 3	Loop0	1.1.1.9/32	PE 4	Loop0	6.6.6.9/32
	Eth1/1	100.1.1.2/24		Eth1/1	120.1.1.2/24
	POS5/1	10.1.1.1/24		POS5/1	20.1.1.2/24
CE 1	Loop0	2.2.2.9/32	CE 2	Loop0	5.5.5.9/32
	POS5/0	10.1.1.2/24		POS5/0	21.1.1.2/24
	POS5/1	11.1.1.1/24		POS5/1	20.1.1.1/24
PE 1	Loop0	3.3.3.9/32	PE 2	Loop0	4.4.4.9/32
	POS5/0	11.1.1.2/24		POS5/0	30.1.1.2/24
	POS5/1	30.1.1.1/24		POS5/1	21.1.1.1/24

Configuration procedure

1. Configure MPLS L3VPN on the provider carrier backbone. Enable IS-IS as the IGP, enable LDP between PE 1 and PE 2, and establish an MP-IBGP peer relationship between the PEs:

Configure PE 1.

```

<PE1> system-view
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 3.3.3.9 32
[PE1-LoopBack0] quit
[PE1] mpls lsr-id 3.3.3.9
[PE1] mpls ldp
[PE1-ldp] quit
[PE1] isis 1
[PE1-isis-1] network-entity 10.0000.0000.0000.0004.00
[PE1-isis-1] quit
[PE1] interface loopback 0
[PE1-LoopBack0] isis enable 1
[PE1-LoopBack0] quit
[PE1] interface pos 5/1
    
```

```

[PE1-POS5/1] ip address 30.1.1.1 24
[PE1-POS5/1] isis enable 1
[PE1-POS5/1] mpls enable
[PE1-POS5/1] mpls ldp enable
[PE1-POS5/1] mpls ldp transport-address interface
[PE1-POS5/1] quit
[PE1] bgp 100
[PE1-bgp] peer 4.4.4.9 as-number 100
[PE1-bgp] peer 4.4.4.9 connect-interface loopback 0
[PE1-bgp] address-family vpnv4
[PE1-bgp-vpnv4] peer 4.4.4.9 enable
[PE1-bgp-vpnv4] quit
[PE1-bgp] quit

```

Configure PE 2 in the same way that PE 1 is configured. (Details not shown.)

After completing the configurations, execute the **display mpls ldp peer** command on PE 1 or PE 2 to see that the LDP session has been successfully established. Execute the **display bgp peer vpnv4** command to see that a BGP peer relationship has been established and has reached the Established state. Execute the **display isis peer** command to see that the IS-IS neighbor relationship has been set up. Take PE 1 as an example:

```

[PE1] display mpls ldp peer
Total number of peers: 1
Peer LDP ID          State          LAM  Role    GR   MD5   KA Sent/Rcvd
4.4.4.9:0            Operational    DU   Active  Off  Off   8/8
[PE1] display bgp peer vpnv4

BGP local router ID: 3.3.3.9
Local AS number: 100
Total number of peers: 1                Peers in established state: 1

Peer                AS  MsgRcvd  MsgSent  OutQ  PrefRcv  Up/Down  State
4.4.4.9            100      3         6      0      0 00:00:32 Established
[PE1] display isis peer

Peer information for ISIS(1)
-----

System Id: 0000.0000.0005
Interface: POS5/1                Circuit Id: 0000.0000.0005.02
State: Up      HoldTime: 8s      Type: L1(L1L2)      PRI: 64

System Id: 0000.0000.0005
Interface: POS5/1                Circuit Id: 0000.0000.0005.02
State: Up      HoldTime: 8s      Type: L2(L1L2)      PRI: 64

```

2. Configure the customer carrier network—enable IS-IS as the IGP and enable LDP between PE 3 and CE 1, and between PE 4 and CE 2:

Configure PE 3.

```

<PE3> system-view
[PE3] interface loopback 0

```

```

[PE3-LoopBack0] ip address 1.1.1.9 32
[PE3-LoopBack0] quit
[PE3] mpls lsr-id 1.1.1.9
[PE3] mpls ldp
[PE3-ldp] quit
[PE3] isis 2
[PE3-isis-2] network-entity 10.0000.0000.0000.0001.00
[PE3-isis-2] quit
[PE3] interface loopback 0
[PE3-LoopBack0] isis enable 2
[PE3-LoopBack0] quit
[PE3] interface pos 5/1
[PE3-POS5/1] ip address 10.1.1.1 24
[PE3-POS5/1] isis enable 2
[PE3-POS5/1] mpls enable
[PE3-POS5/1] mpls ldp enable
[PE3-POS5/1] mpls ldp transport-address interface
[PE3-POS5/1] quit

```

Configure CE 1.

```

<CE1> system-view
[CE1] interface loopback 0
[CE1-LoopBack0] ip address 2.2.2.9 32
[CE1-LoopBack0] quit
[CE1] mpls lsr-id 2.2.2.9
[CE1] mpls ldp
[CE1-ldp] quit
[CE1] isis 2
[CE1-isis-2] network-entity 10.0000.0000.0000.0002.00
[CE1-isis-2] quit
[CE1] interface loopback 0
[CE1-LoopBack0] isis enable 2
[CE1-LoopBack0] quit
[CE1] interface POS 5/0
[CE1-POS5/0] ip address 10.1.1.2 24
[CE1-POS5/0] isis enable 2
[CE1-POS5/0] mpls enable
[CE1-POS5/0] mpls ldp enable
[CE1-POS5/0] mpls ldp transport-address interface
[CE1-POS5/0] quit

```

After the configurations, PE 3 and CE 1 can establish an LDP session and IS-IS neighbor relationship between them.

Configure PE 4 and CE 2 in the same way that PE 3 and CE 1 are configured. (Details not shown.)

3. Perform configuration to allow CEs of the customer carrier to access PEs of the provider carrier, and redistribute IS-IS routes to BGP and BGP routes to IS-IS on the PEs:

Configure PE 1.

```

[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 200:1

```

```

[PE1-vpn-instance-vpn1] vpn-target 1:1
[PE1-vpn-instance-vpn1] quit
[PE1] mpls ldp
[PE1-ldp] vpn-instance vpn1
[PE1-ldp-vpn-instance-vpn1] quit
[PE1-ldp] quit
[PE1] isis 2 vpn-instance vpn1
[PE1-isis-2] network-entity 10.0000.0000.0000.0003.00
[PE1-isis-2] import-route bgp
[PE1-isis-2] quit
[PE1] interface pos 5/0
[PE1-POS5/0] ip binding vpn-instance vpn1
[PE1-POS5/0] ip address 11.1.1.2 24
[PE1-POS5/0] isis enable 2
[PE1-POS5/0] mpls enable
[PE1-POS5/0] mpls ldp enable
[PE1-POS5/0] mpls ldp transport-address interface
[PE1-POS5/0] quit
[PE1] bgp 100
[PE1-bgp] ip vpn-instance vpn1
[PE1-bgp-vpn1] address-family ipv4 unicast
[PE1-bgp-ipv4-vpn1] import isis 2
[PE1-bgp-ipv4-vpn1] quit
[PE1-bgp-vpn1] quit
[PE1-bgp] quit

```

Configure CE 1.

```

[CE1] interface pos 5/1
[CE1-POS5/1] ip address 11.1.1.1 24
[CE1-POS5/1] isis enable 2
[CE1-POS5/1] mpls enable
[CE1-POS5/1] mpls ldp enable
[CE1-POS5/1] mpls ldp transport-address interface
[CE1-POS5/1] quit

```

After the configurations, PE 1 and CE 1 can establish an LDP session and IS-IS neighbor relationship between them.

Configure PE 2 and CE 2 in the same way that PE 1 and CE 1 are configured. (Details not shown.)

4. Perform configuration to connect CEs of the end customers to the PEs of the customer carrier:

Configure CE 3.

```

<CE3> system-view
[CE3] interface ethernet 1/1
[CE3-Ethernet1/1] ip address 100.1.1.1 24
[CE3-Ethernet1/1] quit
[CE3] bgp 65410
[CE3-bgp] peer 100.1.1.2 as-number 100
[CE3-bgp] address-family ipv4 unicast
[CE3-bgp-ipv4] peer 100.1.1.2 enable

```



```

[CE3-bgp-ipv4] import-route direct
[CE3-bgp-ipv4] quit
[CE3-bgp] quit
# Configure PE 3.
[PE3] ip vpn-instance vpn1
[PE3-vpn-instance-vpn1] route-distinguisher 100:1
[PE3-vpn-instance-vpn1] vpn-target 1:1
[PE3-vpn-instance-vpn1] quit
[PE3] interface ethernet 1/1
[PE3-Ethernet1/1] ip binding vpn-instance vpn1
[PE3-Ethernet1/1] ip address 100.1.1.2 24
[PE3-Ethernet1/1] quit
[PE3] bgp 100
[PE3-bgp] ip vpn-instance vpn1
[PE3-bgp-vpn1] peer 100.1.1.1 as-number 65410
[PE3-bgp-vpn1] address-family ipv4 unicast
[PE3-bgp-ipv4-vpn1] peer 100.1.1.1 enable
[PE3-bgp-ipv4-vpn1] import-route direct
[PE3-bgp-ipv4-vpn1] quit
[PE3-bgp-vpn1] quit
[PE3-bgp] quit

```

Configure PE 4 and CE 4 in the same way that PE 3 and CE 3 are configured. (Details not shown.)

5. Configure an MP-IBGP peer relationship between the PEs of the customer carrier to exchange the VPN routes of the end customers:

```

# Configure PE 3.
[PE3] bgp 100
[PE3-bgp] peer 6.6.6.9 as-number 100
[PE3-bgp] peer 6.6.6.9 connect-interface loopback 0
[PE3-bgp] address-family vpnv4
[PE3-bgp-vpnv4] peer 6.6.6.9 enable
[PE3-bgp-vpnv4] quit
[PE3-bgp] quit

```

Configure PE 4 in the same way that PE 3 is configured. (Details not shown.)

Verifying the configuration

After completing all the configurations, execute the **display ip routing-table** command on PE 1 and PE 2. The output shows that only routes of the provider carrier network are present in the public network routing table of PE 1 and PE 2. Take PE 1 as an example:

```

[PE1] display ip routing-table
Routing Tables: Public
          Destinations : 7           Routes : 7
Destination/Mask    Proto  Pre  Cost    NextHop         Interface
3.3.3.9/32          Direct  0    0       127.0.0.1       InLoop0
4.4.4.9/32          ISIS    15  10       30.1.1.2        POS5/1
30.1.1.0/24         Direct  0    0       30.1.1.1        POS5/1
30.1.1.1/32         Direct  0    0       127.0.0.1       InLoop0
30.1.1.2/32         Direct  0    0       30.1.1.2        POS5/1

```

```

127.0.0.0/8          Direct 0    0          127.0.0.1    InLoop0
127.0.0.1/32        Direct 0    0          127.0.0.1    InLoop0

```

Execute the **display ip routing-table vpn-instance** command on PE 1 and PE 2. The output shows that the internal routes of the customer carrier network are present in the VPN routing tables, but the VPN routes that the customer carrier maintains are not. Take PE 1 as an example:

```
[PE1] display ip routing-table vpn-instance vpn1
```

```
Routing Tables: vpn1
```

```

          Destinations : 11          Routes : 11
Destination/Mask    Proto  Pre  Cost   NextHop    Interface
1.1.1.9/32          ISIS   15   20     11.1.1.1   POS5/0
2.2.2.9/32          ISIS   15   10     11.1.1.1   POS5/0
5.5.5.9/32          BGP    255  0      4.4.4.9    NULL0
6.6.6.9/32          BGP    255  0      4.4.4.9    NULL0
10.1.1.0/24         ISIS   15   20     11.1.1.1   POS5/0
11.1.1.0/24         Direct 0    0      11.1.1.1   POS5/0
11.1.1.1/32         Direct 0    0      127.0.0.1  InLoop0
11.1.1.2/32         Direct 0    0      11.1.1.2   POS5/0
20.1.1.0/24         BGP    255  0      4.4.4.9    NULL0
21.1.1.0/24         BGP    255  0      4.4.4.9    NULL0
21.1.1.2/32         BGP    255  0      4.4.4.9    NULL0

```

Execute the **display ip routing-table** command on CE 1 and CE 2. The output shows that the internal routes of the customer carrier network are present in the public network routing tables, but the VPN routes that the customer carrier maintains are not. Take CE 1 as an example:

```
[CE1] display ip routing-table
```

```
Routing Tables: Public
```

```

          Destinations : 16          Routes : 16
Destination/Mask    Proto  Pre  Cost   NextHop    Interface
1.1.1.9/32          ISIS   15   10     10.1.1.2   POS5/0
2.2.2.9/32          Direct 0    0      127.0.0.1  InLoop0
5.5.5.9/32          ISIS   15   74     11.1.1.2   POS5/1
6.6.6.9/32          ISIS   15   74     11.1.1.2   POS5/1
10.1.1.0/24         Direct 0    0      10.1.1.2   POS5/0
10.1.1.1/32         Direct 0    0      10.1.1.1   POS5/0
10.1.1.2/32         Direct 0    0      127.0.0.1  InLoop0
11.1.1.0/24         Direct 0    0      11.1.1.1   POS5/1
11.1.1.1/32         Direct 0    0      127.0.0.1  InLoop0
11.1.1.2/32         Direct 0    0      11.1.1.2   POS5/1
20.1.1.0/24         ISIS   15   74     11.1.1.2   POS5/1
21.1.1.0/24         ISIS   15   74     11.1.1.2   POS5/1
21.1.1.2/32         ISIS   15   74     11.1.1.2   POS5/1
127.0.0.0/8         Direct 0    0      127.0.0.1  InLoop0
127.0.0.1/32        Direct 0    0      127.0.0.1  InLoop0

```

Execute the **display ip routing-table** command on PE 3 and PE 4. The output shows that the internal routes of the customer carrier network are present in the public network routing tables. Take PE 3 as an example:

```
[PE3] display ip routing-table
```

```
Routing Tables: Public
```

```

          Destinations : 11          Routes : 11

```

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
1.1.1.9/32	Direct	0	0	127.0.0.1	InLoop0
2.2.2.9/32	ISIS	15	10	10.1.1.2	POS5/1
5.5.5.9/32	ISIS	15	84	10.1.1.2	POS5/1
6.6.6.9/32	ISIS	15	84	10.1.1.2	POS5/1
10.1.1.0/24	Direct	0	0	10.1.1.1	POS5/1
10.1.1.1/32	Direct	0	0	127.0.0.1	InLoop0
10.1.1.2/32	Direct	0	0	10.1.1.2	POS5/1
11.1.1.0/24	ISIS	15	20	10.1.1.2	POS5/1
20.1.1.0/24	ISIS	15	84	10.1.1.2	POS5/1
21.1.1.0/24	ISIS	15	84	10.1.1.2	POS5/1
21.1.1.2/32	ISIS	15	84	10.1.1.2	POS5/1
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0

Execute the **display ip routing-table vpn-instance** command on PE 3 and PE 4. The output shows that the routes of the remote VPN customers are present in the VPN routing tables. Take PE 3 as an example:

```
[PE3] display ip routing-table vpn-instance vpn1
```

```
Routing Tables: vpn1
```

```
Destinations : 3          Routes : 3
```

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
100.1.1.0/24	Direct	0	0	100.1.1.2	Eth1/1
100.1.1.2/32	Direct	0	0	127.0.0.1	InLoop0
120.1.1.0/24	BGP	255	0	6.6.6.9	NULL0

PE 3 and PE 4 can ping each other.

CE 3 and CE 4 can ping each other.

Configuring nested VPN

Network requirements

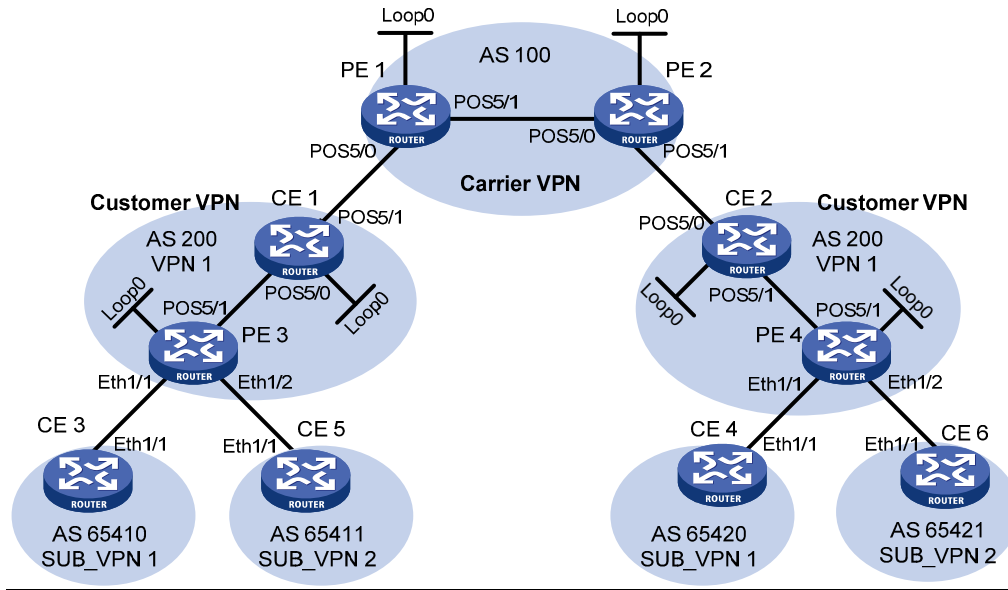
The service provider provides nested VPN services for users, as shown in [Figure 55](#).

- PE 1 and PE 2 are PE devices on the service provider backbone. Both of them support the nested VPN function.
- CE 1 and CE 2 are provider CEs connected to the service provider backbone. Both of them support VPNv4 routes.
- PE 3 and PE 4 are PE devices of the customer VPN. Both of them support MPLS L3VPN.
- CE 3 through CE 6 are CE devices of sub-VPNs in the customer VPN.

The key of nested VPN configuration is to understand the processing of routes of sub-VPNs on the service provider PEs:

- When receiving a VPNv4 route from a provider CE (CE 1 or CE 2, in this example), a provider PE replaces the RD of the VPNv4 route with the RD of the MPLS VPN on the service provider network, adds the export target attribute of the MPLS VPN on the service provider network to the extended community attribute list, and then forwards the VPNv4 route.
- To implement exchange of sub-VPN routes between customer PEs and service provider PEs, MP-EBGP peers must be established between provider PEs and provider CEs.

Figure 55 Network diagram



Device	Interface	IP address	Device	Interface	IP address
CE 1	Loop0	2.2.2.9/32	CE 2	Loop0	5.5.5.9/32
	POS5/0	10.1.1.2/24		POS5/0	21.1.1.2/24
	POS5/1	11.1.1.1/24		POS5/1	20.1.1.1/24
CE 3	Eth1/1	100.1.1.1/24	CE 4	Eth1/1	120.1.1.1/24
CE 5	Eth1/1	110.1.1.1/24	CE 6	Eth1/1	130.1.1.1/24
PE 1	Loop0	3.3.3.9/32	PE 2	Loop0	4.4.4.9/32
	POS5/0	11.1.1.2/24		POS5/0	30.1.1.2/24
	POS5/1	30.1.1.1/24		POS5/1	21.1.1.1/24
PE 3	Loop0	1.1.1.9/32	PE 4	Loop0	6.6.6.9/32
	Eth1/1	100.1.1.2/24		Eth1/1	120.1.1.2/24
	Eth1/2	110.1.1.2/24		Eth1/2	130.1.1.2/24
	POS5/1	10.1.1.1/24		POS5/1	20.1.1.2/24

Configuration procedure

1. Configure MPLS L3VPN on the service provider backbone—enable IS-IS, enable LDP, and establish an MP-IBGP peer relationship between PE 1 and PE 2:

Configure PE 1.

```

<PE1> system-view
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 3.3.3.9 32
[PE1-LoopBack0] quit
[PE1] mpls lsr-id 3.3.3.9
[PE1] mpls ldp
[PE1-ldp] quit
[PE1] isis 1
[PE1-isis-1] network-entity 10.0000.0000.0000.0004.00
[PE1-isis-1] quit
[PE1] interface loopback 0
[PE1-LoopBack0] isis enable 1
    
```

```

[PE1-LoopBack0] quit
[PE1] interface pos 5/1
[PE1-POS5/1] ip address 30.1.1.1 24
[PE1-POS5/1] isis enable 1
[PE1-POS5/1] mpls enable
[PE1-POS5/1] mpls ldp enable
[PE1-POS5/1] mpls ldp transport-address interface
[PE1-POS5/1] quit
[PE1] bgp 100
[PE1-bgp] peer 4.4.4.9 as-number 100
[PE1-bgp] peer 4.4.4.9 connect-interface loopback 0
[PE1-bgp] address-family vpnv4
[PE1-bgp-vpnv4] peer 4.4.4.9 enable
[PE1-bgp-vpnv4] quit
[PE1-bgp] quit

```

Configure PE 2 in the same way that PE 1 is configured. (Details not shown.)

After completing the configurations, execute commands **display mpls ldp peer**, **display bgp peer vpnv4**, and **display isis peer** on either PE 1 or PE 2. The output shows that the LDP session has been established, the BGP peer relationship has been established and reached the Established state, and the IS-IS neighbor relationship has been established.

Take PE 1 as an example:

```
[PE1] display mpls ldp peer
```

```
Total number of peers: 1
```

Peer LDP ID	State	LAM	Role	GR	MD5	KA Sent/Rcvd
4.4.4.9:0	Operational	DU	Active	Off	Off	8/8

```
[PE1] display bgp peer vpnv4
```

```
BGP local router ID: 3.3.3.9
```

```
Local AS number: 100
```

```
Total number of peers: 1                Peers in established state: 1
```

Peer	AS	MsgRcvd	MsgSent	OutQ	PrefRcv	Up/Down	State
4.4.4.9	100	3	6	0	0	00:00:32	Established

```
[PE1] display isis peer
```

```
Peer information for ISIS(1)
```

```
System Id: 0000.0000.0005
```

```
Interface: POS5/1                Circuit Id: 0000.0000.0005.02
```

```
State: Up      HoldTime: 8s      Type: L1(L1L2)      PRI: 64
```

```
System Id: 0000.0000.0005
```

```
Interface: POS5/1                Circuit Id: 0000.0000.0005.02
```

```
State: Up      HoldTime: 8s      Type: L2(L1L2)      PRI: 64
```

2. Configure the customer VPN—enable IS-IS and enable LDP between PE 3 and CE 1, and between PE 4 and CE 2:

Configure PE 3.

```

<PE3> system-view
[PE3] interface loopback 0
[PE3-LoopBack0] ip address 1.1.1.9 32
[PE3-LoopBack0] quit
[PE3] mpls lsr-id 1.1.1.9
[PE3] mpls ldp
[PE3-ldp] quit
[PE3] isis 2
[PE3-isis-2] network-entity 10.0000.0000.0000.0001.00
[PE3-isis-2] quit
[PE3] interface loopback 0
[PE3-LoopBack0] isis enable 2
[PE3-LoopBack0] quit
[PE3] interface pos 5/1
[PE3-POS5/1] ip address 10.1.1.1 24
[PE3-POS5/1] isis enable 2
[PE3-POS5/1] mpls enable
[PE3-POS5/1] mpls ldp enable
[PE3-POS5/1] quit

```

Configure CE 1.

```

<CE1> system-view
[CE1] interface loopback 0
[CE1-LoopBack0] ip address 2.2.2.9 32
[CE1-LoopBack0] quit
[CE1] mpls lsr-id 2.2.2.9
[CE1] mpls ldp
[CE1-ldp] quit
[CE1] isis 2
[CE1-isis-2] network-entity 10.0000.0000.0000.0002.00
[CE1-isis-2] quit
[CE1] interface loopback 0
[CE1-LoopBack0] isis enable 2
[CE1-LoopBack0] quit
[CE1] interface pos 5/0
[CE1-POS5/0] ip address 10.1.1.2 24
[CE1-POS5/0] isis enable 2
[CE1-POS5/0] mpls enable
[CE1-POS5/0] mpls ldp enable
[CE1-POS5/0] quit

```

After the configurations, an LDP session and IS-IS neighbor relationship can be established between PE 3 and CE 1.

Configure PE 4 and CE 2 in the same way that PE 3 and CE 1 are configured. (Details not shown.)

3. Connect CE 1 and CE 2 to service provider PEs:

Configure PE 1.

```

[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 200:1

```

```

[PE1-vpn-instance-vpn1] vpn-target 1:1
[PE1-vpn-instance-vpn1] quit
[PE1] interface pos 5/0
[PE1-POS5/0] ip binding vpn-instance vpn1
[PE1-POS5/0] ip address 11.1.1.2 24
[PE1-POS5/0] mpls enable
[PE1-POS5/0] quit
[PE1] bgp 100
[PE1-bgp] ip vpn-instance vpn1
[PE1-bgp-vpn1] peer 11.1.1.1 as-number 200
[PE1-bgp-vpn1] quit
[PE1-bgp] quit

```

Configure CE 1.

```

[CE1] interface pos 5/1
[CE1-POS5/1] ip address 11.1.1.1 24
[CE1-POS5/1] mpls enable
[CE1-POS5/1] quit
[CE1] bgp 200
[CE1-bgp] peer 11.1.1.2 as-number 100
[CE1-bgp] quit

```

Configure PE 2 and CE 2 in the same way that PE 1 and CE 1 are configured. (Details not shown.)

4. Connect sub-VPN CEs to the customer VPN PEs:

Configure CE 3.

```

<CE3> system-view
[CE3] interface ethernet 1/1
[CE3-Ethernet1/1] ip address 100.1.1.1 24
[CE3-Ethernet1/1] quit
[CE3] bgp 65410
[CE3-bgp] peer 100.1.1.2 as-number 200
[CE3-bgp] address-family ipv4 unicast
[CE3-bgp-ipv4] peer 100.1.1.2 enable
[CE3-bgp-ipv4] import-route direct
[CE3-bgp-ipv4] quit
[CE3-bgp] quit

```

Configure CE 5.

```

<CE5> system-view
[CE5] interface ethernet 1/1
[CE5-Ethernet1/1] ip address 110.1.1.1 24
[CE5-Ethernet1/1] quit
[CE5] bgp 65411
[CE5-bgp] peer 110.1.1.2 as-number 200
[CE5-bgp] address-family ipv4 unicast
[CE5-bgp-ipv4] peer 110.1.1.2 enable
[CE5-bgp-ipv4] import-route direct
[CE5-bgp-ipv4] quit
[CE5-bgp] quit

```

Configure PE 3.

```
[PE3] ip vpn-instance SUB_VPN1
[PE3-vpn-instance-SUB_VPN1] route-distinguisher 100:1
[PE3-vpn-instance-SUB_VPN1] vpn-target 2:1
[PE3-vpn-instance-SUB_VPN1] quit
[PE3] interface ethernet 1/1
[PE3-Ethernet1/1] ip binding vpn-instance SUB_VPN1
[PE3-Ethernet1/1] ip address 100.1.1.2 24
[PE3-Ethernet1/1] quit
[PE3] ip vpn-instance SUB_VPN2
[PE3-vpn-instance-SUB_VPN2] route-distinguisher 101:1
[PE3-vpn-instance-SUB_VPN2] vpn-target 2:2
[PE3-vpn-instance-SUB_VPN2] quit
[PE3] interface ethernet 1/2
[PE3-Ethernet1/2] ip binding vpn-instance SUB_VPN2
[PE3-Ethernet1/2] ip address 110.1.1.2 24
[PE3-Ethernet1/2] quit
[PE3] bgp 200
[PE3-bgp] ip vpn-instance SUB_VPN1
[PE3-bgp-SUB_VPN1] peer 100.1.1.1 as-number 65410
[PE3-bgp-SUB_VPN1] address-family ipv4 unicast
[PE3-bgp-ipv4-SUB_VPN1] peer 100.1.1.1 enable
[PE3-bgp-ipv4-SUB_VPN1] import-route direct
[PE3-bgp-ipv4-SUB_VPN1] quit
[PE3-bgp-SUB_VPN1] quit
[PE3-bgp] ip vpn-instance SUB_VPN2
[PE3-bgp-SUB_VPN2] peer 110.1.1.1 as-number 65411
[PE3-bgp-SUB_VPN2] address-family ipv4 unicast
[PE3-bgp-ipv4-SUB_VPN2] peer 110.1.1.1 enable
[PE3-bgp-ipv4-SUB_VPN2] import-route direct
[PE3-bgp-ipv4-SUB_VPN2] quit
[PE3-bgp-SUB_VPN2] quit
[PE3-bgp] quit
```

Configure PE 4, CE 4 and CE 6 in the same way that PE 3, CE 3, and CE 5 are configured. (Details not shown.)

5. Establish MP-EBGP peer relationship between service provider PEs and their CEs to exchange user VPNv4 routes:

On PE 1, enable nested VPN, and enable VPNv4 route exchange with CE 1.

```
[PE1] bgp 100
[PE1-bgp] address-family vpnv4
[PE1-bgp-vpnv4] nesting-vpn
[PE1-bgp-vpnv4] quit
[PE1-bgp] ip vpn-instance vpn1
[PE1-bgp-vpn1] address-family vpnv4
[PE1-bgp-vpnv4-vpn1] peer 11.1.1.1 enable
[PE1-bgp-vpnv4-vpn1] quit
[PE1-bgp-vpn1] quit
[PE1-bgp] quit
```



```

# On CE 1, enable VPNv4 route exchange with PE 1.
[CE1] bgp 200
[CE1-bgp] address-family vpnv4
[CE1-bgp-vpnv4] peer 11.1.1.2 enable

# Allow the local AS number to appear in the AS-PATH attribute of the routes received.
[CE1-bgp-vpnv4] peer 11.1.1.2 allow-as-loop 2

# Disable route target based filtering of received VPNv4 routes.
[CE1-bgp-vpnv4] undo policy vpn-target
[CE1-bgp-vpnv4] quit
[CE1-bgp] quit

# Configure PE 2 and CE 2 in the same way that PE 1 and CE 1 are configured. (Details not
shown.)
6. Establish MP-IBGP peer relationships between sub-VPN PEs and CEs of the customer VPN to
exchange VPNv4 routes of sub-VPNs:

# Configure PE 3.
[PE3] bgp 200
[PE3-bgp] peer 2.2.2.9 as-number 200
[PE3-bgp] peer 2.2.2.9 connect-interface loopback 0
[PE3-bgp] address-family vpnv4
[PE3-bgp-vpnv4] peer 2.2.2.9 enable

# Allow the local AS number to appear in the AS-PATH attribute of the routes received.
[PE3-bgp-vpnv4] peer 2.2.2.9 allow-as-loop 2
[PE3-bgp-vpnv4] quit
[PE3-bgp] quit

# Configure CE 1.
[CE1] bgp 200
[CE1-bgp] peer 1.1.1.9 as-number 200
[CE1-bgp] peer 1.1.1.9 connect-interface loopback 0
[CE1-bgp] address-family vpnv4
[CE1-bgp-vpnv4] peer 1.1.1.9 enable
[CE1-bgp-vpnv4] undo policy vpn-target
[CE1-bgp-vpnv4] quit
[CE1-bgp] quit

# Configure PE 4 and CE 2 in the same way that PE 3 and CE 1 are configured. (Details not
shown.)

```

Verifying the configuration

After completing all the configurations, execute the **display ip routing-table** command on PE 1 and PE 2 to verify that the public routing tables contain only routes on the service provider network. Take PE 1 as an example:

```
[PE1] display ip routing-table
```

```
Destinations : 14          Routes : 14
```

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
0.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
3.3.3.9/32	Direct	0	0	127.0.0.1	InLoop0

```

4.4.4.9/32          ISIS    15   10           30.1.1.2       POS5/1
30.1.1.0/24        Direct  0    0           30.1.1.1       POS5/1
30.1.1.1/32        Direct  0    0           127.0.0.1      InLoop0
30.1.1.2/32        Direct  0    0           30.1.1.2       POS5/1
30.1.1.255/32     Direct  0    0           30.1.1.2       POS5/1
127.0.0.0/8        Direct  0    0           127.0.0.1      InLoop0
127.0.0.0/32       Direct  0    0           127.0.0.1      InLoop0
127.0.0.1/32      Direct  0    0           127.0.0.1      InLoop0
127.255.255.255/32 Direct  0    0           127.0.0.1      InLoop0
224.0.0.0/4        Direct  0    0           0.0.0.0        NULL0
224.0.0.0/24       Direct  0    0           0.0.0.0        NULL0
255.255.255.255/32 Direct  0    0           127.0.0.1      InLoop0

```

Execute the **display ip routing-table vpn-instance** command on PE 1 and PE 2 to verify that the VPN routing tables contain sub-VPN routes. Take PE 1 as an example:

```
[PE1] display ip routing-table vpn-instance vpn1
```

```
Destinations : 16          Routes : 16
```

```

Destination/Mask    Proto  Pre  Cost           NextHop         Interface
0.0.0.0/32         Direct  0    0           127.0.0.1      InLoop0
11.1.1.0/24        Direct  0    0           11.1.1.1       POS5/0
11.1.1.1/32        Direct  0    0           127.0.0.1      InLoop0
11.1.1.2/32        Direct  0    0           11.1.1.2       POS5/0
11.1.1.255/32     Direct  0    0           11.1.1.2       POS5/0
100.1.1.0/24      BGP    255  0           11.1.1.1       NULL0
110.1.1.0/24      BGP    255  0           11.1.1.1       NULL0
120.1.1.0/24      BGP    255  0           4.4.4.9        NULL0
127.0.0.0/8        Direct  0    0           127.0.0.1      InLoop0
127.0.0.0/32       Direct  0    0           127.0.0.1      InLoop0
127.0.0.1/32      Direct  0    0           127.0.0.1      InLoop0
127.255.255.255/32 Direct  0    0           127.0.0.1      InLoop0
130.1.1.0/24      BGP    255  0           4.4.4.9        NULL0
224.0.0.0/4        Direct  0    0           0.0.0.0        NULL0
224.0.0.0/24       Direct  0    0           0.0.0.0        NULL0
255.255.255.255/32 Direct  0    0           127.0.0.1      InLoop0

```

Execute the **display bgp routing-table vpnv4** command on CE 1 and CE 2 to verify that the VPNv4 routing tables on the customer VPN contain internal sub-VPN routes. Take CE 1 as an example:

```
[CE1] display bgp routing-table vpnv4
```

```
BGP Local router ID is 11.11.11.11
```

```

Status codes: * - valid, > - best, d - damped, h - history,
              s - suppressed, S - Stale, i - internal, e - external
Origin: i - IGP, e - EGP, ? - incomplete

```

```
Total number of routes from all PEs: 4
```

```
Route Distinguisher: 100:1
```

```
Total number of routes: 1
```

Network	NextHop	MED	LocPrf	PrefVal	Path/Ogn
* > 100.1.1.0/24	1.1.1.9			0	200 65410?

Route Distinguisher: 101:1
Total number of routes: 1

Network	NextHop	MED	LocPrf	PrefVal	Path/Ogn
* > 110.1.1.0/24	1.1.1.9			0	200 65411?

Route Distinguisher: 200:1
Total number of routes: 1

Network	NextHop	MED	LocPrf	PrefVal	Path/Ogn
* > 120.1.1.0/24	11.1.1.2			0	100 200 65420?

Route Distinguisher: 201:1
Total number of routes: 1

Network	NextHop	MED	LocPrf	PrefVal	Path/Ogn
* > 130.1.1.0/24	11.1.1.2			0	100 200 65421?

Execute the **display ip routing-table vpn-instance SUB_VPN1** command on PE 3 and PE 4 to verify that the VPN routing tables contain routes sent by provider PEs to sub-VPNs. Take PE 3 as an example:

[PE3] display ip routing-table vpn-instance SUB_VPN1

Destinations : 13 Routes : 13

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
0.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
100.1.1.0/24	Direct	0	0	100.1.1.2	Eth1/1
100.1.1.0/32	Direct	0	0	100.1.1.2	Eth1/1
100.1.1.2/32	Direct	0	0	127.0.0.1	InLoop0
100.1.1.255/32	Direct	0	0	100.1.1.2	Eth1/1
120.1.1.0/24	BGP	255	0	2.2.2.9	NULL0
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0
127.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0
224.0.0.0/4	Direct	0	0	0.0.0.0	NULL0
224.0.0.0/24	Direct	0	0	0.0.0.0	NULL0
255.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0

Execute the **display ip routing-table** command on CE 3 and CE 4 to verify that the routing tables contain routes of remote sub-VPNs. Take CE 3 as an example:

```
[CE3] display ip routing-table
```

```
Destinations : 13          Routes : 13
```

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
0.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
100.1.1.0/24	Direct	0	0	100.1.1.1	Eth1/1
100.1.1.0/32	Direct	0	0	100.1.1.1	Eth1/1
100.1.1.1/32	Direct	0	0	127.0.0.1	InLoop0
100.1.1.255/24	Direct	0	0	100.1.1.1	Eth1/1
120.1.1.0/24	BGP	255	0	100.1.1.2	Eth1/1
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0
127.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0
224.0.0.0/4	Direct	0	0	0.0.0.0	NULL0
224.0.0.0/24	Direct	0	0	0.0.0.0	NULL0
255.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0

Execute the **display ip routing-table** command on CE 5 and CE 6 to verify that the routing tables contain routes of remote sub-VPNs. Take CE5 as an example:

```
[CE5] display ip routing-table
```

```
Destinations : 13          Routes : 13
```

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
0.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
110.1.1.0/24	Direct	0	0	110.1.1.1	Eth1/1
110.1.1.0/32	Direct	0	0	110.1.1.1	Eth1/1
110.1.1.1/32	Direct	0	0	127.0.0.1	InLoop0
110.1.1.255/32	Direct	0	0	110.1.1.1	Eth1/1
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0
127.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0
130.1.1.0/24	BGP	255	0	110.1.1.2	Eth1/1
224.0.0.0/4	Direct	0	0	0.0.0.0	NULL0
224.0.0.0/24	Direct	0	0	0.0.0.0	NULL0
255.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0

CE 3 and CE 4 can ping each other.

CE 5 and CE 6 can ping each other.

CE 3 and CE 6 cannot ping each other.

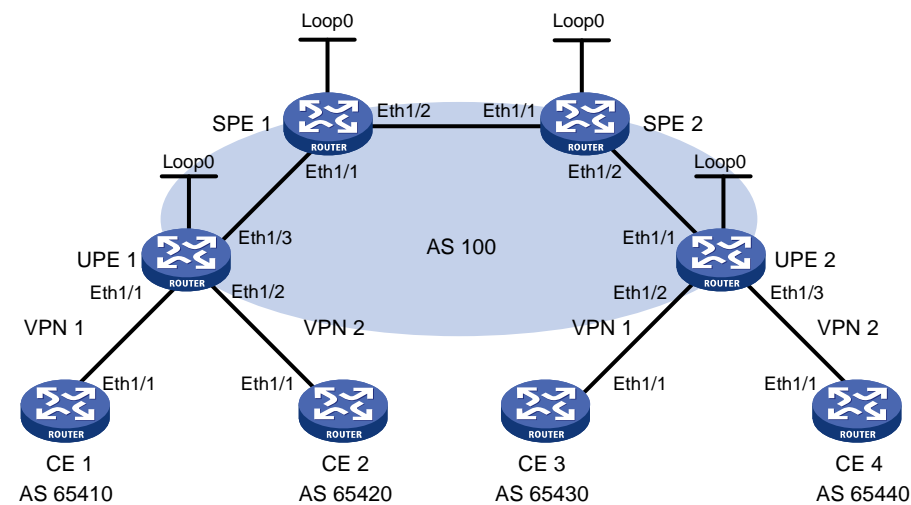
Configuring HoVPN

Network requirements

As shown in Figure 56, there are two levels of networks: the backbone and the MPLS VPN networks.

- SPEs act as PEs to allow MPLS VPNs to access the backbone.
- UPEs act as PEs of the MPLS VPNs to allow end users to access the VPNs.
- Performance requirements for the UPEs are lower than those for the SPEs.
- SPEs advertise routes permitted by routing policies to UPEs, permitting CE 1 and CE 3 in VPN 1 to communicate with each other and forbidding CE 2 and CE 4 in VPN 2 to communicate with each other.

Figure 56 Network diagram



Device	Interface	IP address	Device	Interface	IP address
CE 1	Eth1/1	10.2.1.1/24	CE 3	Eth1/1	10.1.1.1/24
CE 2	Eth1/1	10.4.1.1/24	CE 4	Eth1/1	10.3.1.1/24
UPE 1	Loop0	1.1.1.9/32	UPE 2	Loop0	4.4.4.9/32
	Eth1/1	10.2.1.2/24		Eth1/1	172.2.1.1/24
	Eth1/2	10.4.1.2/24		Eth1/2	10.1.1.2/24
	Eth1/3	172.1.1.1/24		Eth1/3	10.3.1.2/24
SPE 1	Loop0	2.2.2.9/32	SPE 2	Loop0	3.3.3.9/32
	Eth1/1	172.1.1.2/24		Eth1/1	180.1.1.2/24
	Eth1/2	180.1.1.1/24		Eth1/2	172.2.1.2/24

Configuration procedure

1. Configure UPE 1:

Configure basic MPLS and MPLS LDP to establish LDP LSPs.

```
<UPE1> system-view
[UPE1] interface loopback 0
[UPE1-LoopBack0] ip address 1.1.1.9 32
[UPE1-LoopBack0] quit
[UPE1] mpls lsr-id 1.1.1.9
[UPE1] mpls ldp
```

```

[UEP1-ldp] quit
[UEP1] interface ethernet 1/3
[UEP1-Ethernet1/3] ip address 172.1.1.1 24
[UEP1-Ethernet1/3] mpls enable
[UEP1-Ethernet1/3] mpls ldp enable
[UEP1-Ethernet1/3] quit
# Configure the IGP protocol (OSPF, in this example).
[UEP1] ospf
[UEP1-ospf-1] area 0
[UEP1-ospf-1-area-0.0.0.0] network 172.1.1.0 0.0.0.255
[UEP1-ospf-1-area-0.0.0.0] network 1.1.1.9 0.0.0.0
[UEP1-ospf-1-area-0.0.0.0] quit
[UEP1-ospf-1] quit
# Configure VPN instances vpn1 and vpn2, allowing CE 1 and CE 2 to access UPE 1.
[UEP1] ip vpn-instance vpn1
[UEP1-vpn-instance-vpn1] route-distinguisher 100:1
[UEP1-vpn-instance-vpn1] vpn-target 100:1 both
[UEP1-vpn-instance-vpn1] quit
[UEP1] ip vpn-instance vpn2
[UEP1-vpn-instance-vpn2] route-distinguisher 100:2
[UEP1-vpn-instance-vpn2] vpn-target 100:2 both
[UEP1-vpn-instance-vpn2] quit
[UEP1] interface ethernet 1/1
[UEP1-Ethernet1/1] ip binding vpn-instance vpn1
[UEP1-Ethernet1/1] ip address 10.2.1.2 24
[UEP1-Ethernet1/1] quit
[UEP1] interface ethernet 1/2
[UEP1-Ethernet1/2] ip binding vpn-instance vpn2
[UEP1-Ethernet1/2] ip address 10.4.1.2 24
[UEP1-Ethernet1/2] quit
# Establish an MP-IBGP peer relationship with SPE 1.
[UEP1] bgp 100
[UEP1-bgp] peer 2.2.2.9 as-number 100
[UEP1-bgp] peer 2.2.2.9 connect-interface loopback 0
[UEP1-bgp] address-family vpnv4
[UEP1-bgp-vpnv4] peer 2.2.2.9 enable
[UEP1-bgp-vpnv4] quit
# Establish an EBGp peer relationship with CE 1, and redistribute VPN routes into BGP.
[UEP1-bgp] ip vpn-instance vpn1
[UEP1-bgp-vpn1] peer 10.2.1.1 as-number 65410
[UEP1-bgp-vpn1] address-family ipv4 unicast
[UEP1-bgp-ipv4-vpn1] peer 10.2.1.1 enable
[UEP1-bgp-ipv4-vpn1] import-route direct
[UEP1-bgp-ipv4-vpn1] quit
[UEP1-bgp-vpn1] quit
# Establish an EBGp peer relationship with CE 2, and redistribute VPN routes into BGP.
[UEP1-bgp] ip vpn-instance vpn2

```

```

[UPE1-bgp-vpn2] peer 10.4.1.1 as-number 65420
[UPE1-bgp-vpn2] address-family ipv4 unicast
[UPE1-bgp-ipv4-vpn2] peer 10.4.1.1 enable
[UPE1-bgp-ipv4-vpn2] import-route direct
[UPE1-bgp-ipv4-vpn2] quit
[UPE1-bgp-vpn2] quit
[UPE1-bgp] quit

```

2. Configure CE 1:

```

<CE1> system-view
[CE1] interface ethernet 1/1
[CE1-Ethernet1/1] ip address 10.2.1.1 255.255.255.0
[CE1-Ethernet1/1] quit
[CE1] bgp 65410
[CE1-bgp] peer 10.2.1.2 as-number 100
[CE1-bgp] address-family ipv4 unicast
[CE1-bgp-ipv4] peer 10.2.1.2 enable
[CE1-bgp-ipv4] import-route direct
[CE1-bgp-ipv4] quit
[CE1-bgp] quit

```

3. Configure CE 2:

```

<CE2> system-view
[CE2] interface ethernet 1/1
[CE2-Ethernet1/1] ip address 10.4.1.1 255.255.255.0
[CE2-Ethernet1/1] quit
[CE2] bgp 65420
[CE2-bgp] peer 10.4.1.2 as-number 100
[CE2-bgp] address-family ipv4 unicast
[CE2-bgp-ipv4] peer 10.4.1.2 enable
[CE2-bgp-ipv4] import-route direct
[CE2-bgp-ipv4] quit
[CE2-bgp] quit

```

4. Configure UPE 2:

Configure basic MPLS and MPLS LDP to establish LDP LSPs.

```

<UPE2> system-view
[UPE2] interface loopback 0
[UPE2-LoopBack0] ip address 4.4.4.9 32
[UPE2-LoopBack0] quit
[UPE2] mpls lsr-id 4.4.4.9
[UPE2] mpls ldp
[UPE2-ldp] quit
[UPE2] interface ethernet 1/1
[UPE2-Ethernet1/1] ip address 172.2.1.1 24
[UPE2-Ethernet1/1] mpls enable
[UPE2-Ethernet1/1] mpls ldp enable
[UPE2-Ethernet1/1] quit

```

Configure the IGP protocol (OSPF, in this example).

```

[UPE2] ospf

```

```

[UPE2-ospf-1] area 0
[UPE2-ospf-1-area-0.0.0.0] network 172.2.1.0 0.0.0.255
[UPE2-ospf-1-area-0.0.0.0] network 4.4.4.9 0.0.0.0
[UPE2-ospf-1-area-0.0.0.0] quit
[UPE2-ospf-1] quit
# Configure VPN instances vpn1 and vpn2, allowing CE 3 and CE 4 to access UPE 2.
[UPE2] ip vpn-instance vpn1
[UPE2-vpn-instance-vpn1] route-distinguisher 300:1
[UPE2-vpn-instance-vpn1] vpn-target 100:1 both
[UPE2-vpn-instance-vpn1] quit
[UPE2] ip vpn-instance vpn2
[UPE2-vpn-instance-vpn2] route-distinguisher 400:2
[UPE2-vpn-instance-vpn2] vpn-target 100:2 both
[UPE2-vpn-instance-vpn2] quit
[UPE2] interface ethernet 1/2
[UPE2-Ethernet1/2] ip binding vpn-instance vpn1
[UPE2-Ethernet1/2] ip address 10.1.1.2 24
[UPE2-Ethernet1/2] quit
[UPE2] interface ethernet 1/3
[UPE2-Ethernet1/3] ip binding vpn-instance vpn2
[UPE2-Ethernet1/3] ip address 10.3.1.2 24
[UPE2-Ethernet1/3] quit
# Establish an MP-IBGP peer relationship with SPE 2.
[UPE2] bgp 100
[UPE2-bgp] peer 3.3.3.9 as-number 100
[UPE2-bgp] peer 3.3.3.9 connect-interface loopback 0
[UPE2-bgp] address-family vpnv4
[UPE2-bgp-vpnv4] peer 3.3.3.9 enable
[UPE2-bgp-vpnv4] quit
# Establish an EBGP peer relationship with CE 3, and redistribute VPN routes into BGP.
[UPE2-bgp] ip vpn-instance vpn1
[UPE2-bgp-vpn1] peer 10.1.1.1 as-number 65430
[UPE2-bgp-vpn1] address-family ipv4 unicast
[UPE2-bgp-ipv4-vpn1] peer 10.1.1.1 enable
[UPE2-bgp-ipv4-vpn1] import-route direct
[UPE2-bgp-ipv4-vpn1] quit
[UPE2-bgp-vpn1] quit
# Establish an EBGP peer relationship with CE 4, and redistribute VPN routes into BGP.
[UPE2-bgp] ip vpn-instance vpn2
[UPE2-bgp-vpn2] peer 10.3.1.1 as-number 65440
[UPE2-bgp-vpn2] address-family ipv4 unicast
[UPE2-bgp-ipv4-vpn2] peer 10.3.1.1 enable
[UPE2-bgp-ipv4-vpn2] import-route direct
[UPE2-bgp-ipv4-vpn2] quit
[UPE2-bgp-vpn2] quit
[UPE2-bgp] quit

```

5. Configure CE 3:


```

<CE3> system-view
[CE3] interface ethernet 1/1
[CE3-Ethernet1/1] ip address 10.1.1.1 255.255.255.0
[CE3-Ethernet1/1] quit
[CE3] bgp 65430
[CE3-bgp] peer 10.1.1.2 as-number 100
[CE3-bgp] address-family ipv4 unicast
[CE3-bgp-ipv4] peer 10.1.1.2 enable
[CE3-bgp-ipv4] import-route direct
[CE3-bgp-ipv4] quit
[CE3-bgp] quit

```

6. Configure CE 4:

```

<CE4> system-view
[CE4] interface ethernet 1/1
[CE4-Ethernet1/1] ip address 10.3.1.1 255.255.255.0
[CE4-Ethernet1/1] quit
[CE4] bgp 65440
[CE4-bgp] peer 10.3.1.2 as-number 100
[CE4-bgp] address-family ipv4 unicast
[CE4-bgp-ipv4] peer 10.3.1.2 enable
[CE4-bgp-ipv4] import-route direct
[CE4-bgp-ipv4] quit
[CE4-bgp] quit

```

7. Configure SPE 1:

Configure basic MPLS and MPLS LDP to establish LDP LSPs.

```

<SPE1> system-view
[SPE1] interface loopback 0
[SPE1-LoopBack0] ip address 2.2.2.9 32
[SPE1-LoopBack0] quit
[SPE1] mpls lsr-id 2.2.2.9
[SPE1] mpls ldp
[SPE1-ldp] quit
[SPE1] interface ethernet 1/1
[SPE1-Ethernet1/1] ip address 172.1.1.2 24
[SPE1-Ethernet1/1] mpls enable
[SPE1-Ethernet1/1] mpls ldp enable
[SPE1-Ethernet1/1] quit
[SPE1] interface ethernet 1/2
[SPE1-Ethernet1/2] ip address 180.1.1.1 24
[SPE1-Ethernet1/2] mpls enable
[SPE1-Ethernet1/2] mpls ldp enable
[SPE1-Ethernet1/2] quit

```

Configure the IGP protocol, OSPF, in this example.

```

[SPE1] ospf
[SPE1-ospf-1] area 0
[SPE1-ospf-1-area-0.0.0.0] network 2.2.2.9 0.0.0.0
[SPE1-ospf-1-area-0.0.0.0] network 172.1.1.0 0.0.0.255
[SPE1-ospf-1-area-0.0.0.0] network 180.1.1.0 0.0.0.255

```

```

[SPE1-ospf-1-area-0.0.0.0] quit
[SPE1-ospf-1] quit
# Configure VPN instances vpn1 and vpn2.
[SPE1] ip vpn-instance vpn1
[SPE1-vpn-instance-vpn1] route-distinguisher 500:1
[SPE1-vpn-instance-vpn1] vpn-target 100:1 both
[SPE1-vpn-instance-vpn1] quit
[SPE1] ip vpn-instance vpn2
[SPE1-vpn-instance-vpn2] route-distinguisher 700:1
[SPE1-vpn-instance-vpn2] vpn-target 100:2 both
[SPE1-vpn-instance-vpn2] quit
# Establish an MP-IBGP peer relationship with UPE 1, and redistribute VPN routes into BGP.
[SPE1] bgp 100
[SPE1-bgp] peer 1.1.1.9 as-number 100
[SPE1-bgp] peer 1.1.1.9 connect-interface loopback 0
[SPE1-bgp] peer 3.3.3.9 as-number 100
[SPE1-bgp] peer 3.3.3.9 connect-interface loopback 0
[SPE1-bgp] address-family vpnv4
[SPE1-bgp-vpnv4] peer 3.3.3.9 enable
[SPE1-bgp-vpnv4] peer 1.1.1.9 enable
[SPE1-bgp-vpnv4] peer 1.1.1.9 upe
[SPE1-bgp-vpnv4] peer 1.1.1.9 next-hop-local
[SPE1-bgp-vpnv4] quit
[SPE1-bgp] ip vpn-instance vpn1
[SPE1-bgp-vpn1] quit
[SPE1-bgp] ip vpn-instance vpn2
[SPE1-bgp-vpn2] quit
[SPE1-bgp] quit
# Advertise to UPE 1 the routes permitted by a routing policy (the routes of CE 3).
[SPE1] ip prefix-list hope index 10 permit 10.1.1.1 24
[SPE1] route-policy hope permit node 0
[SPE1-route-policy-hope-0] if-match ip address prefix-list hope
[SPE1-route-policy-hope-0] quit
[SPE1] bgp 100
[SPE1-bgp] address-family vpnv4
[SPE1-bgp-vpnv4] peer 1.1.1.9 upe route-policy hope export

```

8. Configure SPE 2:

Configure basic MPLS and MPLS LDP to establish LDP LSPs.

```

<SPE2> system-view
[SPE2] interface loopback 0
[SPE2-LoopBack0] ip address 3.3.3.9 32
[SPE2-LoopBack0] quit
[SPE2] mpls lsr-id 3.3.3.9
[SPE2] mpls ldp
[SPE2-ldp] quit
[SPE2] interface ethernet 1/1
[SPE2-Ethernet1/1] ip address 180.1.1.2 24

```

```

[SPE2-Ethernet1/1] mpls enable
[SPE2-Ethernet1/1] mpls ldp enable
[SPE2-Ethernet1/1] quit
[SPE2] interface ethernet 1/2
[SPE2-Ethernet1/2] ip address 172.2.1.2 24
[SPE2-Ethernet1/2] mpls enable
[SPE2-Ethernet1/2] mpls ldp enable
[SPE2-Ethernet1/2] quit
# Configure the IGP protocol, OSPF, in this example.
[SPE2] ospf
[SPE2-ospf-1] area 0
[SPE2-ospf-1-area-0.0.0.0] network 3.3.3.9 0.0.0.0
[SPE2-ospf-1-area-0.0.0.0] network 172.2.1.0 0.0.0.255
[SPE2-ospf-1-area-0.0.0.0] network 180.1.1.0 0.0.0.255
[SPE2-ospf-1-area-0.0.0.0] quit
[SPE2-ospf-1] quit
# Configure VPN instances vpn1 and vpn2.
[SPE2] ip vpn-instance vpn1
[SPE2-vpn-instance-vpn1] route-distinguisher 600:1
[SPE2-vpn-instance-vpn1] vpn-target 100:1 both
[SPE2-vpn-instance-vpn1] quit
[SPE2] ip vpn-instance vpn2
[SPE2-vpn-instance-vpn2] route-distinguisher 800:1
[SPE2-vpn-instance-vpn2] vpn-target 100:2 both
[SPE2-vpn-instance-vpn2] quit
# Establish an MP-IBGP peer relationship with UPE 2, and redistribute VPN routes.
[SPE2] bgp 100
[SPE2-bgp] peer 4.4.4.9 as-number 100
[SPE2-bgp] peer 4.4.4.9 connect-interface loopback 0
[SPE2-bgp] peer 2.2.2.9 as-number 100
[SPE2-bgp] peer 2.2.2.9 connect-interface loopback 0
[SPE2-bgp] address-family vpnv4
[SPE2-bgp-vpnv4] peer 2.2.2.9 enable
[SPE2-bgp-vpnv4] peer 4.4.4.9 enable
[SPE2-bgp-vpnv4] peer 4.4.4.9 upe
[SPE2-bgp-vpnv4] peer 4.4.4.9 next-hop-local
[SPE2-bgp-vpnv4] quit
[SPE2-bgp] ip vpn-instance vpn1
[SPE2-bgp-vpn1] quit
[SPE2-bgp] ip vpn-instance vpn2
[SPE2-bgp-vpn2] quit
[SPE2-bgp] quit
# Advertise to UPE 2 the routes permitted by a routing policy (the routes of CE 1).
[SPE2] ip prefix-list hope index 10 permit 10.2.1.1 24
[SPE2] route-policy hope permit node 0
[SPE2-route-policy-hope-0] if-match ip address prefix-list hope
[SPE2-route-policy-hope-0] quit
[SPE2] bgp 100

```

```
[SPE2-bgp] address-family vpnv4
[SPE2-bgp-vpnv4] peer 4.4.4.9 upe route-policy hope export
```

Verifying the configuration

After completing all the configurations, CE 1 and CE3 can learn each other's interface routes and can ping each other. CE 2 and CE 4 cannot learn each other's interface routes and cannot ping each other.

Configuring OSPF sham links

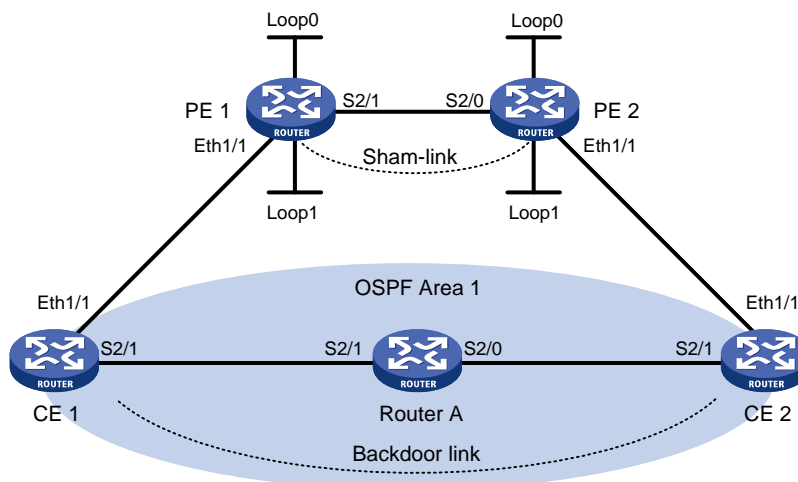
Network requirements

CE 1 and CE 2 belong to VPN 1 and are connected to PE 1 and PE 2.

CE 1 and CE 2 are in the same OSPF area.

VPN traffic between CE 1 and CE 2 is required to be forwarded through the MPLS backbone, instead of through the backdoor route in the OSPF area.

Figure 57 Network diagram



Device	Interface	IP address	Device	Interface	IP address
CE 1	Eth1/1	100.1.1.1/24	CE 2	Eth1/1	120.1.1.1/24
	S2/1	20.1.1.1/24		S2/1	30.1.1.2/24
PE 1	Loop0	1.1.1.9/32	PE 2	Loop0	2.2.2.9/32
	Loop1	3.3.3.3/32		Loop1	5.5.5.5/32
	Eth1/1	100.1.1.2/24		Eth1/1	120.1.1.2/24
	S2/1	10.1.1.1/24		S2/0	10.1.1.2/24
Router A	S2/0	30.1.1.1/24			
	S2/1	20.1.1.2/24			

Configuration procedure

1. Configure OSPF on the customer networks:
Configure conventional OSPF on CE 1, Router A, and CE 2 to advertise addresses of the interfaces as shown in Figure 57. After the configuration, execute the **display ip routing-table** command to see that CE 1 and CE 2 have learned the OSPF route to the Ethernet interface of each other. (Details not shown.)
2. Configure MPLS L3VPN on the backbone:

Configure basic MPLS and MPLS LDP on PE 1 to establish LDP LSPs.

```
<PE1> system-view
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 1.1.1.9 32
[PE1-LoopBack0] quit
[PE1] mpls lsr-id 1.1.1.9
[PE1] mpls ldp
[PE1-ldp] quit
[PE1] interface serial 2/1
[PE1-Serial2/1] ip address 10.1.1.1 24
[PE1-Serial2/1] mpls enable
[PE1-Serial2/1] mpls ldp enable
[PE1-Serial2/1] quit
```

Configure PE 1 to take PE 2 as an MP-IBGP peer.

```
[PE1] bgp 100
[PE1-bgp] peer 2.2.2.9 as-number 100
[PE1-bgp] peer 2.2.2.9 connect-interface loopback 0
[PE1-bgp] address-family vpnv4
[PE1-bgp-vpnv4] peer 2.2.2.9 enable
[PE1-bgp-vpnv4] quit
[PE1-bgp] quit
```

Configure OSPF on PE 1.

```
[PE1]ospf 1
[PE1-ospf-1]area 0
[PE1-ospf-1-area-0.0.0.0] network 1.1.1.9 0.0.0.0
[PE1-ospf-1-area-0.0.0.0] network 10.1.1.0 0.0.0.255
[PE1-ospf-1-area-0.0.0.0] quit
[PE1-ospf-1] quit
```

Configure basic MPLS and MPLS LDP on PE 2 to establish LDP LSPs.

```
<PE2> system-view
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 2.2.2.9 32
[PE2-LoopBack0] quit
[PE2] mpls lsr-id 2.2.2.9
[PE2] mpls ldp
[PE2-ldp] quit
[PE2] interface serial 2/1
[PE2-Serial2/1] ip address 10.1.1.2 24
[PE2-Serial2/1] mpls enable
[PE2-Serial2/1] mpls ldp enable
[PE2-Serial2/1] quit
```

Configure PE 2 to take PE 1 as an MP-IBGP peer.

```
[PE2] bgp 100
[PE2-bgp] peer 1.1.1.9 as-number 100
[PE2-bgp] peer 1.1.1.9 connect-interface loopback 0
[PE2-bgp] address-family vpnv4
[PE2-bgp-vpnv4] peer 1.1.1.9 enable
```

```

[PE2-bgp-vpnv4] quit
[PE2-bgp] quit
# Configure OSPF on PE 2.
[PE2] ospf 1
[PE2-ospf-1] area 0
[PE2-ospf-1-area-0.0.0.0] network 2.2.2.9 0.0.0.0
[PE2-ospf-1-area-0.0.0.0] network 10.1.1.0 0.0.0.255
[PE2-ospf-1-area-0.0.0.0] quit
[PE2-ospf-1] quit

```

3. Configure VPN instances on PEs:

Configure PE 1 to allow CE 1 to access the network.

```

[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 100:1
[PE1-vpn-instance-vpn1] vpn-target 1:1
[PE1-vpn-instance-vpn1] quit
[PE1] interface ethernet 1/1
[PE1-Ethernet1/1] ip binding vpn-instance vpn1
[PE1-Ethernet1/1] ip address 100.1.1.2 24
[PE1-Ethernet1/1] quit
[PE1] ospf 100 vpn-instance vpn1
[PE1-ospf-100] domain-id 10
[PE1-ospf-100] area 1
[PE1-ospf-100-area-0.0.0.1] network 100.1.1.0 0.0.0.255
[PE1-ospf-100-area-0.0.0.1] quit
[PE1-ospf-100] quit
[PE1] bgp 100
[PE1-bgp] ip vpn-instance vpn1
[PE1-bgp-vpn1] address-family ipv4 unicast
[PE1-bgp-ipv4-vpn1] import-route ospf 100
[PE1-bgp-ipv4-vpn1] import-route direct
[PE1-bgp-ipv4-vpn1] quit
[PE1-bgp-vpn1] quit
[PE1-bgp] quit

```

Configure PE 2 to allow CE 2 to access the network.

```

[PE2] ip vpn-instance vpn1
[PE2-vpn-instance-vpn1] route-distinguisher 100:2
[PE2-vpn-instance-vpn1] vpn-target 1:1
[PE2-vpn-instance-vpn1] quit
[PE2] interface ethernet 1/1
[PE2-Ethernet1/1] ip binding vpn-instance vpn1
[PE2-Ethernet1/1] ip address 120.1.1.2 24
[PE2-Ethernet1/1] quit
[PE2] ospf 100 vpn-instance vpn1
[PE2-ospf-100] domain-id 10
[PE2-ospf-100] area 1
[PE2-ospf-100-area-0.0.0.1] network 120.1.1.0 0.0.0.255
[PE2-ospf-100-area-0.0.0.1] quit
[PE2-ospf-100] quit

```

```

[PE2] bgp 100
[PE2-bgp] ip vpn-instance vpn1
[PE2-bgp-vpn1] address-family ipv4 unicast
[PE2-bgp-ipv4-vpn1] import-route ospf 100
[PE2-bgp-ipv4-vpn1] import-route direct
[PE2-bgp-ipv4-vpn1] quit
[PE2-bgp-vpn1] quit
[PE2-bgp] quit

```

After completing the configurations, execute the **display ip routing-table vpn-instance** command on the PEs. The path to the peer CE is along the OSPF route across the customer networks, instead of the BGP route across the backbone.

4. Configure a sham link:

Configure PE 1.

```

[PE1] interface loopback 1
[PE1-LoopBack1] ip binding vpn-instance vpn1
[PE1-LoopBack1] ip address 3.3.3.3 32
[PE1-LoopBack1] quit
[PE1] ospf 100
[PE1-ospf-100] area 1
[PE1-ospf-100-area-0.0.0.1] sham-link 3.3.3.3 5.5.5.5 cost 10
[PE1-ospf-100-area-0.0.0.1] quit
[PE1-ospf-100] quit

```

Configure PE 2.

```

[PE2] interface loopback 1
[PE2-LoopBack1] ip binding vpn-instance vpn1
[PE2-LoopBack1] ip address 5.5.5.5 32
[PE2-LoopBack1] quit
[PE2] ospf 100
[PE2-ospf-100] area 1
[PE2-ospf-100-area-0.0.0.1] sham-link 5.5.5.5 3.3.3.3 cost 10
[PE2-ospf-100-area-0.0.0.1] quit
[PE2-ospf-100] quit

```

Verifying the configuration

After completing the configurations, execute the **display ip routing-table vpn-instance** command again on the PEs. The path to the peer CE is now along the BGP route across the backbone, and that a route to the sham link destination address is present.

Execute the **display ip routing-table** command on the CEs. The next hop of the OSPF route to the peer CE is the Ethernet interface connected to the PE. This means that VPN traffic to the peer is forwarded over the backbone.

Execute the **display ospf sham-link** command on the PEs. The output shows that a sham link has been established. Take PE 1 as an example:

```
[PE1] display ospf sham-link
```

```

          OSPF Process 100 with Router ID 100.1.1.2
                Sham link
Area          Neighbor ID      Source IP      Destination IP  State  Cost

```

```
0.0.0.1          120.1.1.2          3.3.3.3          5.5.5.5          P-2-P 10
```

Execute the **display ospf sham-link area** command. The output shows that the peer state is Full:

```
[PE1] display ospf sham-link area 1
```

```
OSPF Process 100 with Router ID 100.1.1.2
```

```
Sham-Link: 3.3.3.3 --> 5.5.5.5
```

```
Neighbor ID: 120.1.1.2      State: Full
```

```
Area: 0.0.0.1
```

```
Cost: 10  State: P-2-P  Type: Sham
```

```
Timers: Hello 10s, Dead 40s, Retransmit 5s, Transmit Delay 1s
```

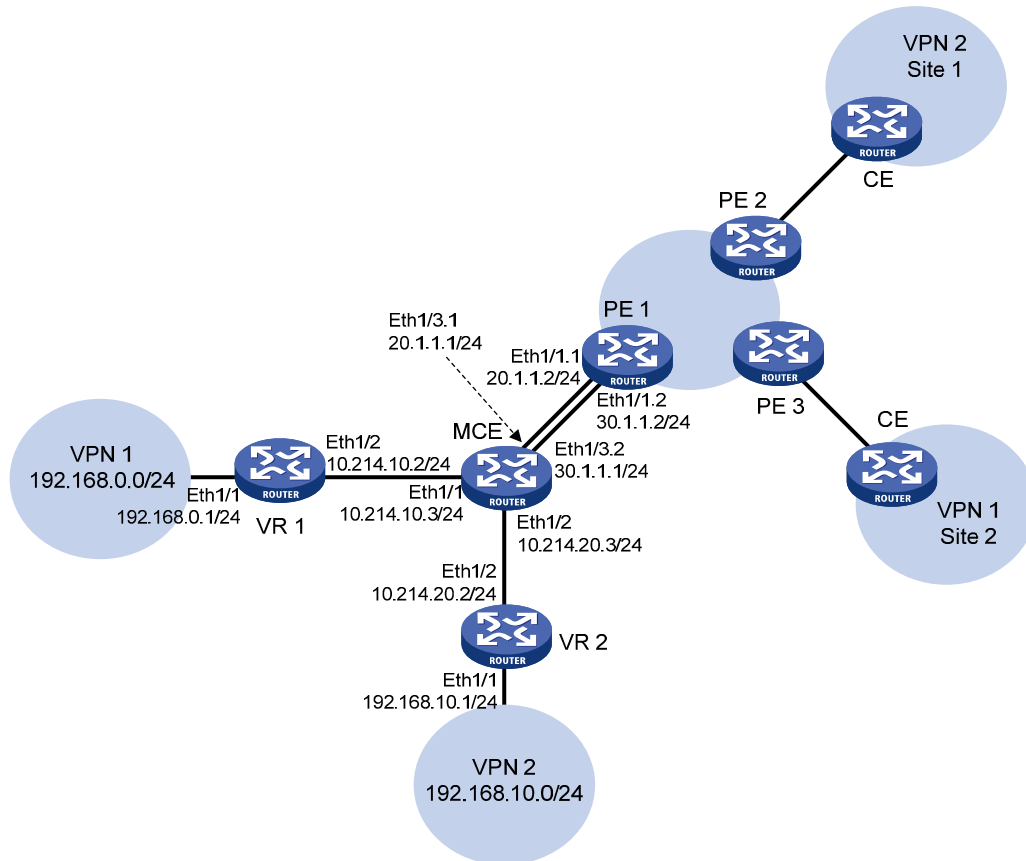
```
Request list: 0  Retransmit list: 0
```

Configuring MCE

Network requirements

As shown in [Figure 58](#), VPN 2 runs RIP. Configure the MCE device to separate routes from different VPNs and advertise the VPN routes to PE 1 through OSPF.

Figure 58 Network diagram



Configuration procedure

Assume that the system name of the MCE device is MCE, the system names of the edge routers of VPN 1 and VPN 2 are VR1 and VR2, and the system name of PE 1 is PE1.

1. Configure VPN instances on the MCE and PE 1:

On the MCE, configure VPN instances **vpn1** and **vpn2**, and specify an RD and route targets for each VPN instance.

```
<MCE> system-view
[MCE] ip vpn-instance vpn1
[MCE-vpn-instance-vpn1] route-distinguisher 10:1
[MCE-vpn-instance-vpn1] vpn-target 10:1
[MCE-vpn-instance-vpn1] quit
[MCE] ip vpn-instance vpn2
[MCE-vpn-instance-vpn2] route-distinguisher 20:1
[MCE-vpn-instance-vpn2] vpn-target 20:1
[MCE-vpn-instance-vpn2] quit
```

Bind interface Ethernet 1/1 with VPN instance **vpn1**, and configure an IP address for the interface.

```
[MCE] interface ethernet 1/1
[MCE-Ethernet1/1] ip binding vpn-instance vpn1
[MCE-Ethernet1/1] ip address 10.214.10.3 24
[MCE-Ethernet1/1] quit
```

Bind interface Ethernet 1/2 with VPN instance **vpn2**, and configure an IP address for the interface.

```
[MCE] interface ethernet 1/2
[MCE-Ethernet1/2] ip binding vpn-instance vpn2
[MCE-Ethernet1/2] ip address 10.214.20.3 24
[MCE-Ethernet1/2] quit
```

On PE 1, configure VPN instances **vpn1** and **vpn2**, and specify an RD and route targets for each VPN instance.

```
<PE1> system-view
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 30:1
[PE1-vpn-instance-vpn1] vpn-target 10:1
[PE1-vpn-instance-vpn1] quit
[PE1] ip vpn-instance vpn2
[PE1-vpn-instance-vpn2] route-distinguisher 40:1
[PE1-vpn-instance-vpn2] vpn-target 20:1
[PE1-vpn-instance-vpn2] quit
```

2. Configure routing between the MCE and VPN sites:

The MCE is connected to VPN 1 directly, and no routing protocol is enabled in VPN 1. Therefore, you can configure static routes.

On VR 1, assign IP address 10.214.10.2/24 to the interface connected to MCE and 192.168.0.1/24 to the interface connected to VPN 1. (Details not shown.)

On VR 1, configure a default route with the next hop as 10.214.10.3.

```
<VR1> system-view
[VR1] ip route-static 0.0.0.0 0.0.0.0 10.214.10.3
```

On the MCE, configure a static route to 192.168.0.0/24, specify the next hop as 10.214.10.2, and bind the static route with VPN instance **vpn1**.

```
[MCE] ip route-static vpn-instance vpn1 192.168.0.0 24 10.214.10.2
```

Run RIP in VPN 2. Configure RIP process 20 for the VPN instance **vpn2** on MCE, so that MCE can learn the routes of VPN 2 and add them to the routing table of the VPN instance **vpn2**.

```
[MCE] rip 20 vpn-instance vpn2
```

Advertise subnet 10.214.10.0.

```
[MCE-rip-20] network 10.214.20.0
```

```
[MCE-rip-20] quit
```

On VR 2, assign IP address 10.214.20.2/24 to the interface connected to the MCE and 192.168.10.1/24 to the interface connected to VPN 2. (Details not shown.)

Configure RIP, and advertise subnets 192.168.10.0 and 10.214.20.0.

```
<VR2> system-view
```

```
[VR2] rip 20
```

```
[VR2-rip-20] network 192.168.10.0
```

```
[VR2-rip-20] network 10.214.20.0
```

On MCE, display the routing tables of VPN instances **vpn1** and **vpn2**.

```
[MCE] display ip routing-table vpn-instance vpn1
```

```
Destinations : 13          Routes : 13
```

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
0.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
10.214.10.0/24	Direct	0	0	10.214.10.3	Eth1/1
10.214.10.0/32	Direct	0	0	10.214.10.3	Eth1/1
10.214.10.3/32	Direct	0	0	127.0.0.1	InLoop0
10.214.10.255/32	Direct	0	0	10.214.10.3	Eth1/1
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0
127.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0
192.168.0.0/24	Static	60	0	10.214.10.2	Eth1/1
224.0.0.0/4	Direct	0	0	0.0.0.0	NULL0
224.0.0.0/24	Direct	0	0	0.0.0.0	NULL0
255.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0

```
[MCE] display ip routing-table vpn-instance vpn2
```

```
Destinations : 13          Routes : 13
```

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
0.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
10.214.20.0/24	Direct	0	0	10.214.20.3	Eth1/2
10.214.20.0/32	Direct	0	0	10.214.20.3	Eth1/2
10.214.20.3/32	Direct	0	0	127.0.0.1	InLoop0
10.214.20.255/32	Direct	0	0	10.214.20.3	Eth1/2
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0
127.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0
192.168.10.0/24	RIP	100	1	10.214.20.2	Eth1/2
224.0.0.0/4	Direct	0	0	0.0.0.0	NULL0

```

224.0.0.0/24          Direct 0    0          0.0.0.0     NULL0
255.255.255.255/32  Direct 0    0          127.0.0.1   InLoop0

```

The output shows that the MCE has learned the private route of VPN 2 through RIP. MCE maintains the routes of VPN 1 and those of VPN 2 in two different routing tables. In this way, routes from different VPNs are separated.

3. Configure routing between the MCE and PE 1:

The MCE is connected to PE 1 through subinterfaces. On MCE, bind subinterface Ethernet 1/3.1 with the VPN instance **vpn1**, configure the subinterface to terminate VLAN 10, and configure an IP address for the subinterface.

```

[MCE] interface ethernet 1/3.1
[MCE-Ethernet1/3.1] ip binding vpn-instance vpn1
[MCE-Ethernet1/3.1] vlan-type dot1q vid 10
[MCE-Ethernet1/3.1] ip address 20.1.1.1 24
[MCE-Ethernet1/3.1] quit

```

On the MCE, bind subinterface Ethernet 1/3.2 with the VPN instance **vpn2**, configure the subinterface to terminate VLAN 20, and configure an IP address for the subinterface.

```

[MCE] interface ethernet 1/3.2
[MCE-Ethernet1/3.2] ip binding vpn-instance vpn2
[MCE-Ethernet1/3.2] vlan-type dot1q vid 20
[MCE-Ethernet1/3.2] ip address 30.1.1.1 24
[MCE-Ethernet1/3.2] quit

```

On PE 1, bind subinterface Ethernet 1/1.1 with the VPN instance **vpn1**, configure the subinterface to terminate VLAN 10, and configure an IP address for the subinterface.

```

[PE1] interface ethernet 1/1.1
[PE1-Ethernet1/1.1] ip binding vpn-instance vpn1
[PE1-Ethernet1/1.1] vlan-type dot1q vid 10
[PE1-Ethernet1/1.1] ip address 20.1.1.2 24
[PE1-Ethernet1/1.1] quit

```

On PE 1, bind subinterface Ethernet 1/1.2 with the VPN instance **vpn2**, configure the subinterface to terminate VLAN 20, and configure an IP address for the subinterface.

```

[PE1] interface ethernet 1/1.2
[PE1-Ethernet1/1.2] ip binding vpn-instance vpn2
[PE1-Ethernet1/1.2] vlan-type dot1q vid 20
[PE1-Ethernet1/1.2] ip address 30.1.1.2 24
[PE1-Ethernet1/1.2] quit

```

Configure the IP address of the interface Loopback0 as 101.101.10.1 for the MCE and as 100.100.10.1 for PE 1. Specify the loopback interface address as the router ID for the MCE and PE 1. (Details not shown.)

Enable OSPF process 10 on the MCE, bind the process to VPN instance **vpn1**, and set the domain ID to 10.

```

[MCE] ospf 10 router-id 101.101.10.1 vpn-instance vpn1
[MCE-ospf-10] vpn-instance-capability simple
[MCE-ospf-10] domain-id 10

```

Advertise subnet 20.1.1.0/24 in area 0, and redistribute the static route of VPN 1.

```

[MCE-ospf-10] area 0
[MCE-ospf-10-area-0.0.0.0] network 20.1.1.0 0.0.0.255
[MCE-ospf-10-area-0.0.0.0] quit

```

```
[MCE-ospf-10] import-route static
# On PE 1, enable OSPF process 10, bind the process to VPN instance vpn1, set the domain ID
to 10, and advertise subnet 20.1.1.0/24 in area 0.
[PE1] ospf 10 router-id 100.100.10.1 vpn-instance vpn1
[PE1-ospf-10] domain-id 10
[PE1-ospf-10] area 0
[PE1-ospf-10-area-0.0.0.0] network 20.1.1.0 0.0.0.255
[PE1-ospf-10-area-0.0.0.0] quit
[PE1-ospf-10] quit
# Configure OSPF process 20 between MCE and PE 1, and redistribute routes from RIP process 20
into OSPF. (Details not shown.)
```

Verifying the configuration

On PE 1, display the routing information for VPN 1. The output shows that the static route of VPN 1 has been redistributed to the OSPF routing table of PE 1.

```
[PE1] display ip routing-table vpn-instance vpn1
```

```
Destinations : 13          Routes : 13
```

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
0.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
20.1.1.0/24	Direct	0	0	20.1.1.2	Eth1/1.1
20.1.1.0/32	Direct	0	0	20.1.1.2	Eth1/1.1
20.1.1.2/32	Direct	0	0	127.0.0.1	InLoop0
20.1.1.255/32	Direct	0	0	20.1.1.2	Eth1/1.1
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0
127.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0
192.168.0.0/24	O_ASE	150	1	20.1.1.1	Eth1/1.1
224.0.0.0/4	Direct	0	0	0.0.0.0	NULL0
224.0.0.0/24	Direct	0	0	0.0.0.0	NULL0
255.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0

On PE 1, display the routing information for VPN 2. The output shows that the RIP route of VPN 2 has been redistributed to the OSPF routing table of PE 1.

```
[PE1] display ip routing-table vpn-instance vpn2
```

```
Destinations : 13          Routes : 13
```

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
0.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
30.1.1.0/24	Direct	0	0	30.1.1.2	Eth1/1.2
30.1.1.0/32	Direct	0	0	30.1.1.2	Eth1/1.2
30.1.1.2/32	Direct	0	0	127.0.0.1	InLoop0
30.1.1.255/32	Direct	0	0	30.1.1.2	Eth1/1.2
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0

127.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0
192.168.10.0/24	O_ASE	150	1	30.1.1.1	Eth1/1.2
224.0.0.0/4	Direct	0	0	0.0.0.0	NULL0
224.0.0.0/24	Direct	0	0	0.0.0.0	NULL0
255.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0

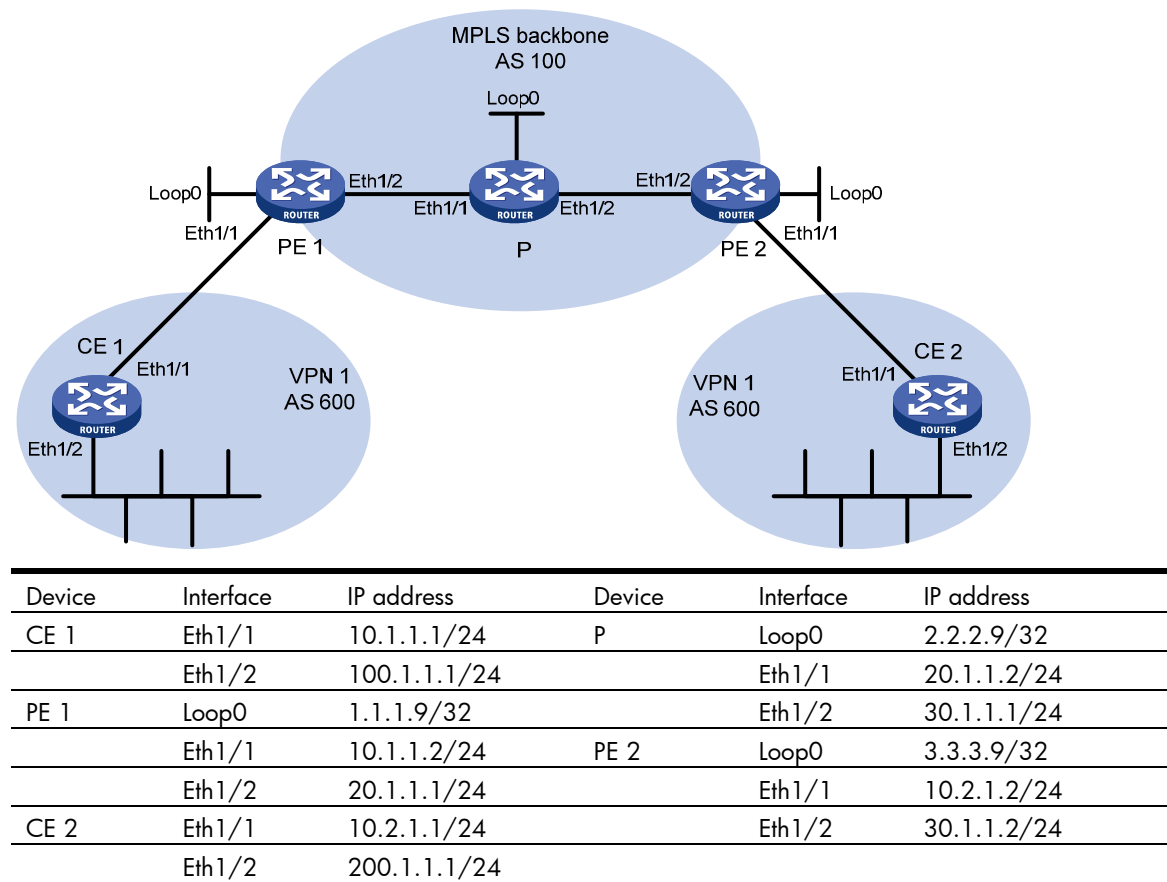
Now, the routing information for the two VPNs has been redistributed into the routing tables on PE 1.

Configuring BGP AS number substitution

Network requirements

As shown in Figure 59, CE 1 and CE 2 belong to VPN 1 and are connected to PE 1 and PE 2. The two CEs have the same AS number, 600. Configure BGP AS number substitution on the PEs to enable the CEs to communicate with each other.

Figure 59 Network diagram



Configuration procedure

1. Configure basic MPLS L3VPN:
 - Configure OSPF on the MPLS backbone to allow the PEs and P device to learn the routes of the loopback interfaces from each other.
 - Configure basic MPLS and MPLS LDP on the MPLS backbone to establish LDP LSPs.
 - Establish MP-IBGP peer relationship between the PEs to advertise VPN IPv4 routes.
 - Configure the VPN instance of VPN 1 on PE 2 to allow CE 2 to access the network.

- Configure the VPN instance of VPN 1 on PE 1 to allow CE 1 to access the network.
- Configure BGP between PE 1 and CE 1, and between PE 2 and CE 2, and redistribute routes of CEs into PEs.

After completing the configurations, execute the **display ip routing-table** command on CE 2. The output shows that CE 2 has learned the route to network 10.1.1.0/24, where the interface used by CE 1 to access PE 1 resides, but it has not learned the route to the VPN (100.1.1.0/24) behind CE 1. The situation on CE 1 is similar.

```
<CE2> display ip routing-table
```

```
Destinations : 17          Routes : 17
```

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
0.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
10.1.1.0/24	BGP	255	0	10.2.1.2	Eth1/1
10.2.1.0/24	Direct	0	0	10.2.1.1	Eth1/1
10.2.1.0/32	Direct	0	0	10.2.1.1	Eth1/1
10.2.1.1/32	Direct	0	0	127.0.0.1	InLoop0
10.2.1.255/32	Direct	0	0	10.2.1.1	Eth1/1
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0
127.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0
200.1.1.0/24	Direct	0	0	200.1.1.1	Eth1/2
200.1.1.0/32	Direct	0	0	200.1.1.1	Eth1/2
200.1.1.1/32	Direct	0	0	127.0.0.1	InLoop0
200.1.1.255/24	Direct	0	0	200.1.1.1	Eth1/2
224.0.0.0/4	Direct	0	0	0.0.0.0	NULL0
224.0.0.0/24	Direct	0	0	0.0.0.0	NULL0
255.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0

Execute the **display ip routing-table vpn-instance** command on the PEs. You can see the route to the VPN behind the peer CE. Take PE 2 as an example:

```
<PE2> display ip routing-table vpn-instance vpn1
```

```
Destinations : 15          Routes : 15
```

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
0.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
10.1.1.0/24	BGP	255	0	1.1.1.9	Eth1/2
10.2.1.0/24	Direct	0	0	10.2.1.2	Eth1/1
10.2.1.0/32	Direct	0	0	10.2.1.2	Eth1/1
10.2.1.2/32	Direct	0	0	127.0.0.1	InLoop0
10.2.1.255/32	Direct	0	0	10.2.1.2	Eth1/1
100.1.1.0/24	BGP	255	0	1.1.1.9	Eth1/2
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0
127.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0
200.1.1.0/24	BGP	255	0	10.2.1.1	Eth1/1

```

224.0.0.0/4          Direct 0    0          0.0.0.0      NULL0
224.0.0.0/24        Direct 0    0          0.0.0.0      NULL0
255.255.255.255/32 Direct 0    0          127.0.0.1    InLoop0

```

Enabling BGP update packet debugging on PE 2, you can see that PE 2 advertises the route to 100.1.1.0/24, and the AS_PATH is 100 600.

```

<PE2> terminal monitor
<PE2> terminal logging level 7
<PE2> debugging bgp update vpn-instance vpn1 10.2.1.1 ipv4
<PE2> refresh bgp all export ipv4 vpn-instance vpn1
*Jun 13 16:12:52:096 2012 PE2 BGP/7/DEBUG: -MDC=1;
      BGP.vpn1: Send UPDATE to peer 10.2.1.1 for following destinations:
      Origin      : Incomplete
      AS Path     : 100 600
      Next Hop    : 10.2.1.2
                  100.1.1.0/24,

```

Execute the **display bgp routing-table ipv4 peer received-routes** command on CE 2. The output shows that CE 2 has not received the route to 100.1.1.0/24.

```
<CE2> display bgp routing-table ipv4 peer 10.2.1.2 received-routes
```

```
Total number of routes: 2
```

```
BGP local router ID is 200.1.1.1
```

```
Status codes: * - valid, > - best, d - dampened, h - history,
              s - suppressed, S - stale, i - internal, e - external
Origin: i - IGP, e - EGP, ? - incomplete
```

Network	NextHop	MED	LocPrf	PrefVal	Path/Ogn
* >e 10.1.1.0/24	10.2.1.2			0	100?
* e 10.2.1.0/24	10.2.1.2	0		0	100?

2. Configure BGP AS number substitution on PE 2:

```

<PE2> system-view
[PE2] bgp 100
[PE2-bgp] ip vpn-instance vpn1
[PE2-bgp-vpn1] peer 10.2.1.1 substitute-as
[PE2-bgp-vpn1] address-family ipv4 unicast
[PE2-bgp-ipv4-vpn1] peer 10.2.1.1 enable
[PE2-bgp-ipv4-vpn1] quit
[PE2-bgp-vpn1] quit
[PE2-bgp] quit

```

Verifying the configuration

The output shows that among the routes advertised by PE 2 to CE 2, the AS_PATH of 100.1.1.0/24 has changed from 100 600 to 100 100:

```

*Jun 13 16:15:59:456 2012 PE2 BGP/7/DEBUG: -MDC=1;
      BGP.vpn1: Send UPDATE to peer 10.2.1.1 for following destinations:
      Origin      : Incomplete
      AS Path     : 100 100

```

```
Next Hop      : 10.2.1.2
100.1.1.0/24,
```

Display again the routing information that CE 2 has received, and the routing table:

```
<CE2> display bgp routing-table ipv4 peer 10.2.1.2 received-routes
```

```
Total number of routes: 3
```

```
BGP local router ID is 200.1.1.1
```

```
Status codes: * - valid, > - best, d - dampened, h - history,
s - suppressed, S - stale, i - internal, e - external
Origin: i - IGP, e - EGP, ? - incomplete
```

Network	NextHop	MED	LocPrf	PrefVal	Path/Ogn
* >e 10.1.1.0/24	10.2.1.2			0	100?
* e 10.2.1.0/24	10.2.1.2	0		0	100?
* >e 100.1.1.0/24	10.2.1.2			0	100 100?

```
<CE2> display ip routing-table
```

```
Destinations : 18          Routes : 18
```

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
0.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
10.1.1.0/24	BGP	255	0	10.2.1.2	Eth1/1
10.2.1.0/24	Direct	0	0	10.2.1.1	Eth1/1
10.2.1.0/32	Direct	0	0	10.2.1.1	Eth1/1
10.2.1.1/32	Direct	0	0	127.0.0.1	InLoop0
10.2.1.255/32	Direct	0	0	10.2.1.1	Eth1/1
100.1.1.0/24	BGP	255	0	10.2.1.2	Eth1/1
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.0/32	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0
127.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0
200.1.1.0/24	Direct	0	0	200.1.1.1	Eth1/2
200.1.1.0/32	Direct	0	0	200.1.1.1	Eth1/2
200.1.1.1/32	Direct	0	0	127.0.0.1	InLoop0
200.1.1.255/32	Direct	0	0	200.1.1.1	Eth1/2
224.0.0.0/4	Direct	0	0	0.0.0.0	NULL0
224.0.0.0/24	Direct	0	0	0.0.0.0	NULL0
255.255.255.255/32	Direct	0	0	127.0.0.1	InLoop0

After you also configure BGP AS substitution on PE 1, the Ethernet interfaces of CE 1 and CE 2 can ping each other.

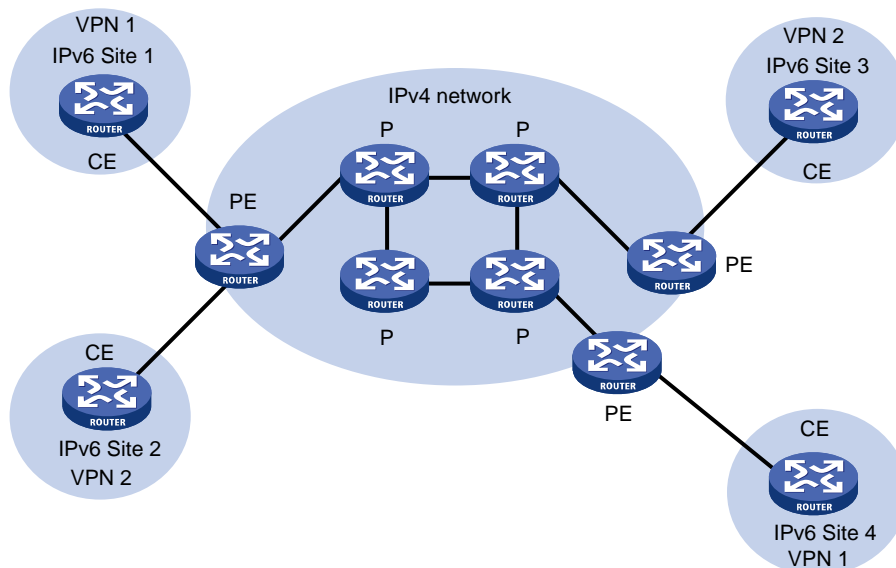
Configuring IPv6 MPLS L3VPN

Overview

IPv6 MPLS L3VPN uses BGP to advertise IPv6 VPN routes and uses MPLS to forward IPv6 VPN packets on the service provider backbone.

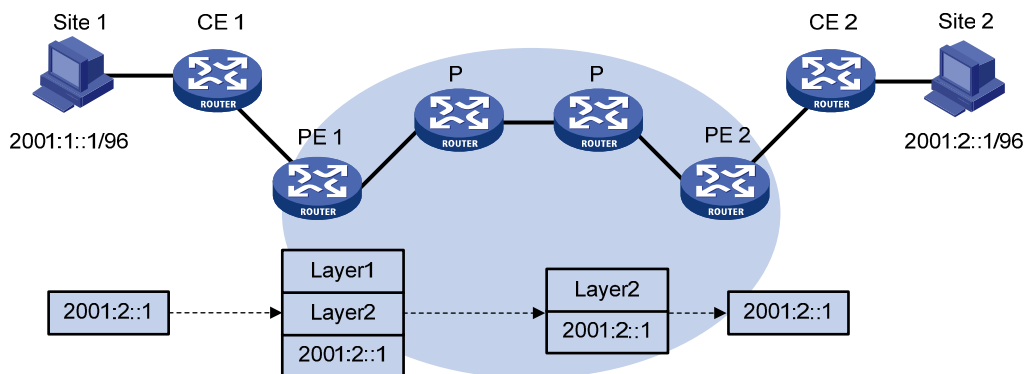
Figure 60 shows a typical IPv6 MPLS L3VPN model. The service provider backbone in the IPv6 MPLS L3VPN model is an IPv4 network. IPv6 runs inside the VPNs and between CE and PE. Therefore, PEs must support both IPv4 and IPv6. The PE-CE interfaces of a PE run IPv6, and the PE-P interface of a PE runs IPv4.

Figure 60 Network diagram for the IPv6 MPLS L3VPN model



IPv6 MPLS L3VPN packet forwarding

Figure 61 IPv6 MPLS L3VPN packet forwarding diagram



As shown in [Figure 61](#), the IPv6 MPLS L3VPN packet forwarding procedure is as follows:

1. The PC at Site 1 sends an IPv6 packet destined for 2001:2::1, the PC at Site 2. CE 1 transmits the packet to PE 1.
2. Based on the inbound interface and destination address of the packet, PE 1 finds a matching entry from the routing table of the VPN instance, labels the packet with both inner and outer labels, and forwards the packet out.
3. The MPLS backbone transmits the packet to PE 2 by outer label. The outer label is removed from the packet at the penultimate hop.
4. According to the inner label and destination address of the packet, PE 2 searches the routing table of the VPN instance to determine the outbound interface, and then forwards the packet out of the interface to CE 2.
5. CE 2 forwards the packet to the destination by IPv6 forwarding.

IPv6 MPLS L3VPN routing information advertisement

The routing information for a local CE is advertised to the remote CE using the following process:

1. From the local CE to the ingress PE.
The local CE advertises standard IPv6 routing information to the ingress PE over an IPv6 static route, RIPng route, OSPFv3 route, IPv6 IS-IS route, IBGP route, or EBGp route.
2. From the ingress PE to the egress PE.
After receiving the standard IPv6 routes from the CE, the ingress PE adds RDs and route targets to create VPN-IPv6 routes, saves the routes to the routing table of the VPN instance created for the CE, and then notifies MPLS to assign VPN labels for the routes.
Then, the ingress PE advertises the VPN-IPv6 routes to the egress PE through MP-BGP.
The egress PE compares the export target attributes of the VPN-IPv6 routes with the import target attributes that it maintains for the VPN instance and, if they are the same, adds the routes to the routing table of the VPN instance.
The PEs use an IGP to ensure the connectivity between them.
3. From the egress PE to the remote peer CE.
The egress PE restores the original IPv6 routes and advertises them to the remote CE over an IPv6 static route, RIPng route, OSPFv3 route, IPv6 IS-IS route, EBGp, or IBGP route.

IPv6 MPLS L3VPN network schemes and functions

IPv6 MPLS L3VPN supports the following network schemes and functions:

- Basic VPN
- Inter-AS VPN option A
- Inter-AS VPN option C
- Carrier's carrier
- Multi-VPN-instance CE

IPv6 MPLS L3VPN configuration task list

By configuring basic IPv6 MPLS L3VPN, you can construct a simple IPv6 VPN network over an MPLS backbone.

To deploy special IPv6 MPLS L3VPN networks, such as inter-AS VPN, you must also perform specific configurations in addition to the basic IPv6 MPLS L3VPN configuration. For details, see the related sections.

Tasks at a glance

[Configuring basic IPv6 MPLS L3VPN](#)

[Configuring inter-AS IPv6 VPN](#)

[Configuring routing on an MCE](#)

Configuring basic IPv6 MPLS L3VPN

The key task in IPv6 MPLS L3VPN configuration is to manage the advertisement of IPv6 VPN routes on the MPLS backbone, including management of PE-CE route exchange and PE-PE route exchange.

To configure basic IPv6 MPLS L3VPN:

Tasks at a glance

Configuring VPN instances:

1. (Required.) [Creating a VPN instance](#)
2. (Required.) [Associating a VPN instance with an interface](#)
3. (Optional.) [Configuring route related attributes for a VPN instance](#)

(Required.) [Configuring routing between a PE and a CE](#)

(Required.) [Configuring routing between PEs](#)

(Optional.) [Configuring BGP VPNv6 route control](#)

Before configuring basic IPv6 MPLS L3VPN, complete the following tasks:

- Configure an IGP on the PEs and Ps to ensure IP connectivity within the MPLS backbone.
- Configure basic MPLS for the MPLS backbone.
- Configure MPLS LDP on PEs and Ps to establish LDP LSPs.

Configuring VPN instances

By configuring VPN instances on a PE, you isolate not only VPN routes from public network routes, but also routes between VPNs. This feature allows VPN instances to be used in network scenarios besides MPLS L3VPNs.

All VPN instance configurations are performed on PEs or MCEs.

Creating a VPN instance

A VPN instance is associated with a site. It is a collection of the VPN membership and routing rules of its associated site. A VPN instance does not necessarily correspond to one VPN.

To create and configure a VPN instance:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a VPN instance and enter VPN instance view.	ip vpn-instance <i>vpn-instance-name</i>	By default, no VPN instance is created. You can configure a maximum of 1023 VPN instances on a PE.
3. Configure an RD for the VPN instance.	route-distinguisher <i>route-distinguisher</i>	By default, no RD is specified.
4. (Optional.) Configure a description for the VPN instance.	description <i>text</i>	By default, no description is configured for a VPN instance. The description should contain the VPN instance's related information, such as its relationship with a certain VPN.
5. (Optional.) Configure an ID for the VPN instance.	vpn-id <i>vpn-id</i>	By default, no ID is configured for a VPN instance.

Associating a VPN instance with an interface

After creating and configuring a VPN instance, associate the VPN instance with the interface connected to the CE.

To associate a VPN instance with an interface:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
3. Associate a VPN instance with the interface.	ip binding vpn-instance <i>vpn-instance-name</i>	By default, no VPN instance is associated with an interface. The ip binding vpn-instance command clears the IP address of the interface. Therefore, re-configure an IP address for the interface after configuring this command.

Configuring route related attributes for a VPN instance

VPN routes are controlled and advertised on a PE using the following process:

- When a VPN route learned from a CE gets redistributed into BGP, BGP associates it with a route target extended community attribute list, which is usually the export target attribute of the VPN instance associated with the CE.

- The VPN instance determines which routes it can accept and redistribute according to the **import-extcommunity** in the route target.
- The VPN instance determines how to change the route target attributes for routes to be advertised according to the **export-extcommunity** in the route target.

To configure route related attributes for a VPN instance:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter VPN instance view or IPv6 VPN view.	<ul style="list-style-type: none"> • Enter VPN instance view: ip vpn-instance <i>vpn-instance-name</i> • Enter IPv6 VPN view: address-family ipv6 	<p>Configurations made in VPN instance view apply to both IPv4 VPN and IPv6 VPN.</p> <p>IPv6 VPN prefers the configurations in IPv6 VPN view over the configurations in VPN instance view.</p>
3. Configure route targets.	vpn-target <i>vpn-target</i> &<1-8> [both export-extcommunity import-extcommunity]	By default, no route targets are configured.
4. Set the maximum number of routes supported.	routing-table limit <i>number</i> { <i>warn-threshold</i> simply-alert }	<p>The default setting depends on the device model. For more information, see the command in <i>MPLS Command Reference</i>.</p> <p>Setting the maximum number of routes for a VPN instance can prevent the PE from storing too many routes.</p>
5. Apply an import routing policy.	import route-policy <i>route-policy</i>	<p>By default, all routes matching the import target attribute are accepted.</p> <p>Make sure the routing policy already exists. Otherwise, the device does not filter received routes.</p> <p>For information about routing policies, see <i>Layer 3—IP Routing Configuration Guide</i>.</p>
6. Apply an export routing policy.	export route-policy <i>route-policy</i>	<p>By default, routes to be advertised are not filtered.</p> <p>Make sure the routing policy already exists. Otherwise, the device does not filter routes to be advertised.</p> <p>For information about routing policies, see <i>Layer 3—IP Routing Configuration Guide</i>.</p>

Step	Command	Remarks
7. Apply a tunnel policy to the VPN instance.	tnl-policy <i>tunnel-policy-name</i>	By default, only one tunnel is selected (no load balancing) in this order: LSP tunnel, GRE tunnel, and CR-LSP tunnel. The specified tunnel policy must have been created. For information about tunnel policies, see <i>Configuring tunnel policies</i> .

Configuring routing between a PE and a CE

You can configure IPv6 static routing, RIPng, OSPFv3, IPv6 IS-IS, EBGp, or IBGP between a PE and a CE.

Configuring IPv6 static routing between a PE and a CE

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure an IPv6 static route for a VPN instance.	ipv6 route-static vpn-instance s-vpn-instance-name ipv6-address prefix-length { interface-type interface-number [next-hop-address] nexthop-address [public] vpn-instance d-vpn-instance-name nexthop-address } [permanent] [preference preference-value] [tag tag-value] [description description-text]	By default, no IPv6 static route is configured for a VPN instance. Perform this configuration on the PE. On the CE, configure a common IPv6 static route. For more information about IPv6 static routing, see <i>Layer 3—IP Routing Configuration Guide</i> .

Configuring RIPng between a PE and a CE

A RIPng process belongs to the public network or a single VPN instance. If you create a RIPng process without binding it to a VPN instance, the process belongs to the public network.

For more information about RIPng, see *Layer 3—IP Routing Configuration Guide*.

To configure RIPng between a PE and a CE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a RIPng process for a VPN instance and enter RIPng view.	ripng [<i>process-id</i>] vpn-instance <i>vpn-instance-name</i>	Perform this configuration on the PE. On the CE, create a common RIPng process.
3. Return to system view.	quit	N/A
4. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A

Step	Command	Remarks
5. Enable RIPng on the interface.	ripng <i>process-id</i> enable	By default, RIPng is disabled on an interface.

Configuring OSPFv3 between a PE and a CE

An OSPFv3 process belongs to the public network or a single VPN instance. If you create an OSPF process without binding it to a VPN instance, the process belongs to the public network.

For more information about OSPFv3, see *Layer 3—IP Routing Configuration Guide*.

To configure OSPFv3 between a PE and a CE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create an OSPFv3 process for a VPN instance and enter OSPFv3 view.	ospfv3 [<i>process-id</i>] vpn-instance <i>vpn-instance-name</i>	Perform this configuration on the PE. On the CE, create a common OSPF process. The maximum number of OSPF processes that a VPN instance can run depends on the device's memory. Deleting a VPN instance also deletes all related OSPFv3 processes.
3. Set the router ID.	router-id <i>router-id</i>	N/A
4. Return to system view.	quit	N/A
5. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
6. Enable OSPFv3 on the interface.	ospfv3 <i>process-id</i> area <i>area-id</i> [instance <i>instance-id</i>]	By default, OSPFv3 is disabled on an interface. Perform this configuration on the PE.

Configuring IPv6 IS-IS between a PE and a CE

An IPv6 IS-IS process belongs to the public network or a single VPN instance. If you create an IPv6 IS-IS process without binding it to a VPN instance, the process belongs to the public network.

For more information about IPv6 IS-IS, see *Layer 3—IP Routing Configuration Guide*.

To configure IPv6 IS-IS between a PE and a CE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create an IPv6 IS-IS process for a VPN instance and enter IS-IS view.	isis [<i>process-id</i>] vpn-instance <i>vpn-instance-name</i>	Perform this configuration on the PE. On the CE, create a common IPv6 IS-IS process.
3. Configure a network entity title for the IS-IS process.	network-entity <i>net</i>	By default, no NET is configured.

Step	Command	Remarks	
4.	Enable IPv6 for the IS-IS process.	ipv6 enable	IPv6 is disabled by default.
5.	Return to system view.	quit	N/A
6.	Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
7.	Enable IPv6 for the IS-IS process on the interface.	isis ipv6 enable [<i>process-id</i>]	IPv6 is disabled on an interface by default.

Configuring EBGW between a PE and a CE

1. Configure the PE:

Step	Command	Remarks	
1.	Enter system view.	system-view	N/A
2.	Enable BGP and enter BGP view.	bgp <i>as-number</i>	N/A
3.	Enter BGP-VPN view.	ip vpn-instance <i>vpn-instance-name</i>	N/A
4.	Configure the CE as the VPN EBGW peer.	peer { <i>group-name</i> <i>ipv6-address</i> } as-number <i>as-number</i>	By default, no BGP peer is configured.
5.	Create and enter BGP-VPN IPv6 unicast address family view.	address-family ipv6 [unicast]	Configuration commands in BGP-VPN IPv6 unicast address family view are the same as those in BGP IPv6 unicast address family view. For details, see <i>Layer 3—IP Routing Configuration Guide</i> .
6.	Enable IPv6 unicast route exchange with the specified peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange IPv6 unicast routes with any peer.
7.	Redistribute the routes of the local CE.	import-route <i>protocol</i> [<i>process-id</i>] [med <i>med-value</i> route-policy <i>route-policy-name</i>] *]	A PE must redistribute the routes of the local CE into its VPN routing table so that it can advertise them to the peer PE.
8.	(Optional.) Configure filtering of advertised routes.	filter-policy { <i>acl6-number</i> prefix-list <i>ipv6-prefix-name</i> } export [<i>protocol</i> <i>process-id</i>]	By default, BGP does not filter advertised routes.
9.	(Optional.) Configure filtering of received routes.	filter-policy { <i>acl6-number</i> prefix-list <i>ipv6-prefix-name</i> } import	By default, the PE does not filter received routes.

2. Configure the CE:

Step	Command	Remarks	
1.	Enter system view.	system-view	N/A
2.	Enter BGP view.	bgp <i>as-number</i>	N/A
3.	Configure the PE as an EBGW peer.	peer { <i>group-name</i> <i>ipv6-address</i> } as-number <i>as-number</i>	By default, no BGP peer is configured.

Step	Command	Remarks
4. Create and enter BGP IPv6 unicast address family view.	address-family ipv6 [unicast]	N/A
5. Enable IPv6 unicast route exchange with the specified peer or peer group.	peer { group-name ip-address } enable	By default, BGP does not exchange IPv6 unicast routes with any peer.
6. (Optional.) Configure route redistribution.	import-route protocol [process-id [med med-value route-policy route-policy-name] *]	A CE must advertise its VPN routes to the connected PE so that the PE can advertise them to the peer CE.

Configuring IBGP between a PE and a CE

Use IBGP between PE and CE only in a basic IPv6 MPLS L3VPN network. In networks such as inter-AS VPN and carrier's carrier, you cannot configure IBGP between PE and CE.

1. Configure the PE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp as-number	N/A
3. Enter BGP VPN view.	ip vpn-instance <i>vpn-instance-name</i>	Configuration commands in BGP VPN view are the same as those in BGP view. For details, see <i>Layer 3—IP Routing Configuration Guide</i> .
4. Configure the CE as the VPN IBGP peer.	peer { group-name ipv6-address } as-number <i>as-number</i>	By default, no BGP peer is created.
5. Create and enter BGP VPN IPv6 unicast family view.	address-family ipv6 [unicast]	N/A
6. Enable IPv6 unicast route exchange with the specified peer.	peer { group-name ipv6-address } enable	By default, BGP does not exchange IPv6 unicast routes with any peer.
7. Configure the CE as a client of the RR.	peer { group-name ipv6-address } reflect-client	By default, no RR or RR client is configured, and the PE does not advertise routes learned from the IBGP peer CE to other IBGP peers, including VPNv6 IBGP peers. The PE advertises routes learned from the CE to other IBGP peers only when you configure the IBGP peer CE as a client of the RR. Configuring an RR does not change the next hop of a route. To change the next hop of a route, configure an inbound policy on the receiving side.

Step	Command	Remarks
8. (Optional.) Enable route reflection between clients.	reflect between-clients	By default, route reflection between clients is enabled.
9. (Optional.) Configure the cluster ID for the RR.	reflector cluster-id { <i>cluster-id</i> <i>ip-address</i> }	By default, the RR uses its own router ID as the cluster ID. If multiple RRs exist in a cluster, use this command to configure the same cluster ID for all RRs in the cluster to avoid routing loops.

2. Configure the CE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Configure the PE as an IBGP peer.	peer { <i>group-name</i> <i>ipv6-address</i> } as-number <i>as-number</i>	By default, no BGP peer is created.
4. Create and enter BGP IPv6 unicast family view.	address-family ipv6 [unicast]	N/A
5. Enable IPv6 unicast route exchange with the specified peer or peer group.	peer { <i>group-name</i> <i>ipv6-address</i> } enable	By default, BGP does not exchange IPv6 unicast routes with any peer.
6. (Optional.) Configure route redistribution.	import-route <i>protocol</i> [<i>process-id</i> [med <i>med-value</i> route-policy <i>route-policy-name</i>] *]	A CE must redistribute its routes to the PE so the PE can advertise them to the peer CE.

Configuring routing between PEs

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Configure the remote PE as the peer.	peer { <i>group-name</i> <i>ipv6-address</i> } as-number <i>as-number</i>	By default, no BGP peer is configured.
4. Specify the source interface for route update packets sent to the specified peer.	peer { <i>group-name</i> <i>ip-address</i> } connect-interface <i>interface-type</i> <i>interface-number</i>	By default, BGP uses the outbound interface of the best route destined to the BGP peer as the source interface.
5. Enter BGP-VPNv6 address family view.	address-family vpnv6	N/A
6. Enable BGP-VPNv6 route exchange with the specified peer.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange BGP-VPNv6 routes with any peer.

Configuring BGP VPNv6 route control

BGP VPNv6 route control is configured similarly with BGP route control, except that it is configured in BGP-VPNv6 address family view. For detailed information about BGP route control, see *Layer 3—IP Routing Configuration Guide*.

To configure BGP VPNv6 route control:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp as-number	N/A
3. Enter BGP-VPNv6 address family view.	address-family vpnv6	N/A
4. (Optional.) Configure filtering of advertised routes.	filter-policy { <i>acl6-number</i> prefix-list <i>ipv6-prefix-name</i> } export [<i>protocol process-id</i>]	By default, the PE does not filter advertised routes.
5. (Optional.) Configure filtering of received routes.	filter-policy { <i>acl6-number</i> prefix-list <i>ipv6-prefix-name</i> } import	By default, the PE does not filter received routes.
6. Configure ACL-based route filtering for the specified peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } filter-policy <i>acl6-number</i> { export import }	By default, no ACL-based route filtering is configured.
7. Configure IPv6 prefix list-based route filtering for the specified peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } prefix-list <i>ipv6-prefix-name</i> { export import }	By default, no IPv6 prefix list-based route filtering is configured.
8. Specify a preferred value for routes received from the peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } preferred-value <i>value</i>	The default preferred value is 0.
9. Configure BGP updates sent to the peer to carry only public AS numbers.	peer { <i>group-name</i> <i>ip-address</i> } public-as-only	By default, a BGP update carries both public and private AS numbers.
10. Apply a routing policy to routes advertised to or received from the peer or peer group.	peer { <i>group-name</i> <i>ip-address</i> } route-policy <i>route-policy-name</i> { export import }	By default, no routing policy is applied for a peer.
11. Enable route target filtering for received BGP-VPNv6 routes.	policy vpn-target	By default, route target filtering is enabled.
12. Configure the local PE as the route reflector and specify the peer as the client.	peer { <i>group-name</i> <i>ip-address</i> } reflect-client	By default, no route reflector or client is configured.
13. Enable route reflection between clients.	reflect between-clients	By default, route reflection between clients is enabled.

Step	Command	Remarks
14. Configure a cluster ID for the route reflector.	reflector cluster-id { <i>cluster-id</i> <i>ip-address</i> }	By default, an RR uses its own router ID as the cluster ID. If more than one RR exists in a cluster, use this command to configure the same cluster ID for all RRs in the cluster to avoid routing loops.
15. Configure filtering of reflected routes.	rr-filter <i>extended-community-list-number</i>	By default, an RR does not filter reflected routes. Only IBGP routes whose extended community attribute matches the specified community list are reflected. By configuring different filtering policies on RRs, you can implement load balancing among the RRs.

Configuring inter-AS IPv6 VPN

If the MPLS backbone spans multiple ASs, you must configure inter-AS IPv6 VPN.

There are three inter-AS VPN solutions (for more information, see "Configuring MPLS L3VPN"). IPv6 MPLS L3VPN supports only inter-AS VPN option A and option C.

Before configuring inter-AS IPv6 VPN, complete these tasks:

- Configure an IGP for the MPLS backbone in each AS to ensure IP connectivity.
- Configure basic MPLS for the MPLS backbone of each AS.
- Configure MPLS LDP for the MPLS backbones so that LDP LSPs can be established.

The following sections describe inter-AS IPv6 VPN option A and option C. Select one according to your network scenario.

Configuring inter-AS IPv6 VPN option A

Inter-AS IPv6 VPN option A applies to scenarios where the number of VPNs and that of VPN routes on the PEs are relatively small.

To configure inter-AS IPv6 option A:

- Configure basic IPv6 MPLS L3VPN on each AS.
- Configure VPN instances on both PEs and ASBR PEs. The VPN instances on PEs allow CEs to access the network, and those on ASBR PEs are for access of the peer ASBR PEs.

For more configuration information, see "Configuring MPLS L3VPN."

In the inter-AS IPv6 VPN option A solution, for the same IPv6 VPN, the route targets configured on the PEs must match those configured on the ASBR-PEs in the same AS to make sure VPN routes sent by the PEs (or ASBR-PEs) can be received by the ASBR-PEs (or PEs). Route targets configured on the PEs in different ASs do not have such requirements.

Configuring inter-AS IPv6 VPN option C

To configure inter-AS IPv6 VPN option C, perform proper configurations on PEs and ASBR PEs, and configure routing policies on the ASBR PEs.

Configuring the PEs

Establish an IBGP peer relationship between a PE and an ASBR PE in an AS, and an MP-EBGP peer relationship between PEs in different ASs.

The PEs and ASBR PEs in an AS must be able to exchange labeled routes.

To configure a PE for inter-AS IPv6 VPN option C:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Configure the ASBR PE in the same AS as an IBGP peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	By default, no BGP peer is configured.
4. Enter BGP IPv4 unicast address family view.	address-family ipv4 [unicast]	N/A
5. Enable the PE to exchange BGP IPv4 unicast routes with the ASBR PE in the same AS.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, the PE does not exchange BGP IPv4 unicast routes with any peer.
6. Enable the PE to exchange labeled routes with the ASBR PE in the same AS.	peer { <i>group-name</i> <i>ip-address</i> } label-route-capability	By default, the PE does not advertise labeled routes to any IPv4 peer/peer group.
7. Return to BGP view.	quit	N/A
8. Configure the PE of another AS as the EBGP peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	N/A
9. Enter BGP-VPNv6 address family view.	address-family vpnv6	N/A
10. Enable the PE to exchange BGP VPNv6 routing information with the EBGP peer.	peer <i>ip-address</i> enable	By default, the PE does not exchange labeled routes with any IPv4 peer/peer group.

Configuring the ASBR PEs

In the inter-AS IPv6 VPN option C solution, an inter-AS LSP is needed, and the routes advertised between the PEs and ASBRs must carry MPLS label information. The configuration is the same as that in the Inter-AS IPv4 VPN option C solution. For more information, see "Configuring MPLS L3VPN."

Configuring the routing policy

A routing policy on an ASBR PE does the following:

- Assigns MPLS labels to routes received from the PEs in the same AS before advertising them to the peer ASBR PE.
- Assigns new MPLS labels to the labeled routes to be advertised to the PEs in the same AS.

The configuration is the same as that in the Inter-AS IPv4 VPN option C solution. For more information, see "Configuring MPLS L3VPN."

Configuring routing on an MCE

An MCE implements service isolation through route isolation. MCE routing configuration includes the following:

- MCE-VPN site routing configuration
- MCE-PE routing configuration

On a PE in an MCE network environment, disable routing loop detection to avoid route loss during route calculation, and disable route redistribution between routing protocols to save system resources.

Before you configure routing on an MCE, complete the following tasks:

- On the MCE, configure VPN instances, and bind the VPN instances with the interfaces connected to the VPN sites and those connected to the PE.
- Configure the link layer and network layer protocols on related interfaces to ensure IP connectivity.

Configuring routing between an MCE and a VPN site

You can configure static routing, RIPng, OSPFv3, IPv6 IS-IS, or EBGp between an MCE and a VPN site.

Configuring static routing between an MCE and a VPN site

An MCE can reach a VPN site through an IPv6 static route. IPv6 static routing on a traditional CE is globally effective and does not support address overlapping among VPNs. An MCE supports binding an IPv6 static route with an IPv6 VPN instance, so that the IPv6 static routes of different IPv6 VPN instances can be isolated from each other.

To configure IPv6 static routing between an MCE and a VPN site:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure an IPv6 static route for an IPv6 VPN instance.	ipv6 route-static vpn-instance <i>s-vpn-instance-name</i> <i>ipv6-address prefix-length</i> { <i>interface-type interface-number</i> [<i>next-hop-address</i>] <i>nexthop-address</i> [public] vpn-instance <i>d-vpn-instance-name nexthop-address</i> } [permanent] [preference <i>preference-value</i>] [tag <i>tag-value</i>] [description <i>description-text</i>]	Use either command as needed. Perform this configuration on the MCE. On a VPN site, configure normal IPv6 static routes.
3. (Optional.) Configure the default preference for IPv6 static routes.	ipv6 route-static default-preference <i>default-preference-value</i>	The default preference for IPv6 static routes is 60.

Configuring RIPng between an MCE and a VPN site

A RIPng process belongs to the public network or a single IPv6 VPN instance. If you create a RIPng process without binding it to an IPv6 VPN instance, the process belongs to the public network. By configuring RIPng process-to-IPv6 VPN instance bindings on a MCE, you allow routes of different VPNs

to be exchanged between the MCE and the sites through different RIPng processes, ensuring the separation and security of IPv6 VPN routes.

For more information about RIPng, see *Layer 3—IP Routing Configuration Guide*.

To configure RIPng between an MCE and a VPN site:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a RIPng process for a VPN instance and enter RIPng view.	ripng [<i>process-id</i>] vpn-instance <i>vpn-instance-name</i>	Perform this configuration on the MCE. On a VPN site, configure normal RIPng.
3. Redistribute remote site routes advertised by the PE.	import-route <i>protocol</i> [<i>process-id</i>] [allow-ibgp] [cost <i>cost</i>] route-policy <i>route-policy-name</i>] *	By default, no routes are redistributed into RIPng.
4. (Optional.) Configure the default cost value for the redistributed routes.	default cost <i>value</i>	The default value is 0.
5. Return to system view.	quit	N/A
6. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
7. Enable RIPng on the interface.	ripng <i>process-id</i> enable	RIPng is disabled by default.

Configuring OSPFv3 between an MCE and a VPN site

An OSPFv3 process belongs to the public network or a single IPv6 VPN instance. If you create an OSPFv3 process without binding it to an IPv6 VPN instance, the process belongs to the public network.

By configuring OSPFv3 process-to-IPv6 VPN instance bindings on a MCE, you allow routes of different IPv6 VPNs to be exchanged between the MCE and the sites through different OSPFv3 processes, ensuring the separation and security of IPv6 VPN routes.

For more information about OSPFv3, see *Layer 3—IP Routing Configuration Guide*.

To configure OSPFv3 between an MCE and a VPN site:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create an OSPFv3 process for a VPN instance and enter OSPFv3 view.	ospfv3 [<i>process-id</i> vpn-instance <i>vpn-instance-name</i>] *	Perform this configuration on the MCE. On a VPN site, configure common OSPFv3. The maximum number of OSPF processes that a VPN instance can run depends on the device's memory. Deleting a VPN instance also deletes all related OSPFv3 processes.
3. Set the router ID.	router-id <i>router-id</i>	N/A

Step	Command	Remarks
4. Redistribute remote site routes advertised by the PE.	import-route <i>protocol</i> [<i>process-id</i> all-processes allow-ibgp] [cost <i>cost</i> route-policy <i>route-policy-name</i> type type] *	By default, no routes are redistributed into OSPFv3.
5. Return to system view.	quit	N/A
6. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
7. Enable OSPFv3 on the interface.	ospfv3 <i>process-id</i> area <i>area-id</i> [instance <i>instance-id</i>]	By default, OSPFv3 is disabled on an interface.

Configuring IPv6 IS-IS between an MCE and a VPN site

An IPv6 IS-IS process belongs to the public network or a single IPv6 VPN instance. If you create an IPv6 IS-IS process without binding it to an IPv6 VPN instance, the process belongs to the public network.

By configuring IPv6 IS-IS process-to-IPv6 VPN instance bindings on a MCE, you allow routes of different IPv6 VPNs to be exchanged between the MCE and the sites through different IPv6 IS-IS processes, ensuring the separation and security of IPv6 VPN routes. For more information about IPv6 IS-IS, see *Layer 3—IP Routing Configuration Guide*.

To configure IPv6 IS-IS between an MCE and a VPN site:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create an IPv6 IS-IS process for a VPN instance and enter IS-IS view.	isis [<i>process-id</i>] vpn-instance <i>vpn-instance-name</i>	Perform this configuration on the MCE. On a VPN site, configure common IPv6 IS-IS.
3. Configure a network entity title for the IS-IS process.	network-entity <i>net</i>	By default, no NET is configured.
4. Enable IPv6 for the IPv6 IS-IS process.	ipv6 enable	By default, IPv6 is disabled.
5. (Optional.) Redistribute remote site routes advertised by the PE.	ipv6 import-route <i>protocol</i> [<i>process-id</i>] [allow-ibgp] [cost <i>cost</i> [level-1 level-1-2 level-2] route-policy <i>route-policy-name</i> tag tag] *	By default, no routes are redistributed to IPv6 IS-IS. If you do not specify the route level in the command, redistributed routes are added to the level-2 routing table.
6. Return to system view.	quit	N/A
7. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
8. Enable the IPv6 IS-IS process on the interface.	isis ipv6 enable [<i>process-id</i>]	By default, no IPv6 IS-IS process is enabled.

Configuring EBGP between an MCE and a VPN site

To use EBGP between an MCE and IPv6 VPN sites, you must configure a BGP peer for each IPv6 VPN instance on the MCE, and redistribute the IGP routes of each VPN instance on the IPv6 VPN sites. You can also configure the filtering of received and advertised routes.

1. Configure the MCE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Enter BGP-VPN view.	ip vpn-instance <i>vpn-instance-name</i>	N/A
4. Specify an IPv6 BGP peer in an AS.	peer { <i>group-name</i> <i>ipv6-address</i> } as-number <i>as-number</i>	By default, no BGP peer is configured.
5. Enter BGP-VPN IPv6 unicast address family view.	address-family ipv6 [unicast]	N/A
6. Enable BGP to exchange IPv6 unicast routes with the specified peer.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange IPv6 unicast routes with any peer.
7. Redistribute remote site routes advertised by the PE.	import-route <i>protocol</i> [<i>process-id</i> [med <i>med-value</i> route-policy <i>route-policy-name</i>] *]	By default, no route redistribution is configured.
8. (Optional.) Configure filtering of advertised routes.	filter-policy { <i>acl6-number</i> prefix-list <i>ipv6-prefix-name</i> } export [<i>protocol process-id</i>]	By default, BGP does not filter advertised routes.
9. (Optional.) Configure filtering of received routes.	filter-policy { <i>acl6-number</i> prefix-list <i>ipv6-prefix-name</i> } import	By default, BGP does not filter received routes.

2. Configure a VPN site:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Configure the MCE as an EBGP peer.	peer { <i>group-name</i> <i>ipv6-address</i> } as-number <i>as-number</i>	By default, no BGP peer is configured.
4. Enter BGP IPv6 unicast address family view.	address-family ipv6 [unicast]	N/A
5. Enable BGP to exchange IPv6 unicast routes with the specified peer.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange IPv6 unicast routes with any peer.
6. Redistribute the IGP routes of the VPN.	import-route <i>protocol</i> [<i>process-id</i> [med <i>med-value</i> route-policy <i>route-policy-name</i>] *]	By default, no routes are redistributed into BGP. A VPN site must advertise IPv6 VPN network addresses it can reach to the connected MCE.

Configuring IBGP between an MCE and a VPN site

To use IBGP between an MCE and a VPN site, you must configure a BGP peer for each IPv6 VPN instance on the MCE, and redistribute the IGP routes of each VPN instance on the VPN site.

1. Configure the MCE:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Enter BGP-VPN view.	ip vpn-instance <i>vpn-instance-name</i>	N/A
4. Configure an IBGP peer.	peer { <i>group-name</i> <i>ipv6-address</i> } as-number <i>as-number</i>	N/A
5. Enter BGP-VPN IPv6 unicast address family view.	address-family ipv6 [unicast]	N/A
6. Enable BGP to exchange IPv6 unicast routes with the peer.	peer { <i>group-name</i> <i>ipv6-address</i> } enable	By default, BGP does not exchange IPv6 unicast routes with any peer.
7. (Optional.) Configure the system to be the RR, and specify the peer as the client of the RR.	peer { <i>group-name</i> <i>ipv6-address</i> } reflect-client	By default, no RR or RR client is configured. After you configure a VPN site as an IBGP peer, the MCE does not advertise the BGP routes learned from the VPN site to other IBGP peers, including VPNv6 peers. The MCE advertises routes learned from a VPN site only when you configure the VPN site as a client of the RR (the MCE).
8. Redistribute remote site routes advertised by the PE into BGP.	import-route <i>protocol</i> [<i>process-id</i> [med <i>med-value</i> route-policy <i>route-policy-name</i>] *]	By default, no routes are redistributed into BGP.
9. (Optional.) Configure filtering of advertised routes.	filter-policy { <i>acl6-number</i> prefix-list <i>ipv6-prefix-name</i> } export [<i>protocol process-id</i>]	By default, BGP does not filter advertised routes.
10. (Optional.) Configure filtering of received routes.	filter-policy { <i>acl6-number</i> prefix-list <i>ipv6-prefix-name</i> } import	By default, BGP does not filter received routes.

2. Configure a VPN site:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Configure the MCE as an IBGP peer.	peer { <i>group-name</i> <i>ipv6-address</i> } as-number <i>as-number</i>	N/A

Step	Command	Remarks
4. Enter BGP-VPN IPv6 unicast address family view.	address-family ipv6 [unicast]	N/A
5. Enable BGP to exchange IPv6 unicast routes with the peer.	peer { group-name ipv6-address } enable	By default, BGP does not exchange IPv6 unicast routes with any peer.
6. Redistribute the IGP routes of the VPN into BGP.	import-route protocol [process-id [med med-value route-policy route-policy-name] *]	By default, no routes are redistributed into BGP. A VPN site must advertise VPN network addresses to the connected MCE.

Configuring routing between an MCE and a PE

MCE-PE routing configuration includes these tasks:

- Binding the MCE-PE interfaces to IPv6 VPN instances.
- Performing routing configurations.
- Redistributing IPv6 VPN routes into the routing protocol running between the MCE and the PE.

Perform the following configuration tasks on the MCE. Configurations on the PE are similar to those on the PE in common IPv6 MPLS L3VPN networks. For more information, see "[Configuring routing between a PE and a CE.](#)"

Configuring IPv6 static routing between an MCE and a PE

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Configure an IPv6 static route for an IPv6 VPN instance.	ipv6 route-static vpn-instance s-vpn-instance-name <i>ipv6-address prefix-length { interface-type interface-number [next-hop-address] nexthop-address [public] vpn-instance d-vpn-instance-name nexthop-address } [permanent] [preference preference-value] [tag tag-value] [description description-text]</i>	By default, no IPv6 static route is configured.
3. (Optional.) Configure the default preference for IPv6 static routes.	ipv6 route-static default-preference default-preference-value	The default value is 60.

Configuring RIPng between an MCE and a PE

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create a RIPng process for an IPv6 VPN instance and enter RIPng view.	ripng [process-id] vpn-instance vpn-instance-name	N/A

Step	Command	Remarks
3. Redistribute VPN routes.	import-route <i>protocol</i> [<i>process-id</i>] [allow-ibgp] [cost <i>cost</i> route-policy <i>route-policy-name</i>] *	By default, no routes are redistributed into RIPng.
4. (Optional.) Configure the default cost value for redistributed routes.	default cost <i>value</i>	The default value is 0.
5. Return to system view.	quit	N/A
6. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
7. Enable the RIPng process on the interface.	ripng <i>process-id</i> enable	By default, RIPng is disabled on an interface.

Configuring OSPFv3 between an MCE and a PE

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create an OSPFv3 process for an IPv6 VPN instance and enter OSPFv3 view.	ospfv3 [<i>process-id</i> vpn-instance <i>vpn-instance-name</i>] *	N/A
3. Set the router ID.	router-id <i>router-id</i>	N/A
4. Redistribute VPN routes.	import-route <i>protocol</i> [<i>process-id</i> all-processes allow-ibgp] [cost <i>cost</i> route-policy <i>route-policy-name</i> type <i>type</i>] *	By default, no routes are redistributed into OSPFv3.
5. (Optional.) Configure filtering of advertised routes.	filter-policy { <i>acl6-number</i> ipv6-prefix <i>ipv6-prefix-name</i> } export [bgp4+ direct isisv6 <i>process-id</i> ospfv3 <i>process-id</i> ripng <i>process-id</i> static]	By default, redistributed routes are not filtered.
6. Return to system view.	quit	N/A
7. Enter interface view.	interface <i>interface-type</i> <i>interface-number</i>	N/A
8. Enable the OSPFv3 process on the interface.	ospfv3 <i>process-id</i> area <i>area-id</i> [instance <i>instance-id</i>]	By default, OSPFv3 is disabled on an interface.

Configuring IPv6 IS-IS between an MCE and a PE

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Create an IS-IS process for an IPv6 VPN instance and enter IS-IS view.	isis [<i>process-id</i>] vpn-instance <i>vpn-instance-name</i>	N/A
3. Configure a network entity title.	network-entity <i>net</i>	By default, no NET is configured.
4. Enable IPv6 for the IS-IS process.	ipv6 enable	By default, IPv6 is disabled.

Step	Command	Remarks
5. (Optional.) Redistribute VPN routes.	ipv6 import-route <i>protocol</i> [<i>process-id</i>] [allow-ibgp] [cost cost [level-1 level-1-2 level-2] route-policy route-policy-name tag tag] *	By default, IPv6 IS-IS does not redistribute routes from any other routing protocol. If you do not specify the route level in the command, the command redistributes routes to the level-2 routing table.
6. (Optional.) Configure filtering of advertised routes.	ipv6 filter-policy { <i>acl6-number</i> prefix-list prefix-list-name route-policy route-policy-name } export [<i>protocol</i> [<i>process-id</i>]]	By default, IPv6 IS-IS does not filter advertised routes.
7. Return to system view.	quit	N/A
8. Enter interface view.	interface <i>interface-type interface-number</i>	N/A
9. Enable the IPv6 IS-IS process on the interface.	isis ipv6 enable [<i>process-id</i>]	By default, IPv6 IS-IS is disabled on an interface.

Configuring EBGP between an MCE and a PE

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enter BGP view.	bgp as-number	N/A
3. Enter BGP-VPN view.	ip vpn-instance vpn-instance-name	N/A
4. Configure the PE as an EBGP peer.	peer { <i>group-name</i> <i>ipv6-address</i> } as-number as-number	By default, no BGP peer is configured.
5. Enter BGP-VPN IPv6 unicast address family view.	address-family ipv6 [unicast]	N/A
6. Enable BGP to exchange IPv6 unicast routes with the specified peer.	peer { <i>group-name</i> <i>ip-address</i> } enable	By default, BGP does not exchange IPv6 unicast routes with any peer.
7. Redistribute VPN routes.	import-route <i>protocol</i> [<i>process-id</i>] [med med-value route-policy route-policy-name] *]	By default, no routes are redistributed into BGP.
8. (Optional.) Configure filtering of advertised routes.	filter-policy { <i>acl6-number</i> prefix-list ipv6-prefix-name } export [<i>protocol process-id</i>]	By default, BGP does not filter advertised routes.
9. (Optional.) Configure filtering of received routes.	filter-policy { <i>acl6-number</i> prefix-list ipv6-prefix-name } import	By default, BGP does not filter received routes.

Configuring IBGP between an MCE and a PE

Step	Command	Remarks
1. Enter system view.	system-view	N/A

Step	Command	Remarks
2. Enter BGP view.	bgp <i>as-number</i>	N/A
3. Enter BGP-VPN view.	ip vpn-instance <i>vpn-instance-name</i>	N/A
4. Configure the PE as an IBGP peer.	peer { <i>group-name</i> <i>ip-address</i> } as-number <i>as-number</i>	N/A
5. Enter BGP-VPN IPv6 unicast address family view.	address-family ipv6 [unicast]	N/A
6. Enable BGP to exchange IPv6 unicast routes with the peer.	peer { <i>group-name</i> <i>ipv6-address</i> } enable	By default, BGP does not exchange IPv6 unicast routes with any peer.
7. Redistribute the VPN routes of the VPN site.	import-route <i>protocol</i> [<i>process-id</i> [med <i>med-value</i> route-policy <i>route-policy-name</i>] *]	By default, no routes are redistributed into BGP.
8. (Optional.) Configure filtering of advertised routes.	filter-policy { <i>acl6-number</i> prefix-list <i>ipv6-prefix-name</i> } export [<i>protocol</i> <i>process-id</i>]	By default, BGP does not filter advertised routes.
9. (Optional.) Configure filtering of received routes.	filter-policy { <i>acl6-number</i> prefix-list <i>ipv6-prefix-name</i> } import	Optional. By default, BGP does not filter received routes.

Displaying and maintaining IPv6 MPLS L3VPN

You can soft-reset or reset BGP sessions to apply new BGP configurations. A soft reset operation updates BGP routing information without tearing down BGP connections. A reset operation updates BGP routing information by tearing down, and then reestablishing BGP connections. Soft reset requires that BGP peers have route refresh capability.

Use the following commands to soft reset or reset BGP sessions:

Task	Command	Remarks
Soft reset BGP sessions for VPNv6 address family.	refresh bgp { <i>ip-address</i> all external group <i>group-name</i> internal } { export import } vpn6	Available in user view.
Reset BGP sessions for VPNv6 address family.	reset bgp { <i>as-number</i> <i>ip-address</i> all external internal group <i>group-name</i> } vpn6	Available in user view.

Use the following commands to display IPv6 MPLS L3VPN:

Task	Command	Remarks
Display the IPv6 routing table for a VPN instance. For more information about this command, see <i>Layer 3—IP Routing Command Reference</i> .	display ipv6 routing-table vpn-instance <i>vpn-instance-name</i> [verbose]	Available in any view.

Task	Command	Remarks
Display information about a specific VPN instance or all VPN instances.	display ip vpn-instance [<i>instance-name</i> <i>vpn-instance-name</i>]	Available in any view.
Display the IPv6 FIB information for a VPN instance.	display ipv6 fib vpn-instance <i>vpn-instance-name</i> [acl6 <i>acl6-number</i> ipv6-prefix <i>ipv6-prefix-name</i>]	Available in any view.
Display FIB entries that match the specified destination IP address in the specified VPN instance.	display ipv6 fib vpn-instance <i>vpn-instance-name</i> <i>ipv6-address</i> [<i>prefix-length</i>]	Available in any view.
Display BGP VPNv6 peer group information.	display bgp group vpnv6 [<i>group-name</i>]	Available in any view.
Display BGP VPNv6 peer information.	display bgp peer vpnv6 [<i>group-name</i> log-info <i>ip-address</i> { log-info verbose } verbose]	Available in any view.
Display BGP VPNv6 routes.	display bgp routing-table vpnv6 [route-distinguisher <i>route-distinguisher</i>] [<i>network-address</i> <i>prefix-length</i>]	Available in any view.
Display BGP VPNv6 route advertisement information.	display bgp routing-table vpnv6 <i>network-address</i> <i>prefix-length</i> advertise-info	Available in any view.
Display BGP VPNv6 routes matching the specified AS PATH list.	display bgp routing-table vpnv6 [route-distinguisher <i>route-distinguisher</i>] as-path-acl <i>as-path-acl-number</i>	Available in any view.
Display BGP VPNv6 routes matching the specified BGP community list.	display bgp routing-table vpnv6 [route-distinguisher <i>route-distinguisher</i>] community-list { { <i>basic-community-list-number</i> <i>comm-list-name</i> } [whole-match] <i>adv-community-list-number</i> }	Available in any view.
Display BGP VPNv6 routes advertised to or received from the specified BGP peer.	display bgp routing-table vpnv6 peer <i>ip-address</i> { advertised-routes received-routes } [<i>network-address</i> <i>prefix-length</i> statistics]	Available in any view.
Display incoming labels for all BGP VPNv6 routes.	display bgp routing-table vpnv6 inlabel	Available in any view.
Display outgoing labels for all BGP VPNv6 routes.	display bgp routing-table vpnv6 outlabel	Available in any view.
Display BGP VPNv6 route statistics.	display bgp routing-table vpnv6 statistics	Available in any view.
Display BGP VPNv6 address family update group information.	display bgp update-group vpnv6 [<i>ip-address</i>]	Available in any view.

IPv6 MPLS L3VPN configuration examples

Configuring IPv6 MPLS L3VPNs

Network requirements

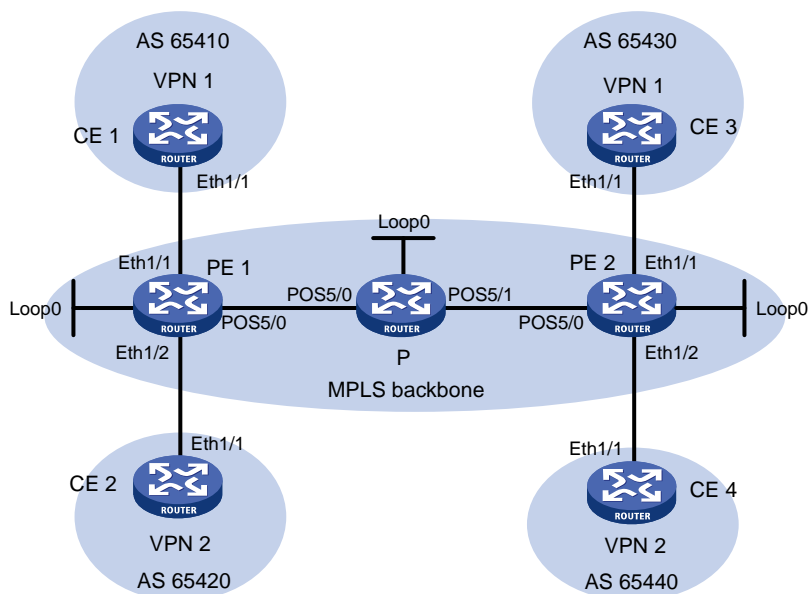
CE 1 and CE 3 belong to VPN 1. CE 2 and CE 4 belong to VPN 2.

VPN 1 uses route target attributes 111:1. VPN 2 uses route target attributes 222:2. Users of different VPNs cannot access each other.

Run EBGP between CEs and PEs to exchange VPN routing information.

PEs use OSPF to communicate with each other and use MP-IBGP to exchange VPN routing information.

Figure 62 Network diagram



Device	Interface	IP address	Device	Interface	IP address
CE 1	Eth1/1	2001:1::1/96	P	Loop0	2.2.2.9/32
PE 1	Loop0	1.1.1.9/32		POS5/0	172.1.1.2/24
	Eth1/1	2001:1::2/96		POS5/1	172.2.1.1/24
	Eth1/2	2001:2::2/96	PE 2	Loop0	3.3.3.9/32
	POS5/0	172.1.1.1/24		Eth1/1	2001:3::2/96
CE 2	Eth1/1	2001:2::1/96		Eth1/2	2001:4::2/96
CE 3	Eth1/1	2001:3::1/96		POS5/0	172.2.1.2/24
CE 4	Eth1/1	2001:4::1/96			

Configuration procedure

1. Configure OSPF on the MPLS backbone to achieve IP connectivity among the PEs and the P router:

Configure PE 1.

```
<PE1> system-view
```

```
[PE1] interface loopback 0
```

```
[PE1-LoopBack0] ip address 1.1.1.9 32
```



```

[PE1-LoopBack0] quit
[PE1] interface pos 5/0
[PE1-POS5/0] ip address 172.1.1.1 24
[PE1-POS5/0] quit
[PE1] ospf
[PE1-ospf-1] area 0
[PE1-ospf-1-area-0.0.0.0] network 172.1.1.0 0.0.0.255
[PE1-ospf-1-area-0.0.0.0] network 1.1.1.9 0.0.0.0
[PE1-ospf-1-area-0.0.0.0] quit
[PE1-ospf-1] quit

```

Configure the P router.

```

<P> system-view
[P] interface loopback 0
[P-LoopBack0] ip address 2.2.2.9 32
[P-LoopBack0] quit
[P] interface pos 5/0
[P-POS5/0] ip address 172.1.1.2 24
[P-POS5/0] quit
[P] interface pos 5/1
[P-POS5/1] ip address 172.2.1.1 24
[P-POS5/1] quit
[P] ospf
[P-ospf-1] area 0
[P-ospf-1-area-0.0.0.0] network 172.1.1.0 0.0.0.255
[P-ospf-1-area-0.0.0.0] network 172.2.1.0 0.0.0.255
[P-ospf-1-area-0.0.0.0] network 2.2.2.9 0.0.0.0
[P-ospf-1-area-0.0.0.0] quit
[P-ospf-1] quit

```

Configure PE 2.

```

<PE2> system-view
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 3.3.3.9 32
[PE2-LoopBack0] quit
[PE2] interface pos 5/0
[PE2-POS5/0] ip address 172.2.1.2 24
[PE2-POS5/0] quit
[PE2] ospf
[PE2-ospf-1] area 0
[PE2-ospf-1-area-0.0.0.0] network 172.2.1.0 0.0.0.255
[PE2-ospf-1-area-0.0.0.0] network 3.3.3.9 0.0.0.0
[PE2-ospf-1-area-0.0.0.0] quit
[PE2-ospf-1] quit

```

After the configurations, OSPF adjacencies are established between PE 1, P, and PE 2. Execute the **display ospf peer** command. The output shows that the adjacency status is Full. Execute the **display ip routing-table** command. The output shows that the PEs have learned the routes to the loopback interfaces of each other. Take PE 1 as an example:

```

[PE1] display ip routing-table protocol ospf

```

Summary Count : 5

OSPF Routing table Status : <Active>

Summary Count : 3

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
2.2.2.9/32	OSPF	10	1	172.1.1.2	POS5/0
3.3.3.9/32	OSPF	10	2	172.1.1.2	POS5/0
172.2.1.0/24	OSPF	10	2	172.1.1.2	POS5/0

OSPF Routing table Status : <Inactive>

Summary Count : 2

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
1.1.1.9/32	OSPF	10	0	1.1.1.9	Loop0
172.1.1.0/24	OSPF	10	1	172.1.1.1	POS5/0

[PE1] display ospf peer verbose

OSPF Process 1 with Router ID 1.1.1.9

Neighbors

Area 0.0.0.0 interface 172.1.1.1(POS5/0)'s neighbors

Router ID: 2.2.2.9 Address: 172.1.1.2 GR State: Normal

State: Full Mode: Nbr is Master Priority: 1

DR: 172.1.1.2 BDR: 172.1.1.1 MTU: 0

Options is 0x02 (-|-|-|-|-|E|-)

Dead timer due in 39 sec

Neighbor is up for 00:00:29

Authentication Sequence: [0]

Neighbor state change count: 6

2. Configure basic MPLS and enable MPLS LDP on the MPLS backbone to establish LDP LSPs:

Configure PE 1.

```
[PE1] mpls lsr-id 1.1.1.9
```

```
[PE1] mpls ldp
```

```
[PE1-ldp] quit
```

```
[PE1] interface pos 5/0
```

```
[PE1-POS5/0] mpls enable
```

```
[PE1-POS5/0] mpls ldp enable
```

```
[PE1-POS5/0] quit
```

Configure the P router.

```
[P] mpls lsr-id 2.2.2.9
```

```
[P] mpls ldp
```

```
[P-ldp] quit
```

```
[P] interface pos 5/0
```

```
[P-POS5/0] mpls enable
```

```
[P-POS5/0] mpls ldp enable
```

```
[P-POS5/0] quit
```

```
[P] interface pos 5/1
```

```
[P-POS5/1] mpls enable
[P-POS5/1] mpls ldp enable
[P-POS5/1] quit
```

Configure PE 2.

```
[PE2] mpls lsr-id 3.3.3.9
[PE2] mpls ldp
[PE2-ldp] quit
[PE2] interface pos 5/0
[PE2-POS5/0] mpls enable
[PE2-POS5/0] mpls ldp enable
[PE2-POS5/0] quit
```

After the configurations, LDP sessions are established between PE 1, P, and PE 2. Execute the **display mpls ldp peer** command. The output shows that the session status is Operational. Execute the **display mpls ldp lsp** command. The output shows the LSPs established by LDP. Take PE 1 as an example:

```
[PE1] display mpls ldp peer
Total number of peers: 1
Peer LDP ID      State          LAM  Role    GR   MD5  KA Sent/Rcvd
2.2.2.9:0        Operational    DU   Passive Off  Off  5/5
[PE1] display mpls ldp lsp
      Status codes: * - stale, L - liberal
      Statistics:
      FECs: 3      Ingress LSPs: 2      Transit LSPs: 2      Egress LSPs: 1

FEC              In/Out Label      Nexthop            OutInterface
1.1.1.9/32       3/-
                  -/1151(L)
2.2.2.9/32       -/3               172.1.1.2          POS5/0
                  1151/3            172.1.1.2          POS5/0
3.3.3.9/32       -/1150            172.1.1.2          POS5/0
                  1150/1150         172.1.1.2          POS5/0
```

3. Configure IPv6 VPN instances on the PEs to allow CE access:

Configure PE 1.

```
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 100:1
[PE1-vpn-instance-vpn1] vpn-target 111:1
[PE1-vpn-instance-vpn1] quit
[PE1] ip vpn-instance vpn2
[PE1-vpn-instance-vpn2] route-distinguisher 100:2
[PE1-vpn-instance-vpn2] vpn-target 222:2
[PE1-vpn-instance-vpn2] quit
[PE1] interface ethernet 1/1
[PE1-Ethernet1/1] ip binding vpn-instance vpn1
[PE1-Ethernet1/1] ipv6 address 2001:1::2 96
[PE1-Ethernet1/1] quit
[PE1] interface ethernet 1/2
[PE1-Ethernet1/2] ip binding vpn-instance vpn2
[PE1-Ethernet1/2] ipv6 address 2001:2::2 96
```

```
[PE1-Ethernet1/2] quit
```

Configure PE 2.

```
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance-vpn1] route-distinguisher 200:1
[PE2-vpn-instance-vpn1] vpn-target 111:1
[PE2-vpn-instance-vpn1] quit
[PE2] ip vpn-instance vpn2
[PE2-vpn-instance-vpn2] route-distinguisher 200:2
[PE2-vpn-instance-vpn2] vpn-target 222:2
[PE2-vpn-instance-vpn2] quit
[PE2] interface ethernet 1/1
[PE2-Ethernet1/1] ip binding vpn-instance vpn1
[PE2-Ethernet1/1] ipv6 address 2001:3::2 96
[PE2-Ethernet1/1] quit
[PE2] interface ethernet 1/2
[PE2-Ethernet1/2] ip binding vpn-instance vpn2
[PE2-Ethernet1/2] ipv6 address 2001:4::2 96
[PE2-Ethernet1/2] quit
```

Configure IP addresses for the CEs according to [Figure 62](#). (Details not shown.)

After completing the configurations, execute the **display ip vpn-instance** command on the PEs to display information about the VPN instances. Use the **ping** command to test connectivity between the PEs and their attached CEs. The PEs can ping their attached CEs. Take PE 1 as an example:

```
[PE1] display ip vpn-instance
  Total VPN-Instances configured : 2
  VPN-Instance Name           RD           Create time
  vpn1                        100:1       2012/02/13 12:49:08
  vpn2                        100:2       2012/02/13 12:49:20
[PE1] ping ipv6 -vpn-instance vpn1 2001:1::1
Ping6(56 bytes) 2001:1::2 --> 2001:1::1, press escape sequence to break
56 bytes from 2001:1::1, icmp_seq=0 hlim=64 time=9.000 ms
56 bytes from 2001:1::1, icmp_seq=1 hlim=64 time=1.000 ms
56 bytes from 2001:1::1, icmp_seq=2 hlim=64 time=0.000 ms
56 bytes from 2001:1::1, icmp_seq=3 hlim=64 time=0.000 ms
56 bytes from 2001:1::1, icmp_seq=4 hlim=64 time=0.000 ms

--- Ping6 statistics for 2001:1::1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/2.000/9.000/3.521 ms
```

4. Establish EBGP peer relationships between the PEs and CEs to allow them to exchange VPN routes:

Configure CE 1.

```
<CE1> system-view
[CE1] bgp 65410
[CE1-bgp] peer 2001:1::2 as-number 100
[CE1-bgp] address-family ipv6 unicast
[CE1-bgp-ipv6] peer 2001:1::2 enable
[CE1-bgp-ipv6] import-route direct
```

```
[CE1-bgp-ipv6] quit
```

```
[CE1-bgp] quit
```

Configure the other CEs (CE 2 through CE 4) in the same way that CE 1 is configured. (Details not shown.)

Configure PE 1.

```
[PE1] bgp 100
```

```
[PE1-bgp] ip vpn-instance vpn1
```

```
[PE1-bgp-vpn1] peer 2001:1::1 as-number 65410
```

```
[PE1-bgp-vpn1] address-family ipv6 unicast
```

```
[PE1-bgp-ipv6-vpn1] peer 2001:1::1 enable
```

```
[PE1-bgp-ipv6-vpn1] import-route direct
```

```
[PE1-bgp-ipv6-vpn1] quit
```

```
[PE1-bgp-vpn1] quit
```

```
[PE1-bgp] ip vpn-instance vpn2
```

```
[PE1-bgp-vpn2] peer 2001:2::1 as-number 65420
```

```
[PE1-bgp-vpn2] address-family ipv6 unicast
```

```
[PE1-bgp-ipv6-vpn2] peer 2001:2::1 enable
```

```
[PE1-bgp-ipv6-vpn2] import-route direct
```

```
[PE1-bgp-ipv6-vpn2] quit
```

```
[PE1-bgp-vpn2] quit
```

```
[PE1-bgp] quit
```

Configure PE 2 in the same way that PE 1 is configured. (Details not shown.)

After completing the configurations, execute the **display bgp peer ipv6 vpn-instance** command on the PEs. BGP peer relationships have been established between the PEs and CEs and have reached the Established state.

5. Configure an MP-IBGP peer relationship between the PEs:

Configure PE 1.

```
[PE1] bgp 100
```

```
[PE1-bgp] peer 3.3.3.9 as-number 100
```

```
[PE1-bgp] peer 3.3.3.9 connect-interface loopback 0
```

```
[PE1-bgp] address-family vpnv6
```

```
[PE1-bgp-vpnv6] peer 3.3.3.9 enable
```

```
[PE1-bgp-vpnv6] quit
```

```
[PE1-bgp] quit
```

Configure PE 2.

```
[PE2] bgp 100
```

```
[PE2-bgp] peer 1.1.1.9 as-number 100
```

```
[PE2-bgp] peer 1.1.1.9 connect-interface loopback 0
```

```
[PE2-bgp] address-family vpnv6
```

```
[PE2-bgp-vpnv6] peer 1.1.1.9 enable
```

```
[PE2-bgp-vpnv6] quit
```

```
[PE2-bgp] quit
```

After completing the configurations, execute the **display bgp peer vpnv6** command on the PEs. The output shows that a BGP peer relationship has been established between the PEs and has reached the Established state.

Verifying the configuration

Execute the **display ipv6 routing-table vpn-instance** command on the PEs. The output shows the routes to the CEs. Take PE 1 as an example:

```
[PE1] display ipv6 routing-table vpn-instance vpn1

Destinations : 6 Routes : 6

Destination: ::1/128                Protocol : Direct
NextHop      : ::1                  Preference: 0
Interface    : InLoop0              Cost      : 0

Destination: 2001:1::/96            Protocol : Direct
NextHop      : ::                  Preference: 0
Interface    : Eth1/1              Cost      : 0

Destination: 2001:1::2/128          Protocol : Direct
NextHop      : ::1                Preference: 0
Interface    : InLoop0            Cost      : 0

Destination: 2001:3::/96            Protocol : BGP4+
NextHop      : ::FFFF:3.3.3.9      Preference: 255
Interface    : POS5/0             Cost      : 0

Destination: FE80::/10              Protocol : Direct
NextHop      : ::                  Preference: 0
Interface    : NULL0              Cost      : 0

Destination: FF00::/8               Protocol : Direct
NextHop      : ::                  Preference: 0
Interface    : NULL0              Cost      : 0

[PE1] display ipv6 routing-table vpn-instance vpn2

Destinations : 6 Routes : 6

Destination: ::1/128                Protocol : Direct
NextHop      : ::1                  Preference: 0
Interface    : InLoop0              Cost      : 0

Destination: 2001:2::/96            Protocol : Direct
NextHop      : ::                  Preference: 0
Interface    : Eth1/2              Cost      : 0

Destination: 2001:2::2/128          Protocol : Direct
NextHop      : ::1                Preference: 0
Interface    : InLoop0            Cost      : 0

Destination: 2001:4::/96            Protocol : BGP4+
NextHop      : ::FFFF:3.3.3.9      Preference: 255
```

```

Interface : POS5/0                                     Cost : 0

Destination: FE80::/10                                Protocol : Direct
NextHop : ::                                          Preference: 0
Interface : NULL0                                     Cost : 0

Destination: FF00::/8                                 Protocol : Direct
NextHop : ::                                          Preference: 0
Interface : NULL0                                     Cost : 0

```

From each CE, ping other CEs. CEs of the same VPN can ping each other, whereas those of different VPNs should not. For example, CE 1 can ping CE 3 (2001:3::1), but cannot ping CE 4 (2001:4::1).

Configuring an IPv6 MPLS L3VPN over a GRE tunnel

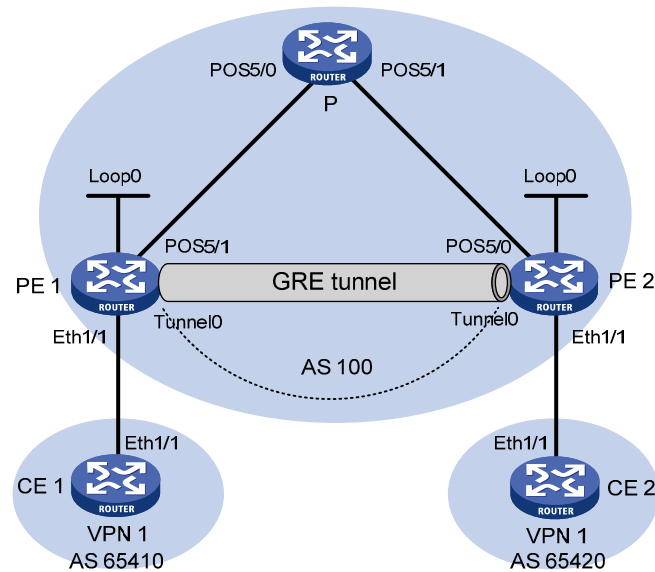
Network requirements

CE 1 and CE 2 belong to VPN 1. The PEs support MPLS, while the P router does not support MPLS and provides only IP functions.

On the backbone, use a GRE tunnel to encapsulate and forward packets for IPv6 MPLS L3VPN.

Configure tunnel policies on the PEs, and specify the tunnel type for VPN traffic as GRE.

Figure 63 Network diagram



Device	Interface	IP address	Device	Interface	IP address
CE 1	Eth1/1	2001:1::1/96	P	POS5/0	172.1.1.2/24
PE 1	Loop0	1.1.1.9/32	PE 2	POS5/1	172.2.1.1/24
	Eth1/1	2001:1::2/96		Loop0	2.2.2.9/32
	POS5/1	172.1.1.1/24		Eth1/1	2001:2::2/96
	Tunnel0	20.1.1.1/24		POS5/0	172.2.1.2/24
CE 2	Eth1/1	2001:2::1/96		Tunnel0	20.1.1.2/24

Configuration procedure

1. Configure an IGP on the MPLS backbone to achieve IP connectivity among the PEs and the P router:

This example uses OSPF. (Details not shown.)

After the configurations, OSPF adjacencies are established between PE 1, P, and PE 2. Execute the **display ospf peer** command. The output shows that the adjacency status is Full. Execute the **display ip routing-table** command. The output shows that the PEs have learned the routes to the loopback interfaces of each other.

2. Configure basic MPLS on the PEs:

Configure PE 1.

```
<PE1> system-view
[PE1] mpls lsr-id 1.1.1.9
```

Configure PE 2.

```
<PE2> system-view
[PE2] mpls lsr-id 2.2.2.9
```

3. Configure VPN instances on the PEs to allow CE access, and apply tunnel policies to the VPN instances to use a GRE tunnel for VPN packet forwarding:

Configure PE 1.

```
[PE1] tunnel-policy gre1
[PE1-tunnel-policy-gre1] tunnel select-seq gre load-balance-number 1
[PE1-tunnel-policy-gre1] quit
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 100:1
[PE1-vpn-instance-vpn1] vpn-target 100:1 both
[PE1-vpn-instance-vpn1] tnl-policy gre1
[PE1-vpn-instance-vpn1] quit
[PE1] interface ethernet 1/1
[PE1-Ethernet1/1] ip binding vpn-instance vpn1
[PE1-Ethernet1/1] ipv6 address 2001:1::2 96
[PE1-Ethernet1/1] quit
```

Configure PE 2.

```
[PE2] tunnel-policy gre1
[PE2-tunnel-policy-gre1] tunnel select-seq gre load-balance-number 1
[PE2-tunnel-policy-gre1] quit
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance-vpn1] route-distinguisher 100:2
[PE2-vpn-instance-vpn1] vpn-target 100:1 both
[PE2-vpn-instance-vpn1] tnl-policy gre1
[PE2-vpn-instance-vpn1] quit
[PE2] interface ethernet 1/1
[PE2-Ethernet1/1] ip binding vpn-instance vpn1
[PE2-Ethernet1/1] ipv6 address 2001:2::2 96
[PE2-Ethernet1/1] quit
```

Configure CE 1.

```
<CE1> system-view
[CE1] interface ethernet 1/1
```



```
[CE1-Ethernet1/1] ipv6 address 2001:1::1 96
[CE1-Ethernet1/1] quit
```

Configure CE 2.

```
<CE2> system-view
[CE2] interface ethernet 1/1
[CE2-Ethernet1/1] ipv6 address 2001:2::1 96
[CE2-Ethernet1/1] quit
```

After completing the configurations, execute the **display ip vpn-instance** command on the PEs to display information about the VPN instance. Use the **ping** command to test connectivity between the PEs and their attached CEs. The PEs can ping their attached CEs. Take PE 1 as an example:

```
[PE1] display ip vpn-instance
Total VPN-Instances configured : 1
VPN-Instance Name          RD          Create time
vpn1                       100:1      2012/02/13 15:59:50
[PE1] ping ipv6 -vpn-instance vpn1 2001:1::1
Ping6(56 bytes) 2001:1::2 --> 2001:1::1, press escape sequence to break
56 bytes from 2001:1::1, icmp_seq=0 hlim=64 time=0.000 ms
56 bytes from 2001:1::1, icmp_seq=1 hlim=64 time=1.000 ms
56 bytes from 2001:1::1, icmp_seq=2 hlim=64 time=0.000 ms
56 bytes from 2001:1::1, icmp_seq=3 hlim=64 time=1.000 ms
56 bytes from 2001:1::1, icmp_seq=4 hlim=64 time=0.000 ms

--- Ping6 statistics for 2001:1::1 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.000/0.400/1.000/0.490 ms
```

4. Establish EBGp peer relationships between PEs and CEs to allow them to exchange VPN routes:

Configure CE 1.

```
[CE1] bgp 65410
[CE1-bgp] peer 2001:1::2 as-number 100
[CE1-bgp] address-family ipv6 unicast
[CE1-bgp-ipv6] peer 2001:1::2 enable
[CE1-bgp-ipv6] import-route direct
[CE1-bgp-ipv6] quit
```

Configure PE 1.

```
[PE1] bgp 100
[PE1-bgp] ip vpn-instance vpn1
[PE1-bgp-vpn1] peer 2001:1::2 as-number 65410
[PE1-bgp-vpn1] address-family ipv6 unicast
[PE1-bgp-ipv6-vpn1] peer 2001:1::2 enable
[PE1-bgp-ipv6-vpn1] import-route direct
[PE1-bgp-ipv6-vpn1] quit
[PE1-bgp-vpn1] quit
[PE1-bgp] quit
```

Configure CE 2 and PE 2 in the same way that CE 1 and PE 1 are configured. (Details not shown.)

After completing the configurations, execute the **display bgp peer ipv6 vpn-instance** command on the PEs. BGP peer relationships have been established between PEs and CEs, and have reached Established state.

Take PE 1 as an example:

```
[PE1] display bgp peer ipv6 vpn-instance vpn1

BGP local router ID : 1.1.1.9
Local AS number : 100
Total number of peers : 1                Peers in established state : 1

Peer                AS  MsgRcvd  MsgSent  OutQ  PrefRcv  Up/Down  State
2001:1::1          65410    5         5       0       1 00:02:03 Established
```

5. Configure an MP-IBGP peer relationship between the PEs:

Configure PE 1.

```
[PE1] bgp 100
[PE1-bgp] peer 2.2.2.9 as-number 100
[PE1-bgp] peer 2.2.2.9 connect-interface loopback 0
[PE1-bgp] address-family vpnv6
[PE1-bgp-vpnv6] peer 2.2.2.9 enable
[PE1-bgp-vpnv6] quit
[PE1-bgp] quit
```

Configure PE 2 in the same way that PE 1 is configured. (Details not shown.)

After completing the configuration, execute the **display bgp peer vpnv6** command on the PEs. A BGP peer relationship has been established between the PEs, and has reached Established state.

```
[PE1] display bgp peer vpnv6

BGP local router ID : 1.1.1.9
Local AS number : 100
Total number of peers : 1                Peers in established state : 1

Peer                AS  MsgRcvd  MsgSent  OutQ  PrefRcv  Up/Down  State
2.2.2.9            100     3         3       0       1 00:00:34 Established
```

6. Configure a GRE tunnel:

Configure PE 1.

```
[PE1] interface tunnel 0 mode gre
[PE1-Tunnel0] source loopback 0
[PE1-Tunnel0] destination 2.2.2.9
[PE1-Tunnel0] ip address 20.1.1.1 24
[PE1-Tunnel0] mpls enable
[PE1-Tunnel0] quit
```

Configure PE 2.

```
[PE2] interface tunnel 0 mode gre
[PE2-Tunnel0] source loopback 0
[PE2-Tunnel0] destination 1.1.1.9
[PE2-Tunnel0] ip address 20.1.1.2 24
[PE2-Tunnel0] mpls enable
[PE2-Tunnel0] quit
```

Verifying the configuration

The CEs have learned the route to each other and can ping each other.

Configuring IPv6 MPLS L3VPN inter-AS option A

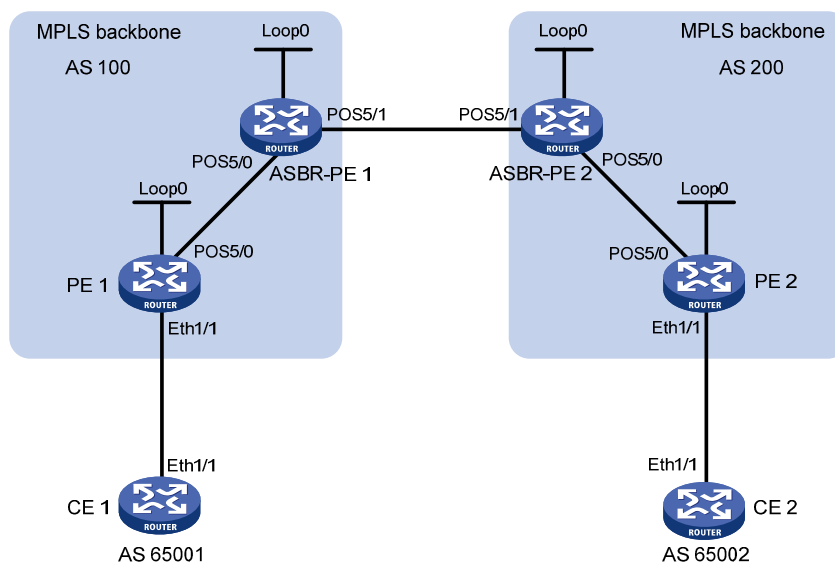
Network requirements

CE 1 and CE 2 belong to the same VPN. CE 1 accesses the network through PE 1 in AS 100 and CE 2 accesses the network through PE 2 in AS 200.

Configure IPv6 MPLS L3VPN inter-AS option A, and use VRF-to-VRF method to manage VPN routes.

Run OSPF on the MPLS backbone of each AS.

Figure 64 Network diagram



Device	Interface	IP address	Device	Interface	IP address
CE 1	Eth1/1	2001:1::1/96	CE 2	Eth1/1	2001:2::1/96
PE 1	Loop0	1.1.1.9/32	PE 2	Loop0	4.4.4.9/32
	Eth1/1	2001:1::2/96		Eth1/1	2001:2::2/96
	POS5/0	172.1.1.2/24		POS5/0	162.1.1.2/24
ASBR-PE1	Loop0	2.2.2.9/32	ASBR-PE2	Loop0	3.3.3.9/32
	POS5/0	172.1.1.1/24		POS5/0	162.1.1.1/24
	POS5/1	2002:1::1/96		POS5/1	2002:1::2/96

Configuration procedure

- Configure an IGP on each MPLS backbone to ensure IP connectivity within the backbone:

This example uses OSPF. Be sure to advertise the route to the 32-bit loopback interface address of each router through OSPF. Use the loopback interface address of a router as the router's LSR ID. (Details not shown.)

After the configurations, each ASBR PE and the PE in the same AS can establish an OSPF adjacency. Execute the **display ospf peer** command and **ping** command. The output shows that the adjacencies are in Full state, and that the PE and ASBR PE in the same AS have learned the routes to the loopback interfaces of each other and can ping each other.

2. Configure basic MPLS and enable MPLS LDP on each MPLS backbone to establish LDP LSPs:

Configure basic MPLS on PE 1 and enable MPLS LDP for both PE 1 and the interface connected to ASBR-PE 1.

```
<PE1> system-view
[PE1] mpls lsr-id 1.1.1.9
[PE1] mpls ldp
[PE1-ldp] quit
[PE1] interface pos 5/0
[PE1-POS5/0] mpls enable
[PE1-POS5/0] mpls ldp enable
[PE1-POS5/0] quit
```

Configure basic MPLS on ASBR-PE 1 and enable MPLS LDP for both ASBR-PE 1 and the interface connected to PE 1.

```
<ASBR-PE1> system-view
[ASBR-PE1] mpls lsr-id 2.2.2.9
[ASBR-PE1] mpls ldp
[ASBR-PE1-ldp] quit
[ASBR-PE1] interface pos 5/0
[ASBR-PE1-POS5/0] mpls enable
[ASBR-PE1-POS5/0] mpls ldp enable
[ASBR-PE1-POS5/0] quit
```

Configure basic MPLS on ASBR-PE 2 and enable MPLS LDP for both ASBR-PE 2 and the interface connected to PE 2.

```
<ASBR-PE2> system-view
[ASBR-PE2] mpls lsr-id 3.3.3.9
[ASBR-PE2] mpls ldp
[ASBR-PE2-ldp] quit
[ASBR-PE2] interface pos 5/0
[ASBR-PE2-POS5/0] mpls enable
[ASBR-PE2-POS5/0] mpls ldp enable
[ASBR-PE2-POS5/0] quit
```

Configure basic MPLS on PE 2 and enable MPLS LDP for both PE 2 and the interface connected to ASBR-PE 2.

```
<PE2> system-view
[PE2] mpls lsr-id 4.4.4.9
[PE2] mpls ldp
[PE2-ldp] quit
[PE2] interface pos 5/0
[PE2-POS5/0] mpls enable
[PE2-POS5/0] mpls ldp enable
[PE2-POS5/0] quit
```

After the configurations, each PE and the ASBR PE in the same AS can establish an LDP neighbor relationship. Execute the **display mpls ldp session** command on the routers. The output shows that the session status is Operational.

3. Configure a VPN instance on the PEs:

For the same VPN, the route targets for the VPN instance on the PE must match those for the VPN instance on the ASBR-PE in the same AS. This is not required for PEs in different ASs.

Configure CE 1.

```
<CE1> system-view
[CE1] interface ethernet 1/1
[CE1-Ethernet1/1] ipv6 address 2001:1::1 96
[CE1-Ethernet1/1] quit
```

Configure PE 1.

```
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 100:1
[PE1-vpn-instance-vpn1] vpn-target 100:1 both
[PE1-vpn-instance-vpn1] quit
[PE1] interface ethernet 1/1
[PE1-Ethernet1/1] ip binding vpn-instance vpn1
[PE1-Ethernet1/1] ipv6 address 2001:1::2 96
[PE1-Ethernet1/1] quit
```

Configure CE 2.

```
<CE2> system-view
[CE2] interface ethernet 1/1
[CE2-Ethernet1/1] ipv6 address 2001:2::1 96
[CE2-Ethernet1/1] quit
```

Configure PE 2.

```
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance-vpn1] route-distinguisher 200:1
[PE2-vpn-instance-vpn1] vpn-target 100:1 both
[PE2-vpn-instance-vpn1] quit
[PE2] interface ethernet 1/1
[PE2-Ethernet1/1] ip binding vpn-instance vpn1
[PE2-Ethernet1/1] ipv6 address 2001:2::2 96
[PE2-Ethernet1/1] quit
```

Configure ASBR-PE 1, creating a VPN instance and binding the VPN instance to the interface connected to ASBR-PE 2 (ASBR-PE 1 considers ASBR-PE 2 its attached CE).

```
[ASBR-PE1] ip vpn-instance vpn1
[ASBR-PE1-vpn-vpn1] route-distinguisher 100:1
[ASBR-PE1-vpn-vpn1] vpn-target 100:1 both
[ASBR-PE1-vpn-vpn1] quit
[ASBR-PE1] interface pos 5/1
[ASBR-PE1-POS5/1] ip binding vpn-instance vpn1
[ASBR-PE1-POS5/1] ipv6 address 2002:1::1 96
[ASBR-PE1-POS5/1] quit
```

Configure ASBR-PE 2, creating a VPN instance and binding the VPN instance to the interface connected to ASBR-PE 1 (ASBR-PE 2 considers ASBR-PE 1 its attached CE).

```
[ASBR-PE2] ip vpn-instance vpn1
[ASBR-PE2-vpn-vpn1] route-distinguisher 200:1
[ASBR-PE2-vpn-vpn1] vpn-target 100:1 both
[ASBR-PE2-vpn-vpn1] quit
[ASBR-PE2] interface pos 5/1
[ASBR-PE2-POS5/1] ip binding vpn-instance vpn1
[ASBR-PE2-POS5/1] ipv6 address 2002:1::2 96
```

```
[ASBR-PE2-POS5/1] quit
```

After completing the configurations, you can view the VPN instance information by executing the **display ip vpn-instance** command.

Each PE can ping its attached CE, and ASBR-PE 1 and ASBR-PE 2 can ping each other.

4. Establish EBGP peer relationships between PEs and CEs to allow them to exchange VPN routes:

Configure CE 1.

```
[CE1] bgp 65001
[CE1-bgp] peer 2001:1::2 as-number 100
[CE1-bgp] address-family ipv6 unicast
[CE1-bgp-ipv6] peer 2001:1::2 enable
[CE1-bgp-ipv6] import-route direct
[CE1-bgp-ipv6] quit
[CE1-bgp] quit
```

Configure PE 1.

```
[PE1] bgp 100
[PE1-bgp] ip vpn-instance vpn1
[PE1-bgp-vpn1] peer 2001:1::1 as-number 65001
[PE1-bgp-vpn1] address-family ipv6 unicast
[PE1-bgp-ipv6-vpn1] peer 2001:1::1 enable
[PE1-bgp-ipv6-vpn1] import-route direct
[PE1-bgp-ipv6-vpn1] quit
[PE1-bgp-vpn1] quit
[PE1-bgp] quit
```

Configure CE 2.

```
[CE2] bgp 65002
[CE2-bgp] peer 2001:2::2 as-number 200
[CE2-bgp] address-family ipv6
[CE2-bgp-ipv6] peer 2001:2::2 enable
[CE2-bgp-ipv6] import-route direct
[CE2-bgp-ipv6] quit
[CE2-bgp] quit
```

Configure PE 2.

```
[PE2] bgp 200
[PE2-bgp] ip vpn-instance vpn1
[PE2-bgp-vpn1] peer 2001:2::1 as-number 65002
[PE2-bgp-vpn1] address-family ipv6 unicast
[PE2-bgp-ipv6-vpn1] peer 2001:2::1 enable
[PE2-bgp-ipv6-vpn1] import-route direct
[PE2-bgp-ipv6-vpn1] quit
[PE2-bgp-vpn1] quit
[PE2-bgp] quit
```

5. Establish an IBGP peer relationship between each PE and the ASBR PE in the same AS and an EBGP peer relationship between the ASBR PEs:

Configure PE 1.

```
[PE1] bgp 100
[PE1-bgp] peer 2.2.2.9 as-number 100
[PE1-bgp] peer 2.2.2.9 connect-interface loopback 0
```

```

[PE1-bgp] address-family vpnv6
[PE1-bgp-vpnv6] peer 2.2.2.9 enable
[PE1-bgp-vpnv6] quit
[PE1-bgp] quit
# Configure ASBR-PE 1.
[ASBR-PE1] bgp 100
[ASBR-PE1-bgp] ip vpn-instance vpn1
[ASBR-PE1-bgp-vpn1] peer 2002:1::2 as-number 200
[ASBR-PE1-bgp-vpn1] address-family ipv6 unicast
[ASBR-PE1-bgp-ipv6-vpn1] peer 2002:1::2 enable
[ASBR-PE1-bgp-ipv6-vpn1] quit
[ASBR-PE1-bgp-vpn1] quit
[ASBR-PE1-bgp] peer 1.1.1.9 as-number 100
[ASBR-PE1-bgp] peer 1.1.1.9 connect-interface loopback 0
[ASBR-PE1-bgp] address-family vpnv6
[ASBR-PE1-bgp-vpnv6] peer 1.1.1.9 enable
[ASBR-PE1-bgp-vpnv6] quit
[ASBR-PE1-bgp] quit
# Configure ASBR-PE 2.
[ASBR-PE2] bgp 200
[ASBR-PE2-bgp] ip vpn-instance vpn1
[ASBR-PE2-bgp-vpn1] peer 2002:1::1 as-number 100
[ASBR-PE2-bgp-vpn1] address-family ipv6 unicast
[ASBR-PE2-bgp-ipv6-vpn1] peer 2002:1::1 enable
[ASBR-PE2-bgp-ipv6-vpn1] quit
[ASBR-PE2-bgp-vpn1] quit
[ASBR-PE2-bgp] peer 4.4.4.9 as-number 200
[ASBR-PE2-bgp] peer 4.4.4.9 connect-interface loopback 0
[ASBR-PE2-bgp] address-family vpnv6
[ASBR-PE2-bgp-vpnv6] peer 4.4.4.9 enable
[ASBR-PE2-bgp-vpnv6] quit
[ASBR-PE2-bgp] quit
# Configure PE 2.
[PE2] bgp 200
[PE2-bgp] peer 3.3.3.9 as-number 200
[PE2-bgp] peer 3.3.3.9 connect-interface loopback 0
[PE2-bgp] address-family vpnv6
[PE2-bgp-vpnv6] peer 3.3.3.9 enable
[PE2-bgp-vpnv6] quit
[PE2-bgp] quit

```

Verifying the configuration

After the configurations, the CEs can learn the route to each other and can ping each other.

Configuring IPv6 MPLS L3VPN inter-AS option C

Network requirements

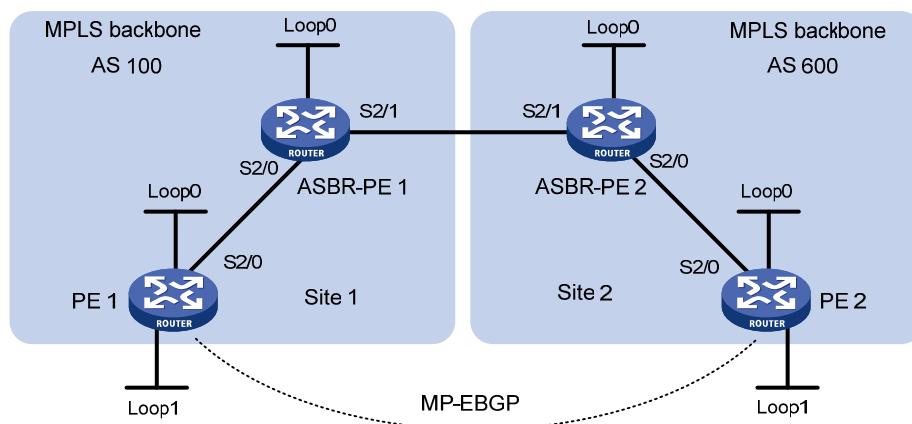
Site 1 and Site 2 belong to the same VPN. Site 1 accesses the network through PE 1 in AS 100 and Site 2 accesses the network through PE 2 in AS 600. PEs in the same AS run IS-IS.

PE 1 and ASBR-PE 1 exchange labeled IPv4 routes by IBGP. PE 2 and ASBR-PE 2 exchange labeled IPv4 routes by IBGP. PE 1 and PE 2 are MP-EBGP peers to exchange VPNv6 routes.

ASBR-PE 1 and ASBR-PE 2 use their respective routing policies and label the routes received from each other.

ASBR-PE 1 and ASBR-PE 2 use EBGP to exchange labeled IPv4 routes.

Figure 65 Network diagram



Device	Interface	IP address	Device	Interface	IP address
PE 1	Loop0	2.2.2.9/32	PE 2	Loop0	5.5.5.9/32
	Loop1	2001:1::1/128		Loop1	2001:1::2/128
	S2/0	1.1.1.2/8		S2/0	9.1.1.2/8
ASBR-PE 1	Loop0	3.3.3.9/32	ASBR-PE 2	Loop0	4.4.4.9/32
	S2/0	1.1.1.1/8		S2/0	9.1.1.1/8
	S2/1	11.0.0.2/8		S2/1	11.0.0.1/8

Configuration procedure

1. Configure PE 1:

Configure IS-IS on PE 1.

```
<PE1> system-view
[PE1] isis 1
[PE1-isis-1] network-entity 10.111.111.111.00
[PE1-isis-1] quit
```

Configure an LSR ID, and enable MPLS and LDP.

```
[PE1] mpls lsr-id 2.2.2.9
[PE1] mpls ldp
[PE1-ldp] quit
```

Configure interface Serial 2/0, and enable IS-IS, MPLS, and LDP on the interface.

```
[PE1] interface serial 2/0
```



```

[PE1-Serial2/0] ip address 1.1.1.2 255.0.0.0
[PE1-Serial2/0] isis enable 1
[PE1-Serial2/0] mpls enable
[PE1-Serial2/0] mpls ldp enable
[PE1-Serial2/0] quit
# Configure interface Loopback 0 and start IS-IS on it.
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 2.2.2.9 32
[PE1-LoopBack0] isis enable 1
[PE1-LoopBack0] quit
# Create VPN instance vpn1, and configure the RD and route target attributes for it.
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 11:11
[PE1-vpn-instance-vpn1] vpn-target 3:3 import-extcommunity
[PE1-vpn-instance-vpn1] vpn-target 3:3 export-extcommunity
[PE1-vpn-instance-vpn1] quit
# Configure interface Loopback 1 and bind the interface to VPN instance vpn1.
[PE1] interface loopback 1
[PE1-LoopBack1] ip binding vpn-instance vpn1
[PE1-LoopBack1] ipv6 address 2001:1::1 128
[PE1-LoopBack1] quit
# Start BGP on PE 1.
[PE1] bgp 100
# Enable the capability to advertise labeled routes to and receive labeled routes from IBGP peer 3.3.3.9.
[PE1-bgp] peer 3.3.3.9 as-number 100
[PE1-bgp] peer 3.3.3.9 connect-interface loopback 0
[PE1-bgp] address-family ipv4 unicast
[PE1-bgp-ipv4] peer 3.3.3.9 enable
[PE1-bgp-ipv4] peer 3.3.3.9 label-route-capability
[PE1-bgp-ipv4] quit
# Configure the maximum hop count from PE 1 to EBGP peer 5.5.5.9 as 10.
[PE1-bgp] peer 5.5.5.9 as-number 600
[PE1-bgp] peer 5.5.5.9 connect-interface loopback 0
[PE1-bgp] peer 5.5.5.9 ebgp-max-hop 10
# Configure peer 5.5.5.9 as a VPNv6 peer.
[PE1-bgp] address-family vpnv6
[PE1-bgp-af-vpnv6] peer 5.5.5.9 enable
[PE1-bgp-af-vpnv6] quit
# Redistribute direct routes to the routing table of vpn1.
[PE1-bgp] ip vpn-instance vpn1
[PE1-bgp-vpn1] address-family ipv6 unicast
[PE1-bgp-ipv6-vpn1] import-route direct
[PE1-bgp-ipv6-vpn1] quit
[PE1-bgp-vpn1] quit
[PE1-bgp] quit

```

2. Configure ASBR-PE 1:

Start IS-IS on ASBR-PE 1.

```
<ASBR-PE1> system-view
[ASBR-PE1] isis 1
[ASBR-PE1-isis-1] network-entity 10.222.222.222.00
[ASBR-PE1-isis-1] quit
```

Configure an LSR ID, and enable MPLS and LDP.

```
[ASBR-PE1] mpls lsr-id 3.3.3.9
[ASBR-PE1] mpls ldp
[ASBR-PE1-ldp] quit
```

Configure interface Serial 2/0, and enable IS-IS, MPLS, and LDP on the interface.

```
[ASBR-PE1] interface serial 2/0
[ASBR-PE1-Serial2/0] ip address 1.1.1.1 255.0.0.0
[ASBR-PE1-Serial2/0] isis enable 1
[ASBR-PE1-Serial2/0] mpls enable
[ASBR-PE1-Serial2/0] mpls ldp enable
[ASBR-PE1-Serial2/0] quit
```

Configure interface Serial 2/1, and enable MPLS on it.

```
[ASBR-PE1] interface serial 2/1
[ASBR-PE1-Serial2/1] ip address 11.0.0.2 255.0.0.0
[ASBR-PE1-Serial2/1] mpls
[ASBR-PE1-Serial2/1] quit
```

Configure interface Loopback 0, and start IS-IS on it.

```
[ASBR-PE1] interface loopback 0
[ASBR-PE1-LoopBack0] ip address 3.3.3.9 32
[ASBR-PE1-LoopBack0] isis enable 1
[ASBR-PE1-LoopBack0] quit
```

Create routing policies.

```
[ASBR-PE1] route-policy policy1 permit node 1
[ASBR-PE1-route-policy-policy1-1] apply mpls-label
[ASBR-PE1-route-policy-policy1-1] quit
[ASBR-PE1] route-policy policy2 permit node 1
[ASBR-PE1-route-policy-policy2-1] if-match mpls-label
[ASBR-PE1-route-policy-policy2-1] apply mpls-label
[ASBR-PE1-route-policy-policy2-1] quit
```

Start BGP on ASBR-PE 1, and apply routing policy **policy2** to routes advertised to IBGP peer 2.2.2.9.

```
[ASBR-PE1] bgp 100
[ASBR-PE1-bgp] peer 2.2.2.9 as-number 100
[ASBR-PE1-bgp] peer 2.2.2.9 connect-interface loopback 0
[ASBR-PE1-bgp] address-family ipv4 unicast
[ASBR-PE1-bgp-ipv4] peer 2.2.2.9 enable
[ASBR-PE1-bgp-ipv4] peer 2.2.2.9 route-policy policy2 export
```

Enable the capability to advertise labeled routes to and receive labeled routes from IBGP peer 2.2.2.9.

```
[ASBR-PE1-bgp-ipv4] peer 2.2.2.9 label-route-capability
```

```

# Redistribute routes from IS-IS process 1
[ASBR-PE1-bgp-ipv4] import-route isis 1
[ASBR-PE1-bgp-ipv4] quit

# Apply routing policy policy1 to routes advertised to EBGp peer 11.0.0.1.
[ASBR-PE1-bgp] peer 11.0.0.1 as-number 600
[ASBR-PE1-bgp] address-family ipv4 unicast
[ASBR-PE1-bgp-ipv4] peer 11.0.0.1 enable
[ASBR-PE1-bgp-ipv4] peer 11.0.0.1 route-policy policy1 export

# Enable the capability to advertise labeled routes to and receive labeled routes from EBGp peer 11.0.0.1.
[ASBR-PE1-bgp-ipv4] peer 11.0.0.1 label-route-capability
[ASBR-PE1-bgp-ipv4] quit
[ASBR-PE1-bgp] quit

```

3. Configure ASBR-PE 2:

```

# Start IS-IS on ASBR-PE 2.
<ASBR-PE2> system-view
[ASBR-PE2] isis 1
[ASBR-PE2-isis-1] network-entity 10.333.333.333.000
[ASBR-PE2-isis-1] quit

# Configure an LSR ID, and enable MPLS and LDP.
[ASBR-PE2] mpls lsr-id 4.4.4.9
[ASBR-PE2] mpls ldp
[ASBR-PE2-ldp] quit

# Configure interface Serial 2/0, and enable IS-IS, MPLS, and LDP on the interface.
[ASBR-PE2] interface serial 2/0
[ASBR-PE2-Serial2/0] ip address 9.1.1.1 255.0.0.0
[ASBR-PE2-Serial2/0] isis enable 1
[ASBR-PE2-Serial2/0] mpls enable
[ASBR-PE2-Serial2/0] mpls ldp enable
[ASBR-PE2-Serial2/0] quit

# Configure interface Loopback 0, and start IS-IS on it.
[ASBR-PE2] interface loopback 0
[ASBR-PE2-LoopBack0] ip address 4.4.4.9 32
[ASBR-PE2-LoopBack0] isis enable 1
[ASBR-PE2-LoopBack0] quit

# Configure interface Serial 2/1, and enable MPLS on it.
[ASBR-PE2] interface serial 2/1
[ASBR-PE2-Serial2/1] ip address 11.0.0.1 255.0.0.0
[ASBR-PE2-Serial2/1] mpls enable
[ASBR-PE2-Serial2/1] quit

# Create routing policies.
[ASBR-PE2] route-policy policy1 permit node 1
[ASBR-PE2-route-policy-policy1-1] apply mpls-label
[ASBR-PE2-route-policy-policy1-1] quit
[ASBR-PE2] route-policy policy2 permit node 1
[ASBR-PE2-route-policy-policy2-1] if-match mpls-label

```

```

[ASBR-PE2-route-policy-policy2-1] apply mpls-label
[ASBR-PE2-route-policy-policy2-1] quit
# Start BGP on ASBR-PE 2, and enable the capability to advertise labeled routes to and receive
labeled routes from IBGP peer 5.5.5.9.
[ASBR-PE2] bgp 600
[ASBR-PE2-bgp] peer 5.5.5.9 as-number 600
[ASBR-PE2-bgp] peer 5.5.5.9 connect-interface loopback 0
[ASBR-PE2-bgp] address-family ipv4 unicast
[ASBR-PE2-bgp-ipv4] peer 5.5.5.9 enable
[ASBR-PE2-bgp-ipv4] peer 5.5.5.9 label-route-capability
# Apply routing policy policy2 to routes advertised to IBGP peer 5.5.5.9.
[ASBR-PE2-bgp-ipv4] peer 5.5.5.9 route-policy policy2 export
# Redistribute routes from IS-IS process 1.
[ASBR-PE2-bgp-ipv4] import-route isis 1
[ASBR-PE2-bgp-ipv4] quit
# Apply routing policy policy1 to routes advertised to EBGP peer 11.0.0.2.
[ASBR-PE2-bgp] peer 11.0.0.2 as-number 100
[ASBR-PE2-bgp] address-family ipv4 unicast
[ASBR-PE2-bgp-ipv4] peer 11.0.0.2 enable
[ASBR-PE2-bgp-ipv4] peer 11.0.0.2 route-policy policy1 export
# Enable the capability to advertise labeled routes to and receive labeled routes from EBGP peer
11.0.0.2.
[ASBR-PE2-bgp-ipv4] peer 11.0.0.2 label-route-capability
[ASBR-PE2-bgp-ipv4] quit
[ASBR-PE2-bgp] quit

```

4. Configure PE 2:

```

# Start IS-IS on PE 2.
<PE2> system-view
[PE2] isis 1
[PE2-isis-1] network-entity 10.444.444.444.444.00
[PE2-isis-1] quit
# Configure an LSR ID, and enable MPLS and LDP.
[PE2] mpls lsr-id 5.5.5.9
[PE2] mpls ldp
[PE2-ldp] quit
# Configure interface Serial 2/0, and enable IS-IS, MPLS, and LDP on the interface.
[PE2] interface serial 2/0
[PE2-Serial2/0] ip address 9.1.1.2 255.0.0.0
[PE2-Serial2/0] isis enable 1
[PE2-Serial2/0] mpls enable
[PE2-Serial2/0] mpls ldp enable
[PE2-Serial2/0] quit
# Configure interface Loopback 0, and start IS-IS on it.
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 5.5.5.9 32
[PE2-LoopBack0] isis enable 1

```

```

[PE2-LoopBack0] quit
# Create VPN instance vpn1, and configure the RD and route target attributes for it.
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance-vpn1] route-distinguisher 11:11
[PE2-vpn-instance-vpn1] vpn-target 3:3 import-extcommunity
[PE2-vpn-instance-vpn1] vpn-target 3:3 export-extcommunity
[PE2-vpn-instance-vpn1] quit
# Configure interface Loopback 1, and bind the interface to VPN instance vpn1.
[PE2] interface loopback 1
[PE2-LoopBack1] ip binding vpn-instance vpn1
[PE2-LoopBack1] ipv6 address 2001:1::2 128
[PE2-LoopBack1] quit
# Start BGP.
[PE2] bgp 600
# Enable the capability to advertise labeled routes to and receive labeled routes from IBGP peer
4.4.4.9.
[PE2-bgp] peer 4.4.4.9 as-number 600
[PE2-bgp] peer 4.4.4.9 connect-interface loopback 0
[PE2-bgp] address-family ipv4 unicast
[PE2-bgp-ipv4] peer 4.4.4.9 enable
[PE2-bgp-ipv4] peer 4.4.4.9 label-route-capability
[PE2-bgp-ipv4] quit
# Configure the maximum hop count from PE 2 to EBGP peer 2.2.2.9 as 10.
[PE2-bgp] peer 2.2.2.9 as-number 100
[PE2-bgp] peer 2.2.2.9 connect-interface loopback 0
[PE2-bgp] peer 2.2.2.9 ebgp-max-hop 10
# Configure peer 2.2.2.9 as a VPNv6 peer.
[PE2-bgp] address-family vpnv6
[PE2-bgp-af-vpnv6] peer 2.2.2.9 enable
[PE2-bgp-af-vpnv6] quit
# Redistribute direct routes to the routing table of VPN instance vpn1.
[PE2-bgp] ip vpn-instance vpn1
[PE2-bgp-vpn1] address-family ipv6 unicast
[PE2-bgp-ipv6-vpn1] import-route direct
[PE2-bgp-ipv6-vpn1] quit
[PE2-bgp-vpn1] quit
[PE2-bgp] quit

```

Verifying the configuration

PE 1 and PE 2 can ping each other. Take PE 1 as an example:

```

[PE1] ping ipv6 -a 2001:1::1 -vpn-instance vpn1 2001:1::2
Ping6(56 bytes) 2001:1::1 --> 2001:1::2, press escape sequence to break
56 bytes from 2001:1::2, icmp_seq=0 hlim=64 time=6.000 ms
56 bytes from 2001:1::2, icmp_seq=1 hlim=64 time=1.000 ms
56 bytes from 2001:1::2, icmp_seq=2 hlim=64 time=1.000 ms
56 bytes from 2001:1::2, icmp_seq=3 hlim=64 time=3.000 ms
56 bytes from 2001:1::2, icmp_seq=4 hlim=64 time=3.000 ms

```

```

--- Ping6 statistics for 2001:1::2 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 1.000/2.800/6.000/1.833 ms

```

Configuring IPv6 MPLS L3VPN carrier's carrier

Network requirements

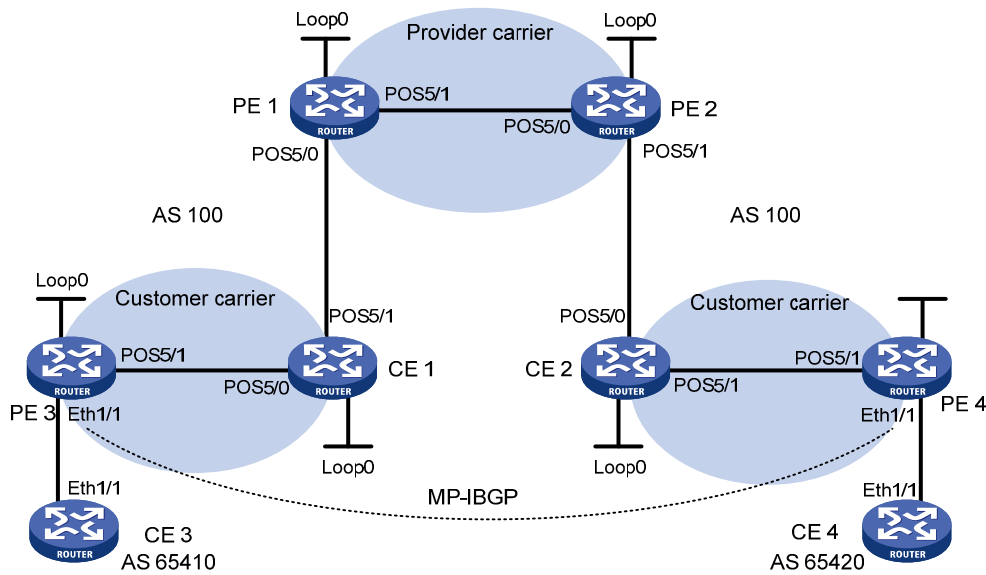
Configure carrier's carrier for the scenario shown in [Figure 66](#). In this scenario:

- PE 1 and PE 2 are the provider carrier's PE routers. They provide VPN services to the customer carrier.
- CE 1 and CE 2 are the customer carrier's routers. They are connected to the provider carrier's backbone as CE routers.
- PE 3 and PE 4 are the customer carrier's PE routers. They provide IPv6 MPLS L3VPN services to end customers.
- CE 3 and CE 4 are customers of the customer carrier.

The key to the carrier's carrier deployment is to configure exchange of two kinds of routes:

- Exchange of the customer carrier's internal routes on the provider carrier's backbone.
- Exchange of the end customers' internal routes between PE 3 and PE 4, the PEs of the customer carrier. In this process, an MP-IBGP peer relationship must be established between PE 3 and PE 4.

Figure 66 Network diagram



Device	Interface	IP address	Device	Interface	IP address
CE 3	Eth1/1	2001:1::1/96	CE 4	Eth1/1	2001:2::1/96
PE 3	Loop0	1.1.1.9/32	PE 4	Loop0	6.6.6.9/32
	Eth1/1	2001:1::2/96		Eth1/1	2001:2::2/96
	POS5/1	10.1.1.1/24		POS5/1	20.1.1.2/24
CE 1	Loop0	2.2.2.9/32	CE 2	Loop0	5.5.5.9/32
	POS5/0	10.1.1.2/24		POS5/0	21.1.1.2/24
	POS5/1	11.1.1.1/24		POS5/1	20.1.1.1/24

PE 1	Loop0	3.3.3.9/32	PE 2	Loop0	4.4.4.9/32
	POS5/0	11.1.1.2/24		POS5/0	30.1.1.2/24
	POS5/1	30.1.1.1/24		POS5/1	21.1.1.1/24

Configuration procedure

1. Configure MPLS L3VPN on the provider carrier backbone. Start IS-IS as the IGP, enable LDP on PE 1 and PE 2, and establish an MP-IBGP peer relationship between the PEs:

Configure PE 1.

```
<PE1> system-view
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 3.3.3.9 32
[PE1-LoopBack0] quit
[PE1] mpls lsr-id 3.3.3.9
[PE1] mpls ldp
[PE1-ldp] quit
[PE1] isis 1
[PE1-isis-1] network-entity 10.0000.0000.0000.0004.00
[PE1-isis-1] quit
[PE1] interface loopback 0
[PE1-LoopBack0] isis enable 1
[PE1-LoopBack0] quit
[PE1] interface pos 5/1
[PE1-POS5/1] ip address 30.1.1.1 24
[PE1-POS5/1] isis enable 1
[PE1-POS5/1] mpls enable
[PE1-POS5/1] mpls ldp enable
[PE1-POS5/1] mpls ldp transport-address interface
[PE1-POS5/1] quit
[PE1] bgp 100
[PE1-bgp] peer 4.4.4.9 as-number 100
[PE1-bgp] peer 4.4.4.9 connect-interface loopback 0
[PE1-bgp] address-family vpnv4
[PE1-bgp-vpnv4] peer 4.4.4.9 enable
[PE1-bgp-vpnv4] quit
[PE1-bgp] quit
```

Configure PE 2 in the same way that PE 1 is configured. (Details not shown.)

After you complete the configurations, execute the **display mpls ldp peer** command on PE 1 or PE 2 to see that the LDP session has been established. Execute the **display bgp peer vpnv4** command and you can see that the BGP peer relationship has been established and has reached the Established state. Execute the **display isis peer** command to see that an IS-IS neighbor relationship has been set up. Take PE 1 as an example:

```
[PE1] display mpls ldp peer
Total number of peers: 1
Peer LDP ID          State          LAM  Role    GR   MD5  KA Sent/Rcvd
4.4.4.9:0            Operational    DU   Active  Off  Off  8/8
[PE1] display bgp peer vpnv4
```

```

BGP local router ID: 3.3.3.9
Local AS number: 100
Total number of peers: 1                Peers in established state: 1

Peer                AS  MsgRcvd  MsgSent  OutQ  PrefRcv  Up/Down  State
4.4.4.9            100      3         6       0       0 00:00:32 Established
[PE1] display isis peer

```

```

Peer information for ISIS(1)
-----

```

```

System Id: 0000.0000.0005
Interface: POS5/1                Circuit Id: 0000.0000.0005.02
State: Up      HoldTime: 8s      Type: L1(L1L2)      PRI: 64

```

```

System Id: 0000.0000.0005
Interface: POS5/1                Circuit Id: 0000.0000.0005.02
State: Up      HoldTime: 8s      Type: L2(L1L2)      PRI: 64

```

2. Configure the customer carrier network. Start IS-IS as the IGP, and enable LDP between PE 3 and CE 1, and between PE 4 and CE 2:

Configure PE 3.

```

<PE3> system-view
[PE3] interface loopback 0
[PE3-LoopBack0] ip address 1.1.1.9 32
[PE3-LoopBack0] quit
[PE3] mpls lsr-id 1.1.1.9
[PE3] mpls ldp
[PE3-ldp] quit
[PE3] isis 2
[PE3-isis-2] network-entity 10.0000.0000.0000.0001.00
[PE3-isis-2] quit
[PE3] interface loopback 0
[PE3-LoopBack0] isis enable 2
[PE3-LoopBack0] quit
[PE3] interface pos 5/1
[PE3-POS5/1] ip address 10.1.1.1 24
[PE3-POS5/1] isis enable 2
[PE3-POS5/1] mpls enable
[PE3-POS5/1] mpls ldp enable
[PE3-POS5/1] mpls ldp transport-address interface
[PE3-POS5/1] quit

```

Configure CE 1.

```

<CE1> system-view
[CE1] interface loopback 0
[CE1-LoopBack0] ip address 2.2.2.9 32
[CE1-LoopBack0] quit
[CE1] mpls lsr-id 2.2.2.9
[CE1] mpls ldp

```



```

[CE1-ldp] quit
[CE1] isis 2
[CE1-isis-2] network-entity 10.0000.0000.0000.0002.00
[CE1-isis-2] quit
[CE1] interface loopback 0
[CE1-LoopBack0] isis enable 2
[CE1-LoopBack0] quit
[CE1] interface POS 5/0
[CE1-POS5/0] ip address 10.1.1.2 24
[CE1-POS5/0] isis enable 2
[CE1-POS5/0] mpls enable
[CE1-POS5/0] mpls ldp enable
[CE1-POS5/0] mpls ldp transport-address interface
[CE1-POS5/0] quit

```

After the configurations, PE 3 and CE 1 can establish an LDP session and IS-IS neighbor relationship between them.

Configure PE 4 and CE 2 in the same way that PE 3 and CE 1 are configured. (Details not shown.)

3. Connect the customer carrier to the provider carrier:

Configure PE 1.

```

[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 200:1
[PE1-vpn-instance-vpn1] vpn-target 1:1
[PE1-vpn-instance-vpn1] quit
[PE1] mpls ldp
[PE1-ldp] vpn-instance vpn1
[PE1-ldp-vpn-instance-vpn1] quit
[PE1-ldp] quit
[PE1] isis 2 vpn-instance vpn1
[PE1-isis-2] network-entity 10.0000.0000.0000.0003.00
[PE1-isis-2] import-route bgp allow-ibgp
[PE1-isis-2] quit
[PE1] interface pos 5/0
[PE1-POS5/0] ip binding vpn-instance vpn1
[PE1-POS5/0] ip address 11.1.1.2 24
[PE1-POS5/0] isis enable 2
[PE1-POS5/0] mpls enable
[PE1-POS5/0] mpls ldp enable
[PE1-POS5/0] mpls ldp transport-address interface
[PE1-POS5/0] quit
[PE1] bgp 100
[PE1-bgp] ip vpn-instance vpn1
[PE1-bgp-vpn1] address-family ipv4 unicast
[PE1-bgp-ipv4-vpn1] import isis 2
[PE1-bgp-ipv4-vpn1] quit
[PE1-bgp-vpn1] quit
[PE1-bgp] quit

```

Configure CE 1.

```
[CE1] interface pos 5/1
[CE1-POS5/1] ip address 11.1.1.1 24
[CE1-POS5/1] isis enable 2
[CE1-POS5/1] mpls enable
[CE1-POS5/1] mpls ldp enable
[CE1-POS5/1] mpls ldp transport-address interface
[CE1-POS5/1] quit
```

After the configurations, PE 1 and CE 1 can establish an LDP session and IS-IS neighbor relationship between them.

Configure PE 2 and CE 2 in the same way that PE 1 and CE 1 are configured. (Details not shown.)

4. Connect end customers to the customer carrier:

Configure CE 3.

```
<CE3> system-view
[CE3] interface ethernet 1/1
[CE3-Ethernet1/1] ipv6 address 2001:1::1 96
[CE3-Ethernet1/1] quit
[CE3] bgp 65410
[CE3-bgp] peer 2001:1::2 as-number 100
[CE3-bgp] address-family ipv6
[CE3-bgp-ipv6] peer 2001:1::2 enable
[CE3-bgp-ipv6] import-route direct
[CE3-bgp-ipv6] quit
[CE3-bgp] quit
```

Configure PE 3.

```
[PE3] ip vpn-instance vpn1
[PE3-vpn-instance-vpn1] route-distinguisher 100:1
[PE3-vpn-instance-vpn1] vpn-target 1:1
[PE3-vpn-instance-vpn1] quit
[PE3] interface ethernet 1/1
[PE3-Ethernet1/1] ip binding vpn-instance vpn1
[PE3-Ethernet1/1] ipv6 address 2001:1::2 96
[PE3-Ethernet1/1] quit
[PE3] bgp 100
[PE3-bgp] ip vpn-instance vpn1
[PE3-bgp-vpn1] peer 2001:1::1 as-number 65410
[PE3-bgp-vpn1] address-family ipv6 unicast
[PE3-bgp-ipv6-vpn1] peer 2001:1::1 enable
[PE3-bgp-ipv6-vpn1] import-route direct
[PE3-bgp-ipv6-vpn1] quit
[PE3-bgp-vpn1] quit
[PE3-bgp] quit
```

Configure PE 4 and CE 4 in the same way that PE 3 and CE 3 are configured. (Details not shown.)

5. Configure an MP-IBGP peer relationship between the PEs of the customer carrier to exchange the VPN routes of the end customers:

Configure PE 3.

```
[PE3] bgp 100
[PE3-bgp] peer 6.6.6.9 as-number 100
[PE3-bgp] peer 6.6.6.9 connect-interface loopback 0
[PE3-bgp] address-family vpnv6
[PE3-bgp-af-vpnv6] peer 6.6.6.9 enable
[PE3-bgp-af-vpnv6] quit
[PE3-bgp] quit
```

Configure PE 4 in the same way that PE 3 is configured. (Details not shown.)

Verifying the configuration

Execute the **display ip routing-table** command on PE 1 and PE 2. The output shows that only routes of the provider carrier network are present in the public network routing table of PE 1 and PE 2. Take PE 1 as an example:

```
[PE1] display ip routing-table
```

Routing Tables: Public

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
3.3.3.9/32	Direct	0	0	127.0.0.1	InLoop0
4.4.4.9/32	ISIS	15	10	30.1.1.2	POS5/1
30.1.1.0/24	Direct	0	0	30.1.1.1	POS5/1
30.1.1.1/32	Direct	0	0	127.0.0.1	InLoop0
30.1.1.2/32	Direct	0	0	30.1.1.2	POS5/1
127.0.0.0/8	Direct	0	0	127.0.0.1	InLoop0
127.0.0.1/32	Direct	0	0	127.0.0.1	InLoop0

Execute the **display ip routing-table vpn-instance** command on PE 1 and PE 2. The output shows that the internal routes of the customer carrier network are present in the VPN routing tables. Take PE 1 as an example:

```
[PE1] display ip routing-table vpn-instance vpn1
```

Routing Tables: vpn1

Destination/Mask	Proto	Pre	Cost	NextHop	Interface
1.1.1.9/32	ISIS	15	20	11.1.1.1	POS5/0
2.2.2.9/32	ISIS	15	10	11.1.1.1	POS5/0
5.5.5.9/32	BGP	255	0	4.4.4.9	NULL0
6.6.6.9/32	BGP	255	0	4.4.4.9	NULL0
10.1.1.0/24	ISIS	15	20	11.1.1.1	POS5/0
11.1.1.0/24	Direct	0	0	11.1.1.1	POS5/0
11.1.1.1/32	Direct	0	0	127.0.0.1	InLoop0
11.1.1.2/32	Direct	0	0	11.1.1.2	POS5/0
20.1.1.0/24	BGP	255	0	4.4.4.9	NULL0
21.1.1.0/24	BGP	255	0	4.4.4.9	NULL0
21.1.1.2/32	BGP	255	0	4.4.4.9	NULL0

Execute the **display ip routing-table** command on CE 1 and CE 2. The output shows that the internal routes of the customer carrier network are present in the public network routing table. Take CE 1 as an example:

```
[CE1] display ip routing-table
```

Routing Tables: Public

```

Destinations : 16          Routes : 16
Destination/Mask  Proto  Pre  Cost  NextHop      Interface
1.1.1.9/32       ISIS   15   10    10.1.1.2     POS5/0
2.2.2.9/32       Direct 0    0     127.0.0.1    InLoop0
5.5.5.9/32       ISIS   15   74    11.1.1.2     POS5/1
6.6.6.9/32       ISIS   15   74    11.1.1.2     POS5/1
10.1.1.0/24      Direct 0    0     10.1.1.2     POS5/0
10.1.1.1/32      Direct 0    0     10.1.1.1     POS5/0
10.1.1.2/32      Direct 0    0     127.0.0.1    InLoop0
11.1.1.0/24      Direct 0    0     11.1.1.1     POS5/1
11.1.1.1/32      Direct 0    0     127.0.0.1    InLoop0
11.1.1.2/32      Direct 0    0     11.1.1.2     POS5/1
20.1.1.0/24      ISIS   15   74    11.1.1.2     POS5/1
21.1.1.0/24      ISIS   15   74    11.1.1.2     POS5/1
21.1.1.2/32      ISIS   15   74    11.1.1.2     POS5/1
127.0.0.0/8      Direct 0    0     127.0.0.1    InLoop0
127.0.0.1/32     Direct 0    0     127.0.0.1    InLoop0

```

Execute the **display ip routing-table** command on PE 3 and PE 4. The output shows that the internal routes of the customer carrier network are present in the public network routing tables. Take PE 3 as an example:

```

[PE3] display ip routing-table
Routing Tables: Public
Destinations : 11          Routes : 11
Destination/Mask  Proto  Pre  Cost  NextHop      Interface
1.1.1.9/32       Direct 0    0     127.0.0.1    InLoop0
2.2.2.9/32       ISIS   15   10    10.1.1.2     POS5/1
5.5.5.9/32       ISIS   15   84    10.1.1.2     POS5/1
6.6.6.9/32       ISIS   15   84    10.1.1.2     POS5/1
10.1.1.0/24      Direct 0    0     10.1.1.1     POS5/1
10.1.1.1/32      Direct 0    0     127.0.0.1    InLoop0
10.1.1.2/32      Direct 0    0     10.1.1.2     POS5/1
11.1.1.0/24      ISIS   15   20    10.1.1.2     POS5/1
20.1.1.0/24      ISIS   15   84    10.1.1.2     POS5/1
21.1.1.0/24      ISIS   15   84    10.1.1.2     POS5/1
21.1.1.2/32      ISIS   15   84    10.1.1.2     POS5/1
127.0.0.0/8      Direct 0    0     127.0.0.1    InLoop0
127.0.0.1/32     Direct 0    0     127.0.0.1    InLoop0

```

Execute the **display ipv6 routing-table vpn-instance** command on PE 3 and PE 4. The output shows that the VPN routing table has the remote VPN route. Take PE 3 as an example:

```

[PE3] display ipv6 routing-table vpn-instance vpn1

Destinations : 6 Routes : 6

Destination: ::1/128          Protocol : Direct
NextHop      : ::1           Preference: 0
Interface    : InLoop0        Cost      : 0

Destination: 2001:1::/96     Protocol : Direct

```

```

NextHop      : ::                               Preference: 0
Interface    : Eth1/1                           Cost       : 0

Destination: 2001:1::2/128                       Protocol   : Direct
NextHop      : ::1                               Preference: 0
Interface    : InLoop0                           Cost       : 0

Destination: 2001:2::/96                           Protocol   : BGP4+
NextHop      : ::FFFF:606:609                     Preference: 0
Interface    : NULL0                               Cost       : 0

Destination: FE80::/10                             Protocol   : Direct
NextHop      : ::                                 Preference: 0
Interface    : NULL0                               Cost       : 0

Destination: FF00::/8                              Protocol   : Direct
NextHop      : ::                                 Preference: 0
Interface    : NULL0                               Cost       : 0

# PE 3 and PE 4 can ping each other.
# CE 3 and CE 4 can ping each other.

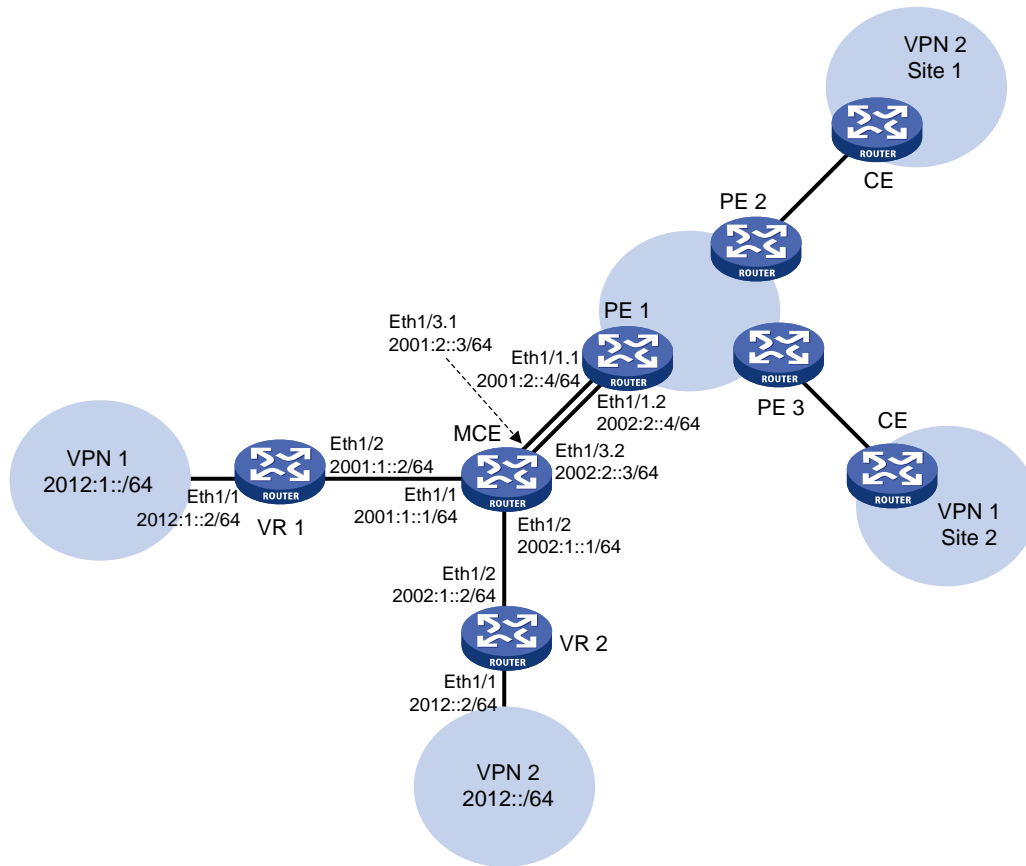
```

Configuring IPv6 MCE

Network requirements

As shown in [Figure 67](#), VPN 2 runs RIPng. Configure the MCE device to separate routes from different VPNs and advertise the VPN routes to PE 1 through OSPFv3.

Figure 67 Network diagram



Configuration procedure

Assume that the system name of the MCE device is MCE, the system names of the edge routers of VPN 1 and VPN 2 are VR1 and VR2, and the system name of PE 1 is PE1.

1. Configure VPN instances on the MCE and PE 1:

On MCE, configure VPN instances **vpn1** and **vpn2**, and specify an RD and route targets for each VPN instance.

```
<MCE> system-view
[MCE] ip vpn-instance vpn1
[MCE-vpn-instance-vpn1] route-distinguisher 10:1
[MCE-vpn-instance-vpn1] vpn-target 10:1
[MCE-vpn-instance-vpn1] quit
[MCE] ip vpn-instance vpn2
[MCE-vpn-instance-vpn2] route-distinguisher 20:1
[MCE-vpn-instance-vpn2] vpn-target 20:1
[MCE-vpn-instance-vpn2] quit
```

Bind interface Ethernet 1/1 with VPN instance **vpn1**, and configure an IPv6 address for the interface.

```
[MCE] interface ethernet 1/1
[MCE-Ethernet1/1] ip binding vpn-instance vpn1
[MCE-Ethernet1/1] ipv6 address 2001:1::1 64
[MCE-Ethernet1/1] quit
```

Bind interface Ethernet 1/2 with VPN instance **vpn2**, and configure an IPv6 address for the interface.

```
[MCE] interface ethernet 1/2
[MCE-Ethernet1/2] ip binding vpn-instance vpn2
[MCE-Ethernet1/2] ipv6 address 2002:1::1 64
[MCE-Ethernet1/2] quit
```

On PE 1, configure VPN instances **vpn1** and **vpn2**, and specify an RD and route targets for each VPN instance.

```
<PE1> system-view
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 30:1
[PE1-vpn-instance-vpn1] vpn-target 10:1
[PE1-vpn-instance-vpn1] quit
[PE1] ip vpn-instance vpn2
[PE1-vpn-instance-vpn2] route-distinguisher 40:1
[PE1-vpn-instance-vpn2] vpn-target 20:1
[PE1-vpn-instance-vpn2] quit
```

2. Configure routing between the MCE and VPN sites:

The MCE is connected to VPN 1 directly, and no routing protocol is enabled in VPN 1. Therefore, you can configure IPv6 static routes.

On VR 1, assign IPv6 address 2001:1::2/64 to the interface connected to the MCE and 2012:1::2/64 to the interface connected to VPN 1. (Details not shown.)

On VR 1, configure a default route with the next hop as 2001:1::1.

```
<VR1> system-view
[VR1] ipv6 route-static :: 0 2001:1::1
```

On the MCE, configure an IPv6 static route to 2012:1::/64, specify the next hop as 2001:1::2, and bind the static route with VPN instance **vpn1**.

```
[MCE] ipv6 route-static vpn-instance vpn1 2012:1:: 64 2001:1::2
```

Run RIPng in VPN 2. Configure RIPng process 20 for the VPN instance **vpn2** on the MCE, so that the MCE can learn the routes of VPN 2 and add them to the routing table of the VPN instance **vpn2**.

```
[MCE] ripng 20 vpn-instance vpn2
```

Advertise subnet 2002:1::/64.

```
[MCE] interface ethernet 1/2
[MCE-Ethernet1/2] ripng 20 enable
[MCE-Ethernet1/2] quit
```

On VR 2, assign IPv6 address 2002:1::2/64 to the interface connected to the MCE. (Details not shown.)

On VR 2, configure RIPng and advertise subnets 2012::/64 and 2002:1::/64.

```
<VR2> system-view
[VR2] ripng 20
[VR2-ripng-20] quit
[VR2] interface ethernet 1/1
[VR2-Ethernet1/1] ripng 20 enable
[VR2-Ethernet1/1] quit
[VR2] interface ethernet 1/2
[VR2-Ethernet1/2] ripng 20 enable
```

```
[VR2-Ethernet1/2] quit
```

```
# On the MCE, display the routing tables of the VPN instances vpn1 and vpn2.
```

```
[MCE] display ipv6 routing-table vpn-instance vpn1
```

```
Destinations : 6 Routes : 6
```

```
Destination: ::1/128                Protocol : Direct
NextHop    : ::1                    Preference: 0
Interface  : InLoop0                Cost      : 0
```

```
Destination: 2001:1::/64             Protocol : Direct
NextHop    : ::                     Preference: 0
Interface  : Eth1/1                 Cost      : 0
```

```
Destination: 2001:1::1/128           Protocol : Direct
NextHop    : ::1                    Preference: 0
Interface  : InLoop0                Cost      : 0
```

```
Destination: 2012:1::/64             Protocol : Static
NextHop    : 2001:1::2              Preference: 60
Interface  : Eth1/1                 Cost      : 0
```

```
Destination: FE80::/10               Protocol : Direct
NextHop    : ::                     Preference: 0
Interface  : NULL0                  Cost      : 0
```

```
Destination: FF00::/8                Protocol : Direct
NextHop    : ::                     Preference: 0
Interface  : NULL0                  Cost      : 0
```

```
[MCE] display ipv6 routing-table vpn-instance vpn2
```

```
Destinations : 6 Routes : 6
```

```
Destination: ::1/128                Protocol : Direct
NextHop    : ::1                    Preference: 0
Interface  : InLoop0                Cost      : 0
```

```
Destination: 2002:1::/64             Protocol : Direct
NextHop    : ::                     Preference: 0
Interface  : Eth1/2                 Cost      : 0
```

```
Destination: 2002:1::1/128           Protocol : Direct
NextHop    : ::1                    Preference: 0
Interface  : InLoop0                Cost      : 0
```

```
Destination: 2012::/64               Protocol : RIPng
NextHop    : FE80::20C:29FF:FE40:701 Preference: 100
Interface  : Eth1/2                 Cost      : 1
```



```

Destination: FE80::/10                                Protocol : Direct
NextHop      : ::                                     Preference: 0
Interface    : NULL0                                  Cost      : 0

```

```

Destination: FF00::/8                                Protocol : Direct
NextHop      : ::                                     Preference: 0
Interface    : NULL0                                  Cost      : 0

```

The output shows that the MCE has learned the private route of VPN 2 through RIPng. The MCE maintains the routes of VPN 1 and VPN 2 in two different routing tables. In this way, routes from different VPNs are separated.

3. Configure routing between the MCE and PE 1:

The MCE is connected to PE 1 through subinterfaces. On the MCE, bind subinterface Ethernet 1/3.1 with the VPN instance **vpn1**, configure the subinterface to terminate VLAN 10, and configure an IPv6 address for the subinterface.

```

[MCE] interface ethernet 1/3.1
[MCE-Ethernet1/3.1] ip binding vpn-instance vpn1
[MCE-Ethernet1/3.1] vlan-type dot1q vid 10
[MCE-Ethernet1/3.1] ipv6 address 2001:2::3 64
[MCE-Ethernet1/3.1] quit

```

On the MCE, bind subinterface Ethernet 1/3.2 with the VPN instance **vpn2**, configure the subinterface to terminate VLAN 20, and configure an IPv6 address for the subinterface.

```

[MCE] interface ethernet 1/3.2
[MCE-Ethernet1/3.2] ip binding vpn-instance vpn2
[MCE-Ethernet1/3.2] vlan-type dot1q vid 20
[MCE-Ethernet1/3.2] ipv6 address 2002:2::3 64
[MCE-Ethernet1/3.2] quit

```

On PE 1, bind subinterface Ethernet 1/1.1 with the VPN instance **vpn1**, configure the subinterface to terminate VLAN 10, and configure an IPv6 address for the subinterface.

```

[PE1] interface ethernet 1/1.1
[PE1-Ethernet1/1.1] ip binding vpn-instance vpn1
[PE1-Ethernet1/1.1] vlan-type dot1q vid 10
[PE1-Ethernet1/1.1] ipv6 address 2001:2::4 64
[PE1-Ethernet1/1.1] quit

```

On PE 1, bind subinterface Ethernet 1/1.2 with the VPN instance **vpn2**, configure the subinterface to terminate VLAN 20, and configure an IPv6 address for the subinterface.

```

[PE1] interface ethernet 1/1.2
[PE1-Ethernet1/1.2] ip binding vpn-instance vpn2
[PE1-Ethernet1/1.2] vlan-type dot1q vid 20
[PE1-Ethernet1/1.2] ipv6 address 2002:2::4 64
[PE1-Ethernet1/1.2] quit

```

Configure the IP address of the interface Loopback0 as 101.101.10.1 for the MCE and as 100.100.10.1 for PE 1. Specify the loopback interface address as the router ID for the MCE and PE 1. (Details not shown.)

Enable OSPFv3 process 10 on the MCE, bind the process to VPN instance **vpn1**, and redistribute the IPv6 static route of VPN 1.

```

[MCE] ospfv3 10 vpn-instance vpn1

```

```

[MCE-ospf-10] router-id 101.101.10.1
[MCE-ospf-10] import-route static
[MCE-ospf-10] quit
# Enable OSPFv3 on interface Ethernet 1/3.1.
[MCE] interface ethernet 1/3.1
[MCE-Ethernet1/3.1] ospfv3 10 area 0.0.0.0
[MCE-Ethernet1/3.1] quit
# On PE 1, enable OSPFv3 process 10 and bind it to VPN instance vpn1.
[PE1] ospfv3 10 vpn-instance vpn1
[PE1-ospf-10] router-id 100.100.10.1
[PE1-ospf-10] quit
# Enable OSPFv3 on subinterface Ethernet 1/1.1.
[PE1] interface ethernet 1/1.1
[PE1-Ethernet1/1.1] ospfv3 10 area 0.0.0.0
[PE1-Ethernet1/1.1] quit

```

Verifying the configuration

On PE 1, display the routing table of VPN instance **vpn1**. The output shows that PE 1 has learned the private route of VPN1 through OSPF.

```
[PE1] display ipv6 routing-table vpn-instance vpn1
```

```
Destinations : 6 Routes : 6
```

```

Destination: ::1/128                                Protocol : Direct
NextHop      : ::1                                  Preference: 0
Interface    : InLoop0                             Cost      : 0

```

```

Destination: 2001:2::/64                            Protocol : Direct
NextHop      : ::                                  Preference: 0
Interface    : Eth1/1.1                           Cost      : 0

```

```

Destination: 2001:2::4/128                          Protocol : Direct
NextHop      : ::1                                  Preference: 0
Interface    : InLoop0                             Cost      : 0

```

```

Destination: 2012:1::/64                            Protocol : OSPFv3
NextHop      : FE80::200:5EFF:FE01:1C05             Preference: 15
Interface    : Eth1/1.1                           Cost      : 10

```

```

Destination: FE80::/10                              Protocol : Direct
NextHop      : ::                                  Preference: 0
Interface    : NULL0                               Cost      : 0

```

```

Destination: FF00::/8                               Protocol : Direct
NextHop      : ::                                  Preference: 0
Interface    : NULL0                               Cost      : 0

```

The following output shows that PE 1 has learned the private route of VPN 2 through OSPFv3:

```
[PE1] display ipv6 routing-table vpn-instance vpn2
```

Destinations : 6 Routes : 6

```
Destination: ::1/128                Protocol : Direct
NextHop    : ::1                    Preference: 0
Interface  : InLoop0                Cost      : 0
```

```
Destination: 2002:2::/64            Protocol : Direct
NextHop    : ::                      Preference: 0
Interface  : Eth1/1.2                Cost      : 0
```

```
Destination: 2002:2::4/128          Protocol : Direct
NextHop    : ::1                      Preference: 0
Interface  : InLoop0                 Cost      : 0
```

```
Destination: 2012::/64              Protocol : OSPFv3
NextHop    : FE80::200:5EFF:FE01:1C06 Preference: 15
Interface  : Eth1/1.2                Cost      : 10
```

```
Destination: FE80::/10              Protocol : Direct
NextHop    : ::                        Preference: 0
Interface  : NULL0                    Cost      : 0
```

```
Destination: FF00::/8               Protocol : Direct
NextHop    : ::                          Preference: 0
Interface  : NULL0                      Cost      : 0
```

Now, the routing information for the two VPNs has been redistributed into the routing table on PE 1.

Configuring MPLS OAM

Overview

MPLS Operation, Administration and Maintenance (OAM) provides fault management tools for MPLS data plane connectivity verification, data plane and control plane consistency verification, and fault locating. These fault management tools include the following types:

- **On-demand tools**—Tools that need to be triggered manually, such as MPLS ping and MPLS traceroute.
- **Proactive tools**—Tools that are triggered by the system automatically, such as periodic MPLS traceroute, and MPLS BFD.

You can use these tools to detect and locate faults of LSPs and MPLS TE tunnels.

MPLS ping

MPLS ping tests the connectivity of an LSP tunnel. At the ingress node, MPLS ping adds the label associated with a tunnel into an MPLS echo request and sends it to the egress node over the tunnel. The egress node processes the request and returns an MPLS echo reply to the ingress node. An MPLS echo reply with a success notification indicates that the tunnel is available for data forwarding, and an MPLS echo reply with an error code indicates that the tunnel has failed.

MPLS traceroute

MPLS traceroute displays the path that an MPLS LSP tunnel travels from the ingress node to the egress node to locate errors on the tunnel. MPLS traceroute consecutively sends MPLS echo requests along the LSP tunnel, with the TTL increasing from 1 to a specific value. Each hop along the tunnel returns an MPLS echo reply to the ingress due to TTL timeout so the ingress can collect information about each hop along the tunnel. This information allows you to locate the failed node or access information for each hop, for example, the label allocated by each downstream hop.

Periodic MPLS traceroute

The periodic MPLS traceroute function automatically traces an LSP tunnel at a specific interval. It locates errors on the LSP tunnel, verifies the consistency of the data plane and control plane, and records the detected errors into system logs. You can check the logs to monitor LSP connectivity.

If both BFD and periodic MPLS traceroute are configured for an LSP and the periodic traceroute function detects a data plane and control plane inconsistency, the device deletes the BFD session for the LSP and re-establishes the BFD session based on the control plane.

MPLS BFD

MPLS BFD uses a BFD session to proactively verify the connectivity of an LSP tunnel or an MPLS TE tunnel.

MPLS BFD establishes a BFD session between the ingress and egress of the tunnel to be inspected, adds the label associated with the tunnel into a BFD control packet at the ingress, sends the packet to the egress node over the tunnel, and determines the tunnel status according to the BFD control packet returned by the egress. When BFD detects a connectivity failure, it triggers the pre-configured action, such as FRR or path protection switching, to ensure uninterrupted traffic forwarding.

A BFD session for LSP or MPLS TE tunnel connectivity verification can be established in one of the following modes:

- **Static mode**—You manually specify the local and remote discriminators through command lines to establish the BFD session.
- **Dynamic mode**—The system automatically runs MPLS ping to negotiate the discriminators to establish the BFD session.

In static mode, the egress node returns a BFD control packet to the ingress node through the reverse tunnel. If no reverse tunnel exists, the ingress node cannot receive the BFD control packet, resulting in a verification failure.

In dynamic mode, the egress node returns a BFD control packet to the ingress node through the reverse tunnel. If no reverse tunnel exists, the egress mode returns a BFD packet through IP routing.

Use the static mode to test the connectivity of a pair of LSPs or MPLS TE tunnels in opposite directions (one from local to remote, and the other from remote to local) between two devices. Use the dynamic mode to test the connectivity of one LSP or MPLS TE tunnel from the local device to the remote device.

Protocols and standards

RFC 4379, Detecting Multi-Protocol Label Switched (MPLS) Data Plane Failures

Configuring MPLS OAM for LSP tunnels

To verify LSP connectivity, you can use the **ping mpls ipv4** command or the **tracert mpls ipv4** command to manually trigger LSP connectivity verification, or configure periodic MPLS traceroute or MPLS BFD to verify LSP connectivity.

Configuring MPLS ping for LSPs

Perform the following task in any view:

Task	Command
Use MPLS ping to verify MPLS LSP connectivity for an IPv4 prefix.	ping mpls [-a <i>source-ip</i> -c <i>count</i> -exp <i>exp-value</i> -h <i>ttl-value</i> -m <i>wait-time</i> -r <i>reply-mode</i> -rtos <i>tos-value</i> -s <i>packet-size</i> -t <i>time-out</i> -v] * ipv4 <i>dest-addr mask-length</i> [destination <i>start-address</i> [<i>end-address</i> [<i>address-increment</i>]]]

Configuring MPLS traceroute for LSPs

Perform the following task in any view:

Task	Command
Use MPLS traceroute to trace the LSPs for an IPv4 prefix.	tracert mpls [-a <i>source-ip</i> -exp <i>exp-value</i> -h <i>ttl-value</i> -r <i>reply-mode</i> -rtos <i>tos-value</i> -t <i>time-out</i> -v fec-check] * ipv4 <i>dest-addr mask-length</i> [destination <i>start-address</i> [<i>end-address</i> [<i>address-increment</i>]]]

Configuring periodic MPLS traceroute for LSPs

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable MPLS BFD.	mpls bfd enable	By default, MPLS BFD is disabled.
3. Enable periodic LSP traceroute for an FEC.	mpls periodic-tracert <i>dest-addr mask-length</i> [-a <i>source-ip</i> -exp <i>exp-value</i> -h <i>ttl-value</i> -m <i>wait-time</i> -rtos <i>tos-value</i> -t <i>time-out</i> -u <i>retry-attempt</i> fec-check] *	By default, periodic LSP traceroute is disabled.

Configuring MPLS BFD for LSPs

To configure MPLS BFD for an LSP, configure both the local and remote devices as described in [Table 3](#).

Table 3 Configurations on the local and remote devices

BFD session establishment mode	Node type	Execute the "mpls bfd enable" command?	Execute the "mpls bfd" command?	Configure the discriminator keyword?
Static mode	Local	Yes	Yes	Yes
	Remote	Yes	Yes	Yes
Dynamic mode	Local	Yes	Yes	No
	Remote	Yes	No	N/A

Follow these guidelines to configure BFD for an LSP tunnel:

- To establish the BFD session in static mode, make sure the local discriminator and remote discriminator configured on the local device are the same as the remote discriminator and local discriminator configured on the remote device.
- The source address of the BFD session is the MPLS LSR ID of the local device. Before configuring BFD for the LSP tunnel, configure an MPLS LSR ID for the local device and make sure a route is available on the remote device to reach the MPLS LSR ID.
- If multiple LSPs exist for an FEC, you can either create a BFD session for an LSP by specifying the next hop of the LSP or create a BFD session for each LSP without specifying a next hop.
- On a BFD session established in static mode, the ingress node and egress node both operate in active mode. On a BFD session established in dynamic mode, the egress node operates in active mode and the ingress node operates in passive mode. Executing the **bfd session init-mode** command on the ingress or egress node does not change the node operating mode.

To configure MPLS BFD for LSPs:

Step	Command	Remarks
1. Enter system view.	system-view	N/A
2. Enable MPLS BFD.	mpls bfd enable	By default, MPLS BFD is disabled.
3. Configure BFD to verify LSP connectivity for an FEC.	mpls bfd <i>dest-addr mask-length</i> [nexthop <i>nexthop-address</i> [discriminator local <i>local-id</i> remote <i>remote-id</i>]] [template <i>template-name</i>]	By default, BFD is not configured to verify LSP connectivity for an FEC.

Configuring MPLS OAM for MPLS TE tunnels

MPLS OAM only supports BFD for MPLS TE tunnel connectivity verification.

To run BFD on an MPLS TE tunnel, configure both the local and remote devices as described in [Table 4](#).

Table 4 Configurations on the local and remote devices

BFD session establishment mode	Node type	Execute the "mpls bfd enable" command?	Execute the "mpls bfd" command?	Configure the discriminator keyword?
Static mode	Local	Yes	Yes	Yes
	Remote	Yes	Yes	Yes
Dynamic mode	Local	Yes	Yes	No
	Remote	Yes	No	N/A

Follow these guidelines to configure BFD for an MPLS TE tunnel:

- To establish the BFD session in static mode, make sure the local discriminator and remote discriminator configured on the local device are the same as the remote discriminator and local discriminator configured on the remote device.
- The source address of the BFD session is the MPLS LSR ID of the local device. Before you configure BFD for the LSP tunnel, configure an MPLS LSR ID for the local device and make sure a route is available on the remote device to reach the MPLS LSR ID.
- On a BFD session established in static mode, the ingress node and egress node both operate in active mode. On a BFD session established in dynamic mode, the egress node operates in active mode and the ingress node operates in passive mode. Executing the **bfd session init-mode** command on the ingress or egress node does not change the node operating mode.
- If both BFD for MPLS TE tunnel connectivity verification and BFD for FRR are enabled, set the BFD detection interval for MPLS TE tunnel connectivity verification to be longer than the BFD detection interval for FRR (for example, use BFD to detect RSVP neighbors) to make sure the BFD session for MPLS TE tunnel connectivity verification is not down during an FRR switchover.

To configure MPLS BFD for MPLS TE tunnels:

Step	Command	Remarks
1. Enter system view.	system-view	N/A

Step	Command	Remarks	
2.	Enable MPLS BFD.	mpls bfd enable	By default, MPLS BFD is disabled.
3.	Enter the view of the MPLS TE tunnel interface.	interface tunnel <i>number</i>	N/A
4.	Configure BFD to verify MPLS TE tunnel connectivity.	mpls bfd [discriminator local <i>local-id</i> remote <i>remote-id</i>] [template <i>template-name</i>]	By default, BFD is not configured to verify MPLS TE tunnel connectivity.

Displaying MPLS OAM

Execute **display** commands in any view.

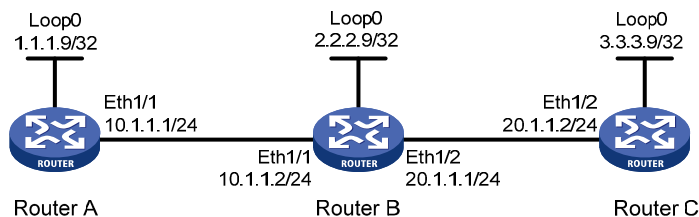
Task	Command
Display BFD information for LSP tunnels or MPLS TE tunnels.	display mpls bfd [ipv4 <i>dest-addr mask-length</i> te tunnel <i>tunnel-number</i>]

BFD for LSP configuration example

Network requirements

Use LDP to establish an LSP from 1.1.1.9/32 to 3.3.3.9/32 and an LSP from 3.3.3.9/32 to 1.1.1.9/32. Use BFD to verify LSP connectivity.

Figure 68 Network diagram



Configuration procedure

1. Configure IP addresses for interfaces. (Details not shown.)
2. Configure OSPF to ensure IP connectivity between the routers:

Configure Router A.

```

<RouterA> system-view
[RouterA] ospf
[RouterA-ospf-1] area 0
[RouterA-ospf-1-area-0.0.0.0] network 1.1.1.9 0.0.0.0
[RouterA-ospf-1-area-0.0.0.0] network 10.1.1.0 0.0.0.255
[RouterA-ospf-1-area-0.0.0.0] quit
[RouterA-ospf-1] quit
  
```

Configure Router B.

```

<RouterB> system-view
  
```



```
[RouterB] ospf
[RouterB-ospf-1] area 0
[RouterB-ospf-1-area-0.0.0.0] network 2.2.2.9 0.0.0.0
[RouterB-ospf-1-area-0.0.0.0] network 10.1.1.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] network 20.1.1.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] quit
[RouterB-ospf-1] quit
```

Configure Router C.

```
<RouterC> system-view
[RouterC] ospf
[RouterC-ospf-1] area 0
[RouterC-ospf-1-area-0.0.0.0] network 3.3.3.9 0.0.0.0
[RouterC-ospf-1-area-0.0.0.0] network 20.1.1.0 0.0.0.255
[RouterC-ospf-1-area-0.0.0.0] quit
[RouterC-ospf-1] quit
```

3. Enable MPLS and LDP:

Configure Router A.

```
[RouterA] mpls lsr-id 1.1.1.9
[RouterA] mpls ldp
[RouterA-ldp] quit
[RouterA] interface ethernet1/1
[RouterA-Ethernet1/1] mpls enable
[RouterA-Ethernet1/1] mpls ldp enable
[RouterA-Ethernet1/1] quit
```

Configure Router B.

```
[RouterB] mpls lsr-id 2.2.2.9
[RouterB] mpls ldp
[RouterB-ldp] quit
[RouterB] interface ethernet1/1
[RouterB-Ethernet1/1] mpls enable
[RouterB-Ethernet1/1] mpls ldp enable
[RouterB-Ethernet1/1] quit
[RouterB] interface ethernet1/2
[RouterB-Ethernet1/2] mpls enable
[RouterB-Ethernet1/2] mpls ldp enable
[RouterB-Ethernet1/2] quit
```

Configure Router C.

```
[RouterC] mpls lsr-id 3.3.3.9
[RouterC] mpls ldp
[RouterC-ldp] quit
[RouterC] interface ethernet1/2
[RouterC-Ethernet1/2] mpls enable
[RouterC-Ethernet1/2] mpls ldp enable
[RouterC-Ethernet1/2] quit
```

4. Enable MPLS BFD, and configure BFD to verify LSP connectivity.

Configure Router A.

```
[RouterA] mpls bfd enable
```

```
[RouterA] mpls bfd 3.3.3.9 32
# Configure Router C.
[RouterC] mpls bfd enable
[RouterC] mpls bfd 1.1.1.9 32
```

Verifying the configuration

Execute the **display mpls bfd** command on Router A and Router C to display BFD information for LSPs. Take Router A as an example:

```
[RouterA] display mpls bfd
Total number of sessions: 2, 2 up, 0 down, 0 init

FEC Type: LSP
FEC Info:
  Destination: 1.1.1.9
  Mask Length: 32
NHLFE ID: -
Local Discr: 514           Remote Discr: 514
Source IP: 1.1.1.9        Destination IP: 3.3.3.9
Session State: Up         Session Role: Active
Template Name: -

FEC Type: LSP
FEC Info:
  Destination: 3.3.3.9
  Mask Length: 32
NHLFE ID: 1025
Local Discr: 513           Remote Discr: 513
Source IP: 1.1.1.9        Destination IP: 127.0.0.1
Session State: Up         Session Role: Passive
Template Name: -
```

The output shows that two BFD sessions have been established between Router A and Router C. One session verifies the connectivity of the LSP from 3.3.3.9/32 to 1.1.1.9/32 and the other session verifies the connectivity of the LSP from 1.1.1.9/32 to 3.3.3.9/32.

Support and other resources

Contacting HP

For worldwide technical support information, see the HP support website:

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

Before contacting HP, collect the following information:

- Product model names and numbers
- Technical support registration number (if applicable)
- Product serial numbers
- Error messages
- Operating system type and revision level
- Detailed questions

Subscription service

HP recommends that you register your product at the Subscriber's Choice for Business website:

<http://www.hp.com/go/wwalerts>

After registering, you will receive email notification of product enhancements, new driver versions, firmware updates, and other product resources.

Related information

Documents

To find related documents, browse to the Manuals page of the HP Business Support Center website:

<http://www.hp.com/support/manuals>

- For related documentation, navigate to the Networking section, and select a networking category.
- For a complete list of acronyms and their definitions, see *HP FlexNetwork Technology Acronyms*.

Websites

- HP.com <http://www.hp.com>
- HP Networking <http://www.hp.com/go/networking>
- HP manuals <http://www.hp.com/support/manuals>
- HP download drivers and software <http://www.hp.com/support/downloads>
- HP software depot <http://www.software.hp.com>
- HP Education <http://www.hp.com/learn>

Conventions

This section describes the conventions used in this documentation set.





Command conventions

Convention	Description
Boldface	Bold text represents commands and keywords that you enter literally as shown.
<i>Italic</i>	<i>Italic</i> text represents arguments that you replace with actual values.
[]	Square brackets enclose syntax choices (keywords or arguments) that are optional.
{ x y ... }	Braces enclose a set of required syntax choices separated by vertical bars, from which you select one.
[x y ...]	Square brackets enclose a set of optional syntax choices separated by vertical bars, from which you select one or none.
{ x y ... } *	Asterisk-marked braces enclose a set of required syntax choices separated by vertical bars, from which you select at least one.
[x y ...] *	Asterisk-marked square brackets enclose optional syntax choices separated by vertical bars, from which you select one choice, multiple choices, or none.
&<1-n>	The argument or keyword and argument combination before the ampersand (&) sign can be entered 1 to n times.
#	A line that starts with a pound (#) sign is comments.








GUI conventions

Convention	Description
Boldface	Window names, button names, field names, and menu items are in bold text. For example, the New User window appears; click OK .
>	Multi-level menus are separated by angle brackets. For example, File > Create > Folder .

Symbols

Convention	Description
 WARNING	An alert that calls attention to important information that if not understood or followed can result in personal injury.
 CAUTION	An alert that calls attention to important information that if not understood or followed can result in data loss, data corruption, or damage to hardware or software.
 IMPORTANT	An alert that calls attention to essential information.
NOTE	An alert that contains additional or supplementary information.
 TIP	An alert that provides helpful information.

Network topology icons

	Represents a generic network device, such as a router, switch, or firewall.
	Represents a routing-capable device, such as a router or Layer 3 switch.
	Represents a generic switch, such as a Layer 2 or Layer 3 switch, or a router that supports Layer 2 forwarding and other Layer 2 features.
	Represents an access controller, a unified wired-WLAN module, or the switching engine on a unified wired-WLAN switch.
	Represents an access point.
	Represents a security product, such as a firewall, a UTM, or a load-balancing or security card that is installed in a device.
	Represents a security card, such as a firewall card, a load-balancing card, or a NetStream card.

Port numbering in examples

The port numbers in this document are for illustration only and might be unavailable on your device.

Index

B C D E I L M O P R S T

B

BFD for LSP configuration example, [281](#)

C

Configuration prerequisites, [11](#)

Configuration procedure, [84](#)

Configuration procedure, [11](#)

Configuring a bidirectional MPLS TE tunnel, [63](#)

Configuring a label acceptance policy, [25](#)

Configuring a label advertisement policy, [24](#)

Configuring a tunnel interface, [53](#)

Configuring a tunnel policy, [99](#)

Configuring an LSP generation policy, [23](#)

Configuring an MPLS TE tunnel to use a dynamic CRLSP, [54](#)

Configuring an MPLS TE tunnel to use a static CRLSP, [54](#)

Configuring an OSPF sham link, [140](#)

Configuring basic IPv6 MPLS L3VPN, [220](#)

Configuring basic MPLS L3VPN, [125](#)

Configuring BGP AS number substitution, [151](#)

Configuring CRLSP backup, [64](#)

Configuring DS-TE, [53](#)

Configuring HoVPN, [139](#)

Configuring inter-AS IPv6 VPN, [229](#)

Configuring inter-AS VPN, [135](#)

Configuring LDP backoff, [22](#)

Configuring LDP GR, [27](#)

Configuring LDP loop detection, [26](#)

Configuring LDP MD5 authentication, [23](#)

Configuring LDP session parameters, [22](#)

Configuring Link Hello parameters, [21](#)

Configuring MPLS MTU, [6](#)

Configuring MPLS OAM for LSP tunnels, [278](#)

Configuring MPLS OAM for MPLS TE tunnels, [280](#)

Configuring MPLS TE FRR, [64](#)

Configuring nested VPN, [138](#)

Configuring routing on an MCE, [142](#)

Configuring routing on an MCE, [231](#)

Configuring RSVP authentication, [95](#)

Configuring RSVP GR, [97](#)

Configuring RSVP hello extension, [95](#)

Configuring RSVP refreshStep

Command

Remarks

, [94](#)

Configuring RSVP Srefresh and reliable RSVP message delivery, [94](#)

Configuring the LDP label distribution control mode, [24](#)

Configuring traffic forwarding, [62](#)

Configuring TTL propagation, [7](#)

Contacting HP, [284](#)

Conventions, [285](#)

D

Displaying and maintaining IPv6 MPLS L3VPN, [239](#)

Displaying and maintaining LDP, [28](#)

Displaying and maintaining MPLS, [9](#)

Displaying and maintaining MPLS L3VPN, [152](#)

Displaying and maintaining MPLS TE, [67](#)

Displaying and maintaining RSVP, [98](#)

Displaying MPLS OAM, [281](#)

Displaying static CRLSPs, [85](#)

Displaying static LSPs, [12](#)

Displaying tunnel information, [101](#)

E

Enabling BFD for RSVP, [97](#)

Enabling LDP, [21](#)

Enabling MPLS, [5](#)

Enabling MPLS TE, [52](#)

Enabling RSVP, [93](#)

Enabling sending of MPLS TTL-expired messages, [9](#)

Enabling SNMP notifications for LDP, [28](#)

Enabling SNMP notifications for MPLS, [9](#)

Enabling SNMP notifications for MPLS L3VPN, [152](#)

Exclusive tunnel configuration example, [101](#)

I

- IPv6 MPLS L3VPN configuration examples, [241](#)
- IPv6 MPLS L3VPN configuration task list, [220](#)

L

- LDP configuration examples, [28](#)
- LDP configuration task list, [20](#)

M

- MPLS configuration task list, [5](#)
- MPLS L3VPN configuration examples, [154](#)
- MPLS L3VPN configuration task list, [124](#)
- MPLS TE configuration examples, [67](#)
- MPLS TE configuration task list, [51](#)

O

- Overview, [277](#)
- Overview, [15](#)
- Overview, [11](#)
- Overview, [43](#)
- Overview, [105](#)
- Overview, [99](#)
- Overview, [90](#)

- Overview, [84](#)
- Overview, [218](#)
- Overview, [1](#)

P

- Preferred tunnel and tunnel selection order configuration example, [103](#)
- Preferred tunnel configuration example, [101](#)
- Protocols and standards, [278](#)

R

- Related information, [284](#)
- Resetting LDP sessions, [27](#)
- RSVP configuration task list, [93](#)

S

- Specifying the label type advertised by the egress, [6](#)
- Specifying the VPN label processing mode on the egress PE, [151](#)
- Static CRLSP configuration example, [85](#)
- Static LSP configuration example, [12](#)

T

- Tunnel selection order configuration example, [102](#)